NTNU
Norwegian University of
Science and Technology

# Ship Motion Control Concepts Considering Actuator Constraints

Ole Nikolai Lyngstadaas

Master of Science in Cybernetics and Robotics
Submission date: June 2018
Supervisor: Morten Breivik, ITK
Co-supervisor: Mikkel Eske Nørgaard Sørensen, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

## Preface

Marine autonomy and automation is a field that has been an interest for me for a long period of time. To achieve the goal of safe and predictable autonomous marine operations, many challenges have to be solved. Autonomous marine systems will start to have a larger presence in our lives in the future, possibly leading to new and more efficient ways to travel, transport and explore the seas. Constraint handling control methods will be an important part of tomorrow's research, as it can contribute to safer and more energy efficient operations.

This thesis is written as a part of a M.Sc. degree in Cybernetics and Robotics at the Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU). I would like to thank my supervisors Morten Breivik and Mikkel Eske Nørgaard Sørensen for their valuable guidance, help and support throughout the project. I would also like to thank my family and friends, for always believing in me and supporting me for the entirety of my studies. Additional thanks go out to the students and everyone affiliated with NTNU for making these years such a great experience, and in particular Tore Egil Sæterdal for the cooperation and the many hours spent in the laboratory together throughout this project. Further thanks goes out to senior engineer Torgeir Wahl at the Department of Marine Technology for his valuable support during the experimental work done in the Marine Cybernetics Laboratory.

The goal of the project was to investigate the effects of handling actuator constraints. This was done through the development of different constraint handling methods which was combined with cascaded feedback controllers. During the semester the supervisors have contributed with guidance through an hourly bi-weekly follow-up meeting where the progress of the thesis and other related topics were discussed. In addition to this, Mikkel assisted with the laboratory experiments.

The work is based upon previous work done by the supervisors, where extensions to their work were developed in order to achieve the above goal. In order to develop a dynamic window-based controller for dynamic positioning, effort was put into extending the 2 degrees of freedom dynamic window-based controller into 3 degrees of freedom. In addition to this, a new magnitude-rate saturation model usable in ship motion control systems was introduced.

Throughout the semester, two laboratory weeks were planned and conducted at the Marine Cybernetics Laboratory at NTNU, where the model ship C/S Inocean Cat I Arctic Drillship was used for the experiments in the ocean basin.

During the first laboratory week, it became clear that it was necessary to make changes to and improve the mathematical ship model, including the thrust allocation model, as the simulated performance of the ship did not correspond well to the actual laboratory performance, which especially had a negative impact on the strictly model-based dynamic window-based controller. Thus, new laboratory performance tests of the model ship were conducted in order to be able to identify and redesign the model parameters, and find actual actuator limitations. In addition to this, it deemed necessary to further improve upon the already implemented laboratory design, as the estimator was not designed to handle lost measurements from the laboratory camera positioning system, resulting in worsening effects on performance. Because of this, extensive effort was put into finding a solution, and a redesign of the estimator was done. These challenges resulted in extra days spent in the laboratory where new implementations were tested in order to have working solutions ready for the second laboratory week.

The work done in this thesis resulted in two publications. In "A Dynamic Window-Based Controller for Dynamic Positioning Satisfying Actuator Magnitude Constraints", the 3 degrees of freedom dynamic window-based controller is presented and tested against a traditional controller, and is accepted to the IFAC CAMS 2018 conference. In "Improvement of Ship Motion Control Using a Magnitude-Rate Saturation Model", accepted to the IEEE CCTA 2018 conference, the magnitude-rate saturation model introduced for ship motion control systems in this thesis is presented, and its performance is tested through laboratory experiments where it is shown that such a model can increase both control accuracy and energy efficiency, and at the same time also reduce actuator wear and tear.

Contributing to two publications during the semester has given an additional workload, but the resulting experience and knowledge with academic work has been a rewarding involvement. In particular, a substantial amount of time was spent on the IEEE CCTA article, where the four first weeks of the semester were spent on writing and doing laboratory work related to the first version of this article.

Ole Nikolai Lyngstadaas
Trondheim, June 11, 2018

## Abstract

Numerous motion controllers and autopilots have been proposed over the years. Most control algorithms found in the literature do not explicitly consider saturation constraints for the actuators. In traditional control theory, an ideal controller might achieve perfect reference tracking in simulations, having no or non-sufficient limitations on the control input. However, in real-life applications it would not be feasible due to limitations in physical output and wear and tear of the actuators.

This thesis presents mathematical modeling of a ship and actuator constraints, and the development and implementation of cascaded feedback controllers, including stability analysis and suggested tuning rules. In addition, a magnitude-rate saturation model is modified to fit ship motion control systems and used to handle actuator constraints. Furthermore, the use of a simplified dynamic window algorithm to handle actuator magnitude constraints for a 3 degrees of freedom dynamic positioning controller for ships, is developed. To accomplish this, a simplified version of the 2 degrees of freedom dynamic window algorithm, where the collision avoidance part of the algorithm is removed, is extended into 3 degrees of freedom. This extended algorithm is then used to design a dynamic window-based controller which guarantees that the velocities remain within a feasible set, while simultaneously respecting the actuator magnitude constraints.

The controllers are tested through numerical simulations and laboratory experiments in the ocean basin at the Marine Cybernetics Laboratory at NTNU. The model-scale ship C/S Inocean Cat I Arctic Drillship is used for the experiments throughout this thesis, and a set of improvements to its mathematical model is presented. A set of performance metrics are used to evaluate the performance of the different controllers in terms of control accuracy and energy efficiency for both pose and velocity tracking, in addition to actuator wear and tear. In addition, the effects of including the presented magnitude-rate saturation model is thoroughly investigated.

## Sammendrag

Mange kontroll- og autopilotsystemer har blitt foreslått gjennom årene. De fleste kontrollalgoritmene som finnes i litteraturen behandler ikke eksplisitt metningsbegrensningene til aktuatorene. I tradisjonell kontrollteori kan en ideell kontroller oppnå perfekt referansefølging i simuleringer, hvor den ikke har noen eller ikke tilstrekkelige begrensninger på kontrollsignalet. I virkelige applikasjoner ville det imidlertidig ikke vært mulig på grunn av fysiske begrensninger og slitasje på aktuatorene.

Denne oppgaven presenterer matematisk modellering av skip og aktuatorbegrensninger, samt design og implementering av kaskadekontrollere, inkludert stabilitetsanalyse og foreslåtte tuningsregler. I tillegg er en mengde- og ratebegresningsmodell modifisert til å passe kontrollsystemer for skip og brukes til å håndtere aktuatorbegrensninger. Videre er en forenklet dynamisk vindualgoritme brukt til å håndtere aktuatorbegrensninger for en dynamisk posisjoneringsskontroller for skip i tre frihetsgrader utviklet. For å oppnå dette blir en dynamisk vindu-algoritme i to frihetsgrader, hvor antikollisjonsdelen av algoritmen er fjernet, utvidet til tre frihetsgrader. Denne utvidede algoritmen brukes så til å designe en dynamisk vindu-basert kontroller som garanterer at hastighetene forblir innenfor et dynamisk mulig sett, samtidig som aktuatorbegrensningene blir overholdt.

Kontrollerne blir testet gjennom numeriske simuleringer og laboratorieforsøk i testbassenget i Marine Cybernetics Laboratory ved NTNU. Modellfartøyet C/S Inocean Cat I Arctic Drillship brukes til forsøkene gjennom hele denne oppgaven, og et sett av forbedringer til den matematiske modellen til fartøyet blir presentert. Et sett med ytelsesmetrikker brukes til å evaluere ytelsen til de forskjellige kontrollerne med hensyn til kontrollnøyaktighet og energieffektivitet for både posisjons- og hastighetsfølging, i tillegg til slitasje på aktuatorene. I tillegg blir effektene av å inkludere den presenterte mengde- og ratebegrensningsmodellen grundig undersøkt.

# Table of Contents

# List of Figures

# List of Tables

# Notation

The notation list is intended as a reference for the reader. Symbols only used in a small part of the thesis are not included in the notation list. All symbols are explained as they are introduced throughout the thesis. Bold symbols denote a vector or a matrix.

| Symbol | Description |
|---|---|
| $(x, y)$ | Cartesian position |
| $\psi$ | Heading angle |
| $(u, v)$ | Body-fixed linear velocities |
| $r$ | Yaw rate |
| $\boldsymbol{\eta}$ | Vessel pose |
| $\boldsymbol{\eta}_t$ | Target vessel pose |
| $\boldsymbol{\nu}$ | Body velocity |
| $\boldsymbol{\nu}_t$ | Target body velocity |
| $\boldsymbol{M}$ | Inertia matrix |
| $\boldsymbol{C}(\boldsymbol{\nu})$ | Coriolis-centripetal matrix |
| $\boldsymbol{D}(\boldsymbol{\nu})$ | Damping matrix |
| $\boldsymbol{R}(\psi)$ | Rotation matrix |
| $\boldsymbol{S}(r)$ | Skew-symmetric matrix |
| $\boldsymbol{\tau}$ | Force vector |
| $\boldsymbol{K}_T$ | Actuator force matrix |
| $\boldsymbol{T}$ | Actuator configuration matrix |
| $\boldsymbol{u}$ | Actuator inputs [V] |
| $\boldsymbol{m}$ | Magnitude constraints |
| $\boldsymbol{r}$ | Rate constraints |
| $\boldsymbol{K}$ | MRS model gain matrix |

Table 1: Notations, continued on next page.

| Symbol | Description |
|---|---|
| $\boldsymbol{z}$ | Error variable |
| $\bar{e}$ | Normed error |
| $V$ | Control Lyapunov function |
| $\boldsymbol{\alpha}(\cdot)$ | Stabilizing function, desired velocity |
| $\boldsymbol{\Gamma}$ | Controller gain matrix |
| $\boldsymbol{\Omega}(\cdot)$ | Nonlinear controller gain matrix |
| $\boldsymbol{K}(\cdot)$ | Nonlinear controller gain function |
| $v_d$ | Desired velocity, CB guidance |
| $v_t$ | Target velocity, CB guidance |
| $\boldsymbol{p}$ | Interceptor position, CB guidance |
| $\boldsymbol{p}_t$ | Target position, CB guidance |
| $\Delta$ | Lookahead distance, tuning variables |
| $T$ | Time constant |
| $\boldsymbol{\nu}_{ss}$ | Steady-state body velocity |
| $\boldsymbol{\nu}^*$ | Current velocity |
| $V_p$ | Set of possible velocities |
| $V_w$ | Dynamic velocity window |
| $V_f$ | Set of dynamically feasible velocities |
| $\boldsymbol{\nu}_d$ | Desired velocity, for the dynamic window |
| $\boldsymbol{\nu}_t$ | Target velocity, for the dynamic window |
| $\boldsymbol{\nu}_f$ | Optimal velocity, for the dynamic window |
| $\boldsymbol{\alpha}_f$ | Feasible velocity, for the dynamic window |
| $\boldsymbol{\alpha}_{DWC}$ | Stabilizing function, desired velocity |

Table 2: Notations, continuation from previous page.

# Abbreviations

The list of abbreviations is intended as a reference for the reader. Abbreviations are explained as they are introduced throughout the thesis.

| **Abbreviations** | **Description** |
|---|---|
| AUV | Autonomous Underwater Vehicle |
| CB | Constant bearing |
| CSAD | C/S Inocean Cat I Drillship |
| CLF | Control Lyapunof function |
| DOF | Degrees-of-freedom |
| DP | Dynamic positioning |
| DW | Dynamic window |
| DWC | Dynamic window-based controller |
| IADC | Integral of absolute differentiated control |
| IAE | Integral of the absolute error |
| IAEW | IAE multiplied by the energy consumption |
| ISS | Input-to-state stable |
| LOS | Line-of-sight |
| LP-LV | Linear pose and velocity feedback |
| NED | North-East-Down |
| MC-Lab | Marine Cybernetics Laboratory |
| MRS | Magnitude-rate-saturation |
| NP-LV | Nonlinear pose and linear velocity feedback |
| NP-NV | Nonlinear pose and velocity feedback |
| UES | Uniformly exponentially stable |
| UGAS | Uniformly asymptotically stable |
| ULES | Uniformly locally exponentially stable |
| UGES | Uniformly globally exponentially stable |

# Chapter 1

# Introduction

## 1.1 Motivation and previous work

Most of today's ships have autopilot systems that usually lead the ship along the desired heading and velocity. Numerous motion controllers and autopilot systems have been developed over the years, but most control systems do not explicitly consider actuator constraints. Examples of traditional motion control designs for ships are given in [1]. Not considering actuator constraints may lead to unsatisfying performance or stability issues, and should be avoided.

Several ways of handling both magnitude and rate constraints have been developed. Handling the constraints may lead to a more energy efficient performance and less wear and tear on the actuators, which are both crucial for a ship's ability to operate at sea over longer periods and to reduce maintenance cost. In [2], model predictive control for systems with actuator magnitude and rate constraints is presented. In [3], the dynamic window algorithm is suggested as a method to perform collision avoidance and deal with constraints imposed by limited velocities and accelerations for mobile robots. The dynamic window algorithm was modified for autonomous underwater vehicles (AUVs) in [4], and showed promising results for handling magnitude and rate constraints for the actuators. In [5], a simplification of the dynamic window algorithm in [4] is developed for a 2 degrees of freedom heading and speed controller, by removing the collision avoidance part of the algorithm.

Building on the algorithm in [5], it is possible to develop a 3 degrees of freedom dynamic window-based controller which can be used for dynamic positioning, while at the same time also satisfying actuator constraints [6]. This is a step in the direction of a state-of-the-art dynamic positioning controller which combines control accuracy, energy efficiency and constraint handling. To develop this controller, the dynamic window algorithm has to be extended into 3 degrees of freedom and combined with a cascaded motion controller, which can be based on the design in [7].

In addition to this, other ways of handling magnitude and rate saturation effects have been developed in the literature. In [8], a magnitude-rate saturation model is derived to address the issue of anti-windup in systems affected by magnitude and rate constraints. By modifying the model in [8], it can be included as a part of a cascaded motion control system for ships to handle actuator constraints, and likely improve energy efficiency and reduce actuator wear and tear [9].

It is then clear that actuator constraint handling is an important and widely researched topic for ship motion control, along with other types of control systems, and that there are much needed research to be done in the field, which will have the capability to potentially further improve overall control performance, efficiency and accuracy.

## 1.2 Problem formulation

The goal of this thesis is to investigate the effects of handling actuator constraints. Firstly, by developing a combination of a dynamic window algorithm and a nonlinear cascaded feedback control algorithm to implement an energy efficient constraint handling controller, and secondly, by implementing other models for the purpose of comparison and a thorough investigation on the effects on ship performance by using such models. Specifically, the following tasks are to be done in the thesis:

- Perform a background and literature review to provide information and relevant references on:
  - Nonlinear feedback control algorithms.
  - Dynamic window-based control and other state-of-the-art constraint handling methods.
  - Marine Cybernetics Laboratory and C/S Inocean Cat I Arctic Drillship.
- Develop and implement a nonlinear cascaded feedback control algorithm as a pose and speed controller for a ship in MATLAB/Simulink.

- Combine a constraint handling algorithm from the literature study with the cascaded feedback controllers to handle actuator constraints.
- Develop and implement a combination of the dynamic window algorithm and a nonlinear cascaded feedback control algorithm as a pose and speed controller for a ship in MATLAB/Simulink to address the problem of magnitude constraints for the actuators.
- Evaluate the closed-loop performance of the combined control algorithms using performance metrics and simulations to investigate the differences in pose and velocity tracking and energy efficiency.
- Test the performance of the algorithms in model-scale experiments in the Marine Cybernetics Laboratory.

## 1.3 Contributions

The contributions of this thesis are:

- An update to the C/S Inocean Cat I Arctic Drillship model, including thrust allocation and actuator limitations.
- Actuator constraint modeling, and a magnitude-rate saturation model used to handle actuator constraints on ships, including a comparison to other methods, an analysis on performance and stability, and suggested tuning rules.
  - The article "Improvement of Ship Motion Control Using a Magnitude-Rate Saturation Model, in Proceedings of the 2nd IEEE Conference on Control Technology and Applications" [9] is based upon this work.
- A review of previously developed nonlinear cascaded feedback controllers, including tuning rules and stability analyses, and the effects on performance by combining these with the above magnitude-rate saturation model.
- The development of a 3 degrees of freedom dynamic window-based controller which satisfies actuator magnitude constraints.
  - The article "A Dynamic Window-Based Controller for Dynamic Positioning Satisfying Actuator Magnitude Constraints, in Proceedings of the 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles" [6] is based upon this work.
- A thorough evaluation of the above control algorithms through a number of numerical simulations and laboratory experiments.
- A fault tolerant estimator designed to handle lost camera positioning measurements.

- Suggestions for alternative uses for the dynamic window algorithm and further research regarding actuator constraint handling methods.

## 1.4 Outline

This thesis is divided into several distinctive chapters. Chapter 2 presents mathematical modeling of a ship, the C/S Inocean Cat I Drillship model and its updated model parameters, actuator constraint modeling and an analysis of different constraint handling methods. Chapter 3 contains constraint handling control methods, including cascaded feedback controllers and the development of a 3 degrees of freedom dynamic window-based controller. Simulation results and an analysis of these, along with the description of several performance metrics, are presented in Chapter 4. Chapter 5 contains information about the Marine Cybernetics Laboratory and the laboratory implementations, in addition to the experimental results and an analysis of these. A thorough examination of the combined results from the numerical simulations and laboratory experiments are discussed in Chapter 6. Some concluding remarks and suggestions for future work are presented in Chapter 7.

# Chapter 2

# Ship modeling and actuator constraints

This chapter contains mathematical modeling of surface vessels, thrust allocation, and actuator constraints. The modeling of the model-scale ship C/S Inocean Cat I Arctic Drillship, hereafter abbreviated CSAD, is presented, and possible ways of modeling and handling actuator constraints are discussed and analysed.

## 2.1 Mathematical modeling of a ship

A ship can be differentially described by 6 degrees of freedom (DOF) equations of motion. The two modes are $(x, y, z)$ and $(\phi, \theta, \psi)$, where the first mode refer to *surge*, *sway*, and *heave*, respectively, which describes the ship's position in a three-dimensional space. The second mode refers to *roll*, *pitch*, and *yaw*, respectively, and describes the orientation of the ship. Assuming that the ship is longitudinally and laterally metacentrically stable for small amplitudes of $\phi = \theta = \dot{\phi} = \dot{\theta} \approx 0$, it is possible to discard the dynamics of roll and pitch. It is also reasonable to assume that the ship floats with $z \approx 0$ in mean, which means that the heave dynamics are discardable as well. The resulting model for the purpose of manoeuvring the ship in the horizontal plane then becomes a 3 DOF model.

If we let an inertial frame be approximated by an earth-fixed reference frame $\{e\}$ called NED (North-East-Down), and a body-fixed frame $\{b\}$ be attached to the ship, the state of the vessel can then be taken as $\boldsymbol{\eta} = [x, y, \psi] \in \mathbb{R}^2 \times \mathbb{S}$, where $\mathbb{S} = [-\pi, \pi)$, and $\boldsymbol{\nu} = [u, v, r] \in \mathbb{R}^3$, where $(x, y)$ is the Cartesian position, $\psi$ is the heading (yaw) angle, $(u, v)$ are the body-fixed linear velocities (surge and sway speed), and $r$ is the yaw rate.

The 3 DOF dynamics of the surface vessel can then be stated as in [1]:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{2.1}$$

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

where

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

is a rotation matrix $\boldsymbol{R} \in SO(3)$, and where $\boldsymbol{M} \in \mathbb{R}^{3\times3}$, $\boldsymbol{C}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$, $\boldsymbol{D}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3]^\top$ represent the inertia matrix, Coriolis and centripetal matrix, damping matrix, and control input vector, respectively. The system matrices are assumed to satisfy the properties $\boldsymbol{M} = \boldsymbol{M}^\top > 0$, $\boldsymbol{C}(\boldsymbol{\nu}) = -\boldsymbol{C}(\boldsymbol{\nu})^\top$ and $\boldsymbol{D}(\boldsymbol{\nu}) > 0$.

### 2.1.1 CSAD model

The model-scale ship CSAD, which is a 1:90 model-scale replica of the full-scale Statoil Cat I Arctic Drillship, with length $L = 2.578\,m$ is shown in Figure 2.1. The following model for CSAD is used for control design and evaluation through both numerical simulations and laboratory experiments in the Marine Cybernetics Laboratory (MC-Lab) throughout this thesis. The numerical values for the model parameters can be found in Table 2.1. The inertia matrix is given as

$$\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_A, \tag{2.4}$$

where

$$\boldsymbol{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix} \tag{2.5}$$

$$\boldsymbol{M}_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix}, \tag{2.6}$$

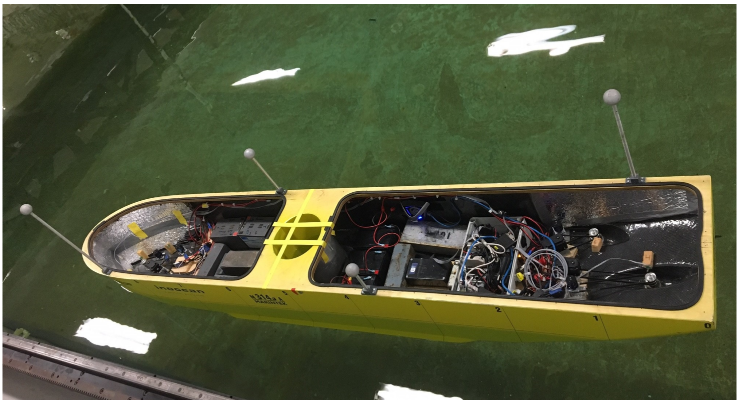


Figure 2.1: C/S Inocean Cat I Arctic Drillship in the MC-Lab.

where $m$ is the mass, $x_g$ is the distance along the x-axis in the body frame from the centre of gravity, and $I_z$ is the moment of inertia about the z-axis in the body frame. The Coriolis and centripetal matrix is given as

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_A(\boldsymbol{\nu}), \tag{2.7}$$

where

$$\boldsymbol{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \tag{2.8}$$

$$\boldsymbol{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -c_{A,13}(\boldsymbol{\nu}) \\ 0 & 0 & c_{A,23}(\boldsymbol{\nu}) \\ c_{A,13}(\boldsymbol{\nu}) & -c_{A,23}(\boldsymbol{\nu}) & 0 \end{bmatrix}, \tag{2.9}$$

with

$$c_{A,13}(\boldsymbol{\nu}) = -Y_{\dot{v}} v - Y_{\dot{r}} r \tag{2.10}$$

$$c_{A,23}(\boldsymbol{\nu}) = -X_{\dot{u}} u. \tag{2.11}$$

Finally, the damping matrix $\boldsymbol{D}(\boldsymbol{\nu})$ is given as

$$\boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{D}_L + \boldsymbol{D}_{NL}(\boldsymbol{\nu}), \tag{2.12}$$

where

$$\boldsymbol{D}_L = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \tag{2.13}$$

$$\boldsymbol{D}_{NL}(\boldsymbol{\nu}) = \begin{bmatrix} d_{NL,11}(\boldsymbol{\nu}) & 0 & 0 \\ 0 & d_{NL,22}(\boldsymbol{\nu}) & d_{NL,23}(\boldsymbol{\nu}) \\ 0 & d_{NL,32}(\boldsymbol{\nu}) & d_{NL,33}(\boldsymbol{\nu}) \end{bmatrix}, \tag{2.14}$$

with

$$d_{NL,11}(\boldsymbol{\nu}) = -X_{|u|u}|u| - X_{uuu}u^2 \tag{2.15}$$

$$d_{NL,22}(\boldsymbol{\nu}) = -Y_{|v|v}|v| - Y_{|r|v}|r| - Y_{vvv}v^2 \tag{2.16}$$

$$d_{NL,23}(\boldsymbol{\nu}) = -Y_{|r|r}|r| - Y_{|v|r}|r| - Y_{rrr}r^2 - Y_{ur}u \tag{2.17}$$

$$d_{NL,32}(\boldsymbol{\nu}) = -N_{|v|v}|v| - N_{|r|v}|v| - N_{vvv}v^2 - N_{uv}u \tag{2.18}$$

$$d_{NL,33}(\boldsymbol{\nu}) = -N_{|r|r}|r| - N_{|v|r}|v| - N_{rrr}r^2 - N_{ur}u. \tag{2.19}$$

To successfully accommodate for the Munk moment, the damping terms

$$Y_{ur} = X_{\dot{u}} \tag{2.20}$$

$$N_{uv} = -(Y_{\dot{v}} - X_{\dot{u}}) \tag{2.21}$$

$$N_{ur} = Y_{\dot{r}}, \tag{2.22}$$

which, when multiplied with surge speed $u$ are linearly increasing with the forward speed, are added to the damping matrix. This will result in a more physically realistic model behaviour, and will ensure that the model will not give rise to a physically impossible motion [5], [20].

### 2.1.2 Experimental tests and updated model parameters for CSAD

Several changes have been done to the numerical model parameters for CSAD throughout the work done in this thesis.

The parameters in Table 2.1 that are marked in bold are the ones that differ from the latest article regarding CSAD [12]. There, the parameters in the added mass matrix $\boldsymbol{M}_A$ were positively signed. This, which is possibly due to a typo, would result in removed mass, which would result in an erroneous impact on the ship's performance in simulations. This mistake was therefore corrected.

Furthermore, the parameters $N_r$, $N_{|r|r}$ and $N_{rrr}$, which relate to the yaw motion, were found to be too small in the original model, which resulted in an infeasible maximum yaw rate. These values had not been scaled to fit a realistic yaw rate, which were found to be approxmimately 5 [deg/s] in the laboratory experiments conducted in the MC-Lab last year [13].

To approximate appropriate values for these parameters, the initial scaling were kept as in [12], with the higher order terms $N_{|r|r}$ and $N_{rrr}$ dependent on $N_r$. The ship model was simulated with a constant maximum yaw moment, see Section 2.3. The value of $N_r$ was then tuned until the model yielded a yaw rate of 5 [deg/s]. In addition, the sign of the parameter $Y_{|v|v}$ was corrected. The final parameter values for the updated CSAD model used in laboratory week 1 are displayed in Table 2.1, where the updated parameters are marked in bold.

Then, at the end of laboratory week 1, several performance tests were conducted to find the maximum velocities and actuator outputs due to poor overall laboratory performance of the model-based dynamic window-based controller throughout the week, and several model discrepancies were found. The model was corrected through the same method as before, using the updated maximum actuator forces and moments, see Section 2.3, and the final values for the updated CSAD model are displayed in Table 2.1. The model behaviour for the laboratory week 1 model and the final model based on the updated thrust allocation model, are shown in Figure 2.2, 2.3 and 2.4. The spikes in these figures are caused by a combination of a poorly calibrated laboratory camera positioning system and a velocity estimator which was not designed to handle the poor calibration. Subsequent work was later done to improve both the camera calibration and estimator. See Section 5.2 for more details. Using this model, the maximum velocities will be 0.4142 [m/s], 0.109 [m/s] and 6.327 [deg/s] in surge, sway and yaw rate, respectively, which corresponds well to the experimental results from the laboratory tests.

| **Parameters** | **Old values [12]** | **Lab week 1 values [9]** | **Updated values** |
|---|---|---|---|
| $L$ | 2.578 | 2.578 | 2.578 |
| $m$ | 127.92 | 127.92 | 127.92 |
| $x_g$ | 0 | **0.0375** | **0.0375** |
| $I_z$ | 61.967 | 61.967 | 61.967 |
| $X_{\dot{u}}$ | 3.262 | **$-3.262$** | **$-10$** |
| $Y_{\dot{v}}$ | 28.89 | **$-28.89$** | **$-105$** |
| $Y_{\dot{r}}$ | 0.525 | **$-0.525$** | **$-0.525$** |
| $N_{\dot{v}}$ | 0.157 | **$-0.157$** | **$-0.157$** |
| $N_{\dot{r}}$ | 13.98 | **$-13.98$** | **$-3.495$** |
| $X_u$ | $-2.332$ | $-2.332$ | **$-5.35$** |
| $X_{\|u\|u}$ | 0 | 0 | 0 |
| $X_{uuu}$ | $-8.557$ | $-8.557$ | **$-19.6312$** |
| $Y_v$ | $-4.673$ | $-4.673$ | **$-10.16$** |
| $Y_{\|v\|v}$ | 0.3976 | **$-0.3976$** | **$-0.8647$** |
| $Y_{vvv}$ | $-313.3$ | $-313.3$ | **$-681.1745$** |
| $N_v$ | 0 | 0 | 0 |
| $N_{\|v\|v}$ | $-0.2088$ | $-0.2088$ | $-0.2088$ |
| $N_{vvv}$ | 0 | 0 | 0 |
| $Y_r$ | $-7.25$ | $-7.25$ | $-7.25$ |
| $Y_{\|r\|r}$ | $-3.450$ | $-3.450$ | $-3.450$ |
| $Y_{rrr}$ | 0 | 0 | 0 |
| $N_r$ | $-0.0168$ | **$-7.141$** | **$-14.55$** |
| $N_{\|r\|r}$ | $-0.0115$ | **$-4.888$** | **$-9.9597$** |
| $N_{rrr}$ | $-0.000358$ | **$-0.152$** | **$-0.3101$** |
| $N_{\|v\|r}$ | 0.08 | 0.08 | 0.08 |
| $N_{\|r\|v}$ | 0.08 | 0.08 | 0.08 |
| $Y_{\|v\|r}$ | $-0.845$ | -0.845 | -0.845 |
| $Y_{\|r\|v}$ | $-0.805$ | -0.805 | -0.805 |

Table 2.1: Numerical values for the ship model parameters for CSAD.

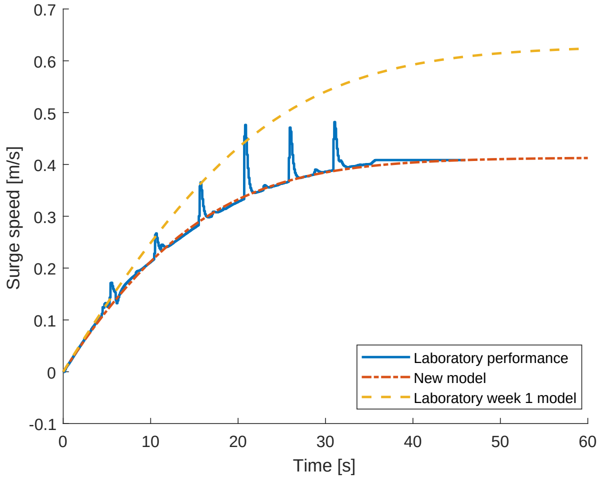


Figure 2.2: Surge speed [m/s] for the updated CSAD models with maximum surge force as input.

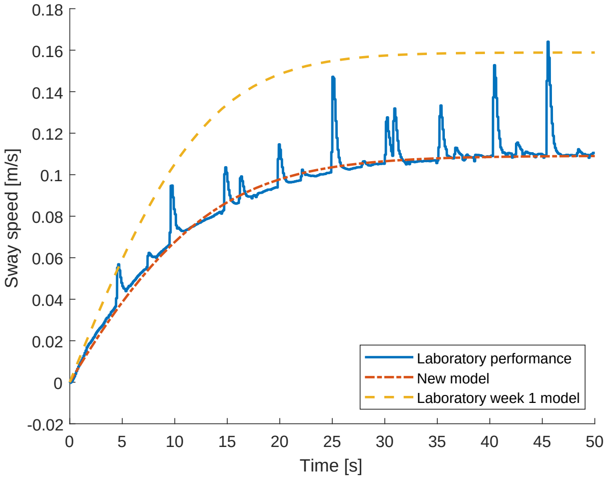


Figure 2.3: Sway speed [m/s] for the updated CSAD models with maximum sway force as input.

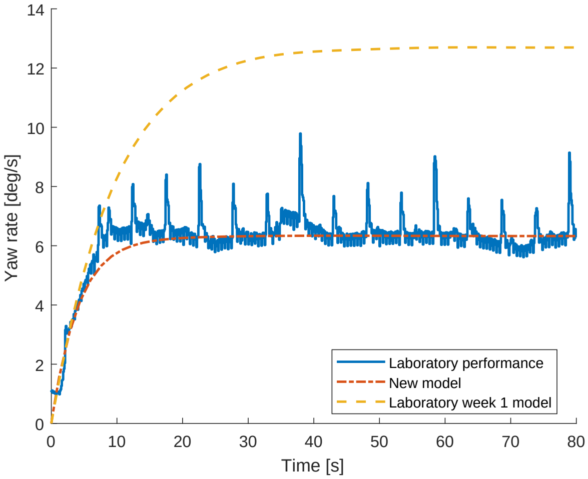


Figure 2.4: Yaw rate [deg/s] for the updated CSAD models with maximum yaw moment as input.

## 2.2 Thrust allocation

The presented ship model describes a fully actuated ship, where the actuator forces and moments are modeled using six azimuth thrusters, $\boldsymbol{u} = [u_1, u_2, u_3, u_4, u_5, u_6]^\top \in \mathbb{R}^6$ [6], [12]. These are related to the input vector $\boldsymbol{\tau}$ through the actuator model

$$\boldsymbol{\tau}(\boldsymbol{u}) = \boldsymbol{T}\boldsymbol{K}_T\boldsymbol{u}, \tag{2.23}$$

where $\boldsymbol{T} \in \mathbb{R}^{3\times 6}$ is an actuator configuration matrix, while $\boldsymbol{K}_T \in \mathbb{R}^{6\times 6}$ is an actuator force matrix. The actuator configuration matrix is

$$\boldsymbol{T} = \begin{bmatrix} c(\delta_1) & c(\delta_2) & c(\delta_3) & c(\delta_4) & c(\delta_5) & c(\delta_6) \\ s(\delta_1) & s(\delta_2) & s(\delta_3) & s(\delta_4) & s(\delta_5) & s(\delta_6) \\ \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 \end{bmatrix}, \tag{2.24}$$

where $c(\delta_i) = \cos(\delta_i)$ and $s(\delta_i) = \sin(\delta_i)$. The constant $\phi_i = L_i \cos(\beta_i)\sin(\delta_i)$, with $L_i = \sqrt{L_{x,i}^2 + L_{y,i}^2}$, where $L_{x,i}$ and $L_{y,i}$ represent the physical placements of the $i$th actuator, and $\beta_i = \tan(L_{x,i}/L_{y,i})$ for $i \in [1, 6]$. The actuator force

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| $L_{x,1}$ | 1.0678 | $L_{y,1}$ | 0.0 |
| $L_{x,2}$ | 0.9344 | $L_{y,2}$ | 0.11 |
| $L_{x,3}$ | 0.9344 | $L_{y,3}$ | -0.11 |
| $L_{x,4}$ | -1.1644 | $L_{y,4}$ | 0.0 |
| $L_{x,5}$ | -0.9911 | $L_{y,5}$ | -0.1644 |
| $L_{x,6}$ | -0.9911 | $L_{y,6}$ | 0.1644 |

Table 2.2: Numerical values for the actuator model parameters.

matrix is given as

$$\boldsymbol{K}_T = \text{diag}([K_{T,1}, K_{T,2}, K_{T,3}, K_{T,4}, K_{T,5}, K_{T,6}]), \tag{2.25}$$

where $K_{T,i} > 0$ is the thrust force from the $i$th propeller. Here, the actuators are fixed to the angles $\boldsymbol{\delta} = [\pi, \pi/4, -\pi/4, 0, 5\pi/4, 3\pi/4]$. The numerical values for $L_{x,i}$ and $L_{y,i}$ are shown in Table 2.2.

## 2.3 Actuator limitations

The six actuators of CSAD are, by design, limited to $u_i \in [-0.5, 0.5]$ [V] [12], where $u_i$ is the actuator input for actuator $i$ in volt. Simulating the vessel with the thrust allocation model for all possible inputs of $\boldsymbol{u} = [u_1, \dots, u_6]$, it is possible to find the actuator forces and moments magnitude limits. These limits will be used in the design of the dynamic window-based controller (DWC) in Section 3.2 and the magnitude-rate saturation (MRS) model given in (2.31) in Section 2.4.2. The simulator setup is shown in Figure 2.5.

The thrust allocation model originally used for this thesis and in laboratory week 1, was given by the thrust coefficients

$$\boldsymbol{K}_T = \text{diag}([0.3763, 0.3901, 0.3776, 0.5641, 0.4799, 0.5588]), \tag{2.26}$$

which resulted in the following actuator magnitude limits: $1.1089\,[N]$ in surge, $0.6387\,[N]$ in sway, and $0.6176\,[Nm]$ in yaw. These values were used for the design of the dynamic window-based controller and the MRS model used in laboratory week 1. However, by the performance tests done in the end of the week, these values were found to be inconsistent with the actual performance of CSAD. The real values were found to be approximately $3.6\,[N]$ in surge, $2.0\,[N]$

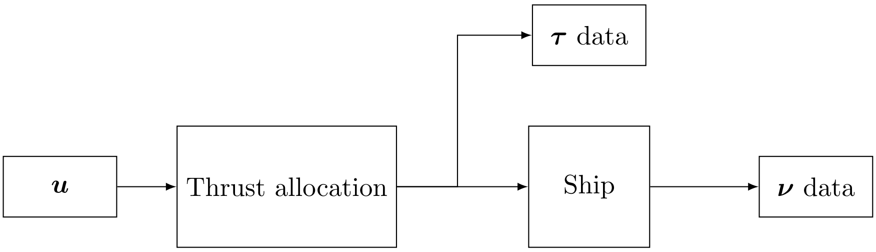


Figure 2.5: Block diagram for the simulator setup used to investigate actuator magnitude constraints $\boldsymbol{\tau}_{max}$ and $\boldsymbol{\tau}_{min}$, and steady-state velocities $\boldsymbol{\nu}$.

in sway, and 1.7 $[Nm]$ in yaw. The thrust coefficient matrix $\boldsymbol{K}_T$ was simply upscaled to fit the measured performance, and a same-valued diagonal matrix was found to be more consistent with the actual actuator performance. By using

$$\boldsymbol{K}_T = \text{diag}([1.491, 1.491, 1.491, 1.491, 1.491, 1.491]), \tag{2.27}$$

a performance which corresponded well to the actual laboratory behaviour was found. The resulting actuator outputs simulated for all values of $\boldsymbol{u}$ are shown in Figure 2.6, giving the actuator magnitude limits 3.5996 [N] in surge, 2.0032 [N] in sway, and 1.7027 [Nm] in yaw. The dynamic window-based controller in this thesis is based upon these actuator limits.

## 2.4 Actuator constraint modeling

Modeling the vessel's actuator constraints is important because it ensures that the controller output remains inside a feasible range of values. Both magnitude and rate constraints will impact a vessel's ability to manoeuvre, and should be handled in the control system to either keep the controller output within desired values, or to give more accurate simulations.

### 2.4.1 Basic methods

Magnitude and rate constraints can easily be implemented using simple well-known methods, as shown below. Combining these models as a cascade system, as shown in Figure 2.7, we can ensure that the controller output will remain within feasible values at all times, and that the vessel can execute the desired task.

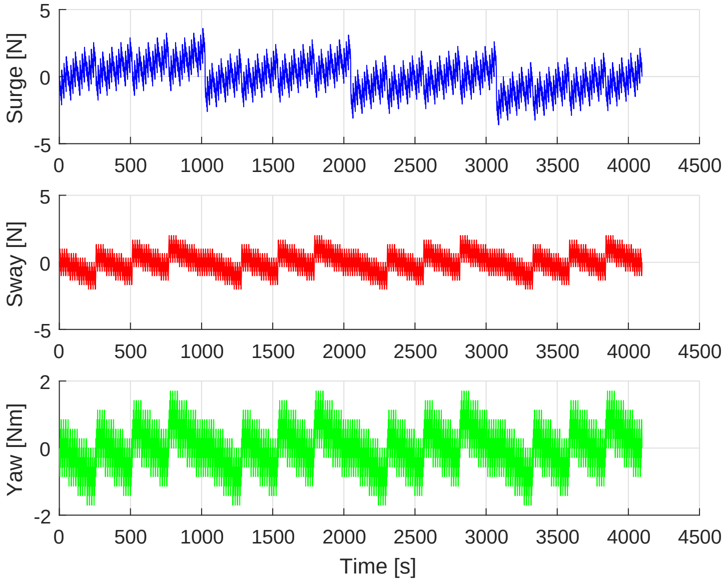


Figure 2.6: Simulated actuator outputs for the presented thrust allocation model.

### Magnitude constraints

The commanded control input with saturation $\boldsymbol{\tau}_s$ is bounded as follows

$$\tau_{s,i}(\tau_{c,i}) = \begin{cases} \tau_{i,min} & \text{if } \tau_{c,i} \leq \tau_{i,min} \\ \tau_i & \text{if } \tau_{i,min} < \tau_{c,i} < \tau_{i,max}, \\ \tau_{i,max} & \text{if } \tau_{c,i} \geq \tau_{i,max} \end{cases} \quad \forall i \in \{1, \dots, 3\}, \tag{2.28}$$

where $\tau_{c,i}$ is the commanded control input with $i \in \{1, 2, 3\}$ without saturation, to control surge, sway and yaw forces and moment, respectively. The saturation limits are represented by $\boldsymbol{\tau}_{min} = [\tau_{1,min}, \tau_{2,min}, \tau_{3,min}]^\top$ and $\boldsymbol{\tau}_{max} = [\tau_{1,max}, \tau_{2,max}, \tau_{3,max}]^\top$ with negative and positive bounded elements, respectively. This model is often referred to as a saturation block in programs such as MATLAB/Simulink.

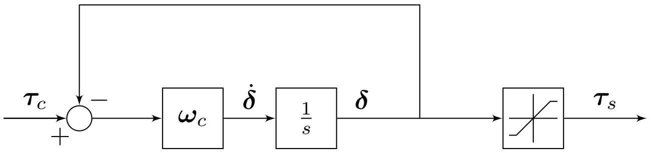


Figure 2.7: Block diagram for the cascaded magnitude and rate saturation model.

**Rate constraints**

The rate constraints can be modeled by a simple first-order low-pass filter. Such a filter has the differential equation

$$\begin{aligned}\dot{\boldsymbol{\delta}} &= \boldsymbol{\omega_c}(\boldsymbol{\tau}_c - \boldsymbol{\delta}) \\ \boldsymbol{\tau}_s &= \boldsymbol{\delta},\end{aligned} \tag{2.29}$$

where $\boldsymbol{\tau}_c$, $\boldsymbol{\delta}$, $\boldsymbol{\tau}_s$ and $\boldsymbol{\omega}_c$ are the input, the state, the output and time constant of the filter, which represents the actuator rate constraints, respectively.

### 2.4.2 A magnitude-rate saturation model

An approach to model magnitude-rate saturation, is to introduce the following dynamical system with a discontinuous right-hand side

$$\begin{aligned}\dot{\boldsymbol{\delta}} &= \mathrm{diag}(\boldsymbol{r})\mathrm{sign}(\mathrm{sat}_m(\boldsymbol{\tau}_c - \boldsymbol{\delta})) \\ \boldsymbol{\tau}_{mrs} &= \boldsymbol{\delta},\end{aligned} \tag{2.30}$$

where $\boldsymbol{\tau}_c$, $\boldsymbol{\delta}$ and $\boldsymbol{\tau}_{mrs}$ are the input, the state, and the output of the MRS, respectively, and where $\boldsymbol{m} = [m_1, \dots, m_p]$ and $\boldsymbol{r} = [r_1, \dots, r_p]$, $p \in \{1, 2, 3\}$, are vectors whose strictly positive components specify the magnitude and rate constraints, respectively.

This discontinuous model exactly describes the MRS effects, but does, however, require special care due to its discontinuity, and is not so easy to handle from an analysis and synthesis point of view. The model is often approximated by a high gain model where the *sign*-function is replaced by a high gain followed by a saturation. Other solutions are also available, and is discussed in greater detail in [8].

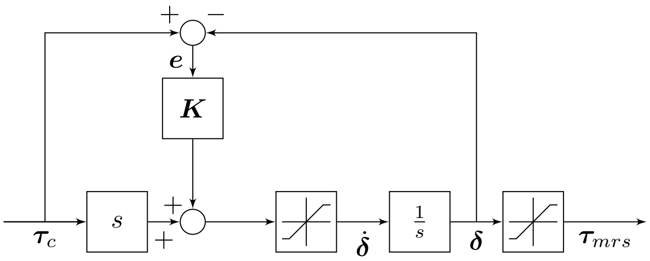


Figure 2.8: Block diagram for the MRS model (2.31).

To get around the problems mentioned above, another MRS model is proposed:

$$\begin{aligned} \dot{\boldsymbol{\delta}} &= sat_{\boldsymbol{r}}(\dot{\boldsymbol{\tau}}_c + \boldsymbol{K}(\boldsymbol{\tau}_c - \boldsymbol{\delta})) \\ \boldsymbol{\tau}_{mrs} &= sat_{\boldsymbol{m}}(\boldsymbol{\delta}), \end{aligned} \tag{2.31}$$

where $\boldsymbol{K} > 0$ is a diagonal tuning matrix. The matrix is introduced in order to avoid an unstable cancellation between the derivative operator $s$ and the integrator in Figure 2.8, where the block diagram for the MRS model is shown. The derivative of the input, $\dot{\boldsymbol{\tau}}_c$, is supposed to exist and can be calculated using numerical derivation. The saturation limits $sat_{\boldsymbol{r}}$ and $sat_{\boldsymbol{m}}$ are modeled as the saturation block given above, and contain the vessel's rate and magnitude constraints, respectively. See [8] for further details.

The main difference between (2.30) and (2.31) is that in (2.30) the magnitude of the input is limited before entering the rate limiter, whereas in (2.31), the rate is limited first and then magnitude next. As a consequence of this difference, the state $\boldsymbol{\delta}$ of (2.30) will never exceed the magnitude bound $\boldsymbol{m}$, but the state of (2.31) can exceed the bound $\boldsymbol{m}$, although its output $\boldsymbol{\tau}_{mrs}$ never does.

For systems that are likely to be subject to integral-windup, which can happen when a large change in controller setpoints occur, or when an ideal output is physically impossible, model (2.31) can effectively be extended with an external signal $\boldsymbol{v}$ to prevent such windup problems. This extension to (2.31) is given by

$$\begin{aligned} \dot{\boldsymbol{\delta}} &= sat_{\boldsymbol{r}}(\dot{\boldsymbol{\tau}}_c + \boldsymbol{K}(\boldsymbol{\tau}_c - \boldsymbol{\delta}) + \boldsymbol{v}) \\ \boldsymbol{\tau}_{mrs} &= sat_{\boldsymbol{m}}(\boldsymbol{\delta}), \end{aligned} \tag{2.32}$$

and is discussed in greater detail in [8]. As we are not looking to handle integral-windup effcts, $\boldsymbol{v} = \boldsymbol{0}$ for the remainder of this thesis.

Model (2.30) and (2.31) require more accurate knowledge about the rate constraints of the vessel's actuators than the combined model (2.28) and (2.29). However, all three models will be implemented and tested through simulations, and the best performing model will be chosen and then combined in cascade with ship motion control systems presented in this thesis to investigate notable effects on performance when considering actuator saturation effects.

### 2.4.3 Analysis of the constraint models

To test the behaviour of the different MRS models, they will be subject to the following test scenarios:

$$\text{Scenario 1: } \tau_1 = \begin{cases} 4\cos(t/2), & \text{if } t < 20 \\ 2\cos(t/4), & \text{if } t \geq 20 \end{cases} \tag{2.33}$$

$$\text{Scenario 2: } \tau_1 = \begin{cases} 2.5, & \text{if } t < 10 \\ -2,5, & \text{if } 10 < t < 20 \\ 4, & \text{if } 20 < t < 30 \\ -4, & \text{if } 30 < t < 40 \\ 1, & \text{if } t \geq 40, \end{cases} \tag{2.34}$$

where the tracking of surge force $\tau_1$ will be used for the analysis.

#### Choice of actuator constraint parameters

The actuator constraint parameters can be tuned in order to affect the ship's performance in a desired way. While limiting the magnitude limits of the actuators can be a good way of saving energy, and in many cases, if not severely reduced, not lead to any drastic changes in performance, tuning the rate constraints can both lead to a more energy efficient performance and less wear and tear for the actuators as the actuators will react less to small disturbances, which usually lead to actuator twitching and an unnecessary use of energy without actually improving accuracy or overall performance [9]. The use of an actuator constraint model, such as the proposed MRS model in (2.31), can therefore, if tuned properly to fit the desired operational environment, be beneficial in many scenarios where accuracy and energy efficiency is more important than achieving the fastest possible convergence toward the target destination.

### Choice of gain matrix $\boldsymbol{K}$ for the MRS model

The diagonal matrix $\boldsymbol{K} = diag([K_{1,1}, K_{2,2}, K_{3,3}])$ is given as a free parameter and is introduced in order to avoid unstable cancellations during linear operation. The choice of this parameter affects the speed of the inner-loop in the MRS model, and should be chosen based on the desired tracking performance. An important observation is that neither of the elements of the matrix can be equal to 1, and thus also $\boldsymbol{K} \neq \boldsymbol{I}$. A $K_{i,i} > 1$, $i \in \{1, 2, 3\}$, will ensure an accurate tracking of the desired rate, while a $K_{i,i} < 1$ will enforce a slower convergence towards the commanded thrust. This can be beneficial for systems where energy efficiency is most important, because the performance will be too slow to follow target spikes in the commanded thrust.

### Simulation of constraint handling methods

Using the same constraints, the different constraint handling methods have been simulated for Scenario 1 in (2.33) and Scenario 2 in (2.34). The results are shown in Figure 2.9 and 2.10. By inspecting the simulation results, its clear that both the alternative constraint handling methods outperform the low-pass filter model. We can see that all the models keep the values within the magnitude and rate constraints of the system. However, model (2.30) becomes almost discontinuous in parts of the simulation, which is realistically infeasible for a system with rate constraints, which means that this model will be neglected. The proposed MRS model in (2.31) is therefore the overall best model to handle actuator constraints, and will be used for numerical simulations and laboratory experiments.

### Suggested tuning rules

Although you can use the MRS model in (2.31) to mimic a system's constraints to get more accurate simulations, the main use of this model in this thesis is to investigate energy efficiency and overall performance increase by limiting the actuator outputs.

Maximizing the output from an actuator is usually very inefficient in terms of energy usage, and often does not contribute a lot to increased performance. Since energy efficiency is in focus, it is suggested to limit the magnitude outputs of the actuators at 80 percent of their maximum values, giving

$$\boldsymbol{m} = [0.8m_{a,1}, 0.8m_{a,2}, 0.8m_{a,3}], \tag{2.35}$$

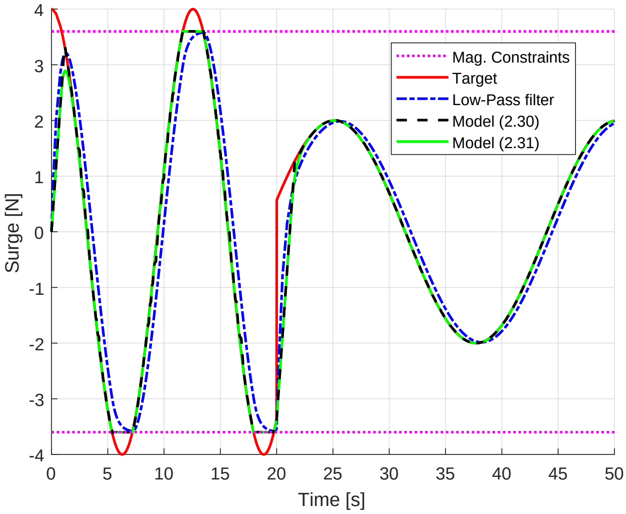


Figure 2.9: Test of actuator constraint methods for scenario 1.

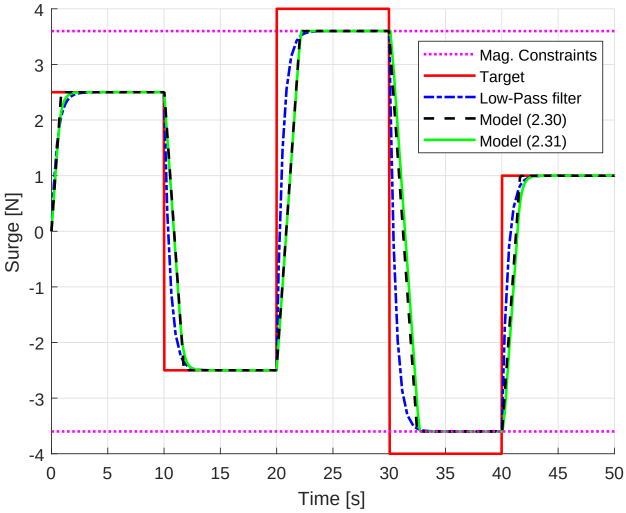


Figure 2.10: Test of actuator constraint methods for scenario 2.

where, specifically for CSAD, $\boldsymbol{m_a} = [3.5996, 2.0032, 1.7027]$ are the actual magnitude limits of the actuators in surge [N], sway [N] and yaw [Nm], respectively.

The rate constraints should be selected as

$$\boldsymbol{r} = [m_1/t_{mrs}, m_2/t_{mrs}, m_3/t_{mrs}], \tag{2.36}$$

where $t_{mrs}$ is the desired time [s] for the actuators to go from zero to maximum thrust allowed by $\boldsymbol{m}$. Here, $t_{mrs} = 1$, as it seemed to fit the desired performance for CSAD.

The gain matrix $\boldsymbol{K}$ can be chosen as

$$\boldsymbol{K} = \text{diag}([K_{1,1}, \frac{m_2}{m_1}K_{1,1}, \frac{m_3}{m_1}K_{1,1}]), \tag{2.37}$$

where the original ratio between the actuator outputs in surge [N], sway [N], and yaw [Nm] has been retained. Under normal operations it is desired to have all the diagonal elements $K_{i,i} > 1$ to ensure a fast tracking of the target signal in all 3 degrees of freedom. Here, $K_{1,1} = 5$ is chosen to make all the elements of $\boldsymbol{K}$ sufficiently large.

A low valued gain matrix $\boldsymbol{K}$ can be beneficial if there are a lot of disturbances in the system, such as spikes in the target signal. The system then might ignore many of the smaller spikes, and thus save energy. Having a low $K_{1,1}$ gain can then be useful for a vessel which highly values energy efficiency over a fast and responsive performance. For comparison purposes, the MRS model for a high and a low value of $K_{1,1}$ is shown in Figure 2.11 and 2.12. The main focus of this thesis will be on $K_{1,1}$ as presented above, but $K_{1,1} = 0.8$ will be tested experimentally in the laboratory.

Note that this tuning is adjusted for the updated ship model and new thrust allocation model, and equals the tuning used in laboratory week 2. A discussion regarding this tuning is presented in Chapter 5 and Chapter 6.

## 2.5 Chapter summary

In this chapter, mathematical modeling of a ship was presented, along with the CSAD model. Necessary changes done to the model were declared and discussed, along with the updated model parameters. The thrust allocation used for the experiments and its actuator limitations were explained in detail. Then, several methods of handling actuator constraints were introduced and analysed, and suggested tuning rules for the constraint handling methods were presented.

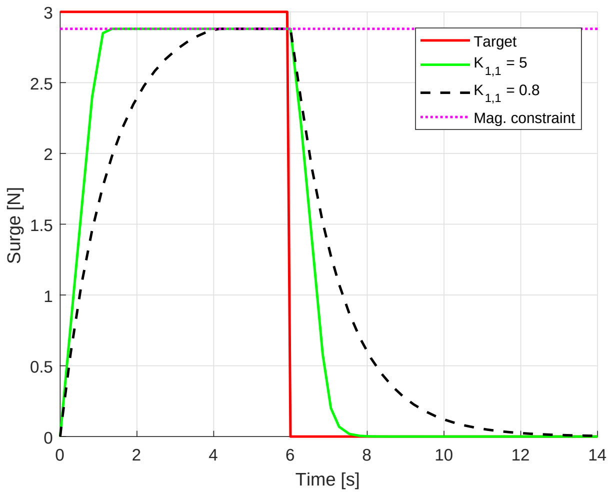


Figure 2.11: Test of different $K_{1,1}$ gains for the MRS model using a square wave.

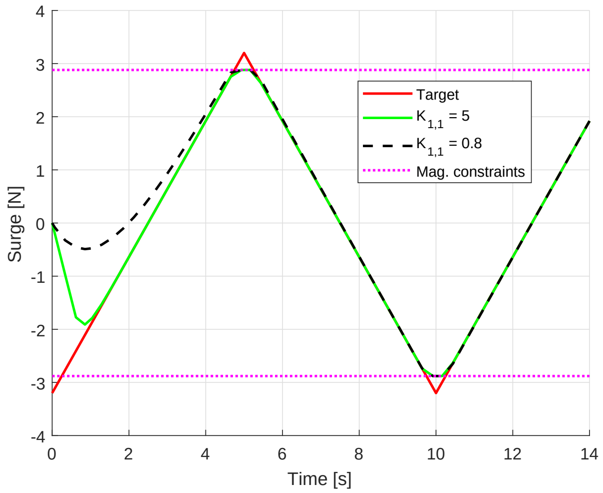


Figure 2.12: Test of different $K_{1,1}$ gains for the MRS model using a triangle wave.

# Chapter 3

# Constraint handling control methods

This chapter contains feedback control design for various control systems, including stability analysis. The effects of using nonlinear feedback terms will be investigated through simulations and performance metrics in Chapter 4, and by laboratory experiments in Chapter 5, where also the effects on performance by using an MRS model will be investigated. In addition to this, dynamic window-based controllers which satisfies actuator magnitude constraints for the use in 2 DOF and 3 DOF operations are presented.

## 3.1 Cascaded feedback controllers

The control objective is to make $\tilde{\boldsymbol{\eta}}(t) \triangleq \boldsymbol{\eta}(t) - \boldsymbol{\eta}_t(t) \to 0$ as $t \to \infty$ and $\tilde{\boldsymbol{\nu}}(t) \triangleq \boldsymbol{\nu}(t) - \boldsymbol{\nu}_t(t) \to 0$ as $t \to \infty$, where $\boldsymbol{\eta}_t = [x_t(t), y_t(t), \psi_t(t)]^\top \in \mathbb{R}^2 \times \mathbb{S}$ is the pose associated with the target point, which is $\mathcal{C}^2$ and bounded and twice differentiable, and $\boldsymbol{\nu}_t = [u_t(t), v_t(t), r_t(t)] \in \mathbb{R}^3$ is the target velocity. The motion of the target is typically defined by a human or generated by a guidance system.

By using a design similar to the backstepping method, as in [16], [19], where the control design is divided into two stages, including the definition of new state variables and deriving the control laws through control Lyapunov functions (CLFs), but where we omit the coupling between the pose and velocity control loops, we end up with a cascaded system. This cascaded system corresponds to a classical inner-outer loop guidance and control structure, where the outer loop handles the kinematics and the inner loop the kinetics. The total system can then be analysed by cascade theory, such as in [11] and [21]. It should be noted that the nonlinear feedback control concept presented here is an initial step toward handling actuator constraints.

The feedback control design in this section is based on the work done in [7] and [15].

It is desirable to investigate the effects of using nonlinear feedback terms, and thus we design and investigate three combinations of linear and nonlinear feedback terms.

For notational simplicity, the time $t$ is omitted in the rest of this chapter.

### 3.1.1 Linear pose and velocity feedback (LP-LV)

Defining the error variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ as

$$\boldsymbol{z}_1 \triangleq \boldsymbol{R}^\top(\psi)(\boldsymbol{\eta} - \boldsymbol{\eta}_t) \tag{3.1}$$

$$\boldsymbol{z}_2 \triangleq \boldsymbol{\nu} - \boldsymbol{\alpha}, \tag{3.2}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^3$ is a vector of stabilising functions, which can be interpreted as a desired velocity and which is to be designed later.

For the kinematic control part, we choose the positive definite CLF as

$$V_1 \triangleq \frac{1}{2}\boldsymbol{z}_1^\top \boldsymbol{z}_1. \tag{3.3}$$

Then, the derivative of $V_1$ with respect to the $\boldsymbol{z}_1$-dynamics becomes

$$\begin{aligned} \dot{V}_1 &= \boldsymbol{z}_1^\top \dot{\boldsymbol{z}}_1 \\ &= \boldsymbol{z}_1^\top(\boldsymbol{S}(r)^\top \boldsymbol{R}^\top(\psi)(\boldsymbol{\eta} - \boldsymbol{\eta}_t) + \boldsymbol{R}^\top(\psi)(\dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_t)) \\ &= \boldsymbol{z}_1^\top(\boldsymbol{S}(r)^\top \boldsymbol{z}_1 + \boldsymbol{R}^\top(\psi)(\dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_t)), \end{aligned} \tag{3.4}$$

where

$$\boldsymbol{S}(r) = \begin{bmatrix} 0 & -r & 0 \\ r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.5}$$

is a skew-symmetric matrix satisfying $\boldsymbol{z}_1^\top \boldsymbol{S}(r)\boldsymbol{z}_1 = 0$, $\forall \boldsymbol{z}_1$. This gives

$$\dot{V}_1 = \boldsymbol{z}_1^\top(\boldsymbol{\nu} - \boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t). \tag{3.6}$$

Using (3.2), we get

$$\begin{aligned}\dot{V}_1 &= \boldsymbol{z}_1^\top(\boldsymbol{z}_2 + \boldsymbol{\alpha} - \boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t) \\ &= \boldsymbol{z}_1^\top \boldsymbol{z}_2 + \boldsymbol{z}_1^\top(\boldsymbol{\alpha} - \boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t),\end{aligned} \tag{3.7}$$

where the stabilising function is chosen as

$$\boldsymbol{\alpha} = \boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t - \boldsymbol{\Gamma}_1\boldsymbol{z}_1, \tag{3.8}$$

where $\boldsymbol{\Gamma}_1 = \boldsymbol{\Gamma}_1^\top > 0$, which results in

$$\dot{V}_1 = -\boldsymbol{z}_1^\top\boldsymbol{\Gamma}_1\boldsymbol{z}_1 + \boldsymbol{z}_1^\top\boldsymbol{z}_2. \tag{3.9}$$

Considering this as a system on the form $\dot{x} = f(t, x, u)$. If $\boldsymbol{z}_2$ is seen as an input with $\boldsymbol{z}_2 = \boldsymbol{0}$, it can be concluded that the origin of the system is uniformly globally exponentially stable (UGES). It can then be concluded from Lemma 4.6 from [10], as repeated in Lemma 1 below, that the subsystem

$$\dot{\boldsymbol{z}}_1 = \boldsymbol{S}(r)^\top\boldsymbol{z}_1 - \boldsymbol{\Gamma}_1\boldsymbol{z}_1 + \boldsymbol{z}_2 \tag{3.10}$$

is input-to-state stable (ISS). Note that (3.9) shows that $\boldsymbol{S}(r)$ in (3.10) does not affect the ISS property.

**Lemma 1** *Suppose $f(t, x, u)$ is continuously differentiable and globally Lipschitz in $(x, u)$, uniformly in $t$. If the unforced system $\dot{x} = f(t, x, 0)$ has a globally exponentially stable equilibrium point at the origin $x = 0$, then the system $\dot{x} = f(t, x, u)$ is input-to-state stable.*

For the kinetic control, the $\boldsymbol{z}_2$-dynamics can be written as

$$\begin{aligned}\boldsymbol{M}\dot{\boldsymbol{z}}_2 &= \boldsymbol{M}(\dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\alpha}}) \\ &= \boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \boldsymbol{M}\dot{\boldsymbol{\alpha}},\end{aligned} \tag{3.11}$$

where the time derivative of (3.8) becomes

$$\dot{\boldsymbol{\alpha}} = \boldsymbol{R}^\top(\psi)\ddot{\boldsymbol{\eta}}_t + \boldsymbol{S}(r)^\top\boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t - \boldsymbol{\Gamma}_1\dot{\boldsymbol{z}}_1, \tag{3.12}$$

where $\boldsymbol{\eta}_t$ is the pose of the target point and $\dot{\boldsymbol{z}}_1$ is given by (3.10). The CLF for $\boldsymbol{z}_2$ is then defined as

$$V_2 \triangleq \frac{1}{2}\boldsymbol{z}_2^\top\boldsymbol{M}\boldsymbol{z}_2. \tag{3.13}$$

The derivative of (3.13) becomes

$$\begin{aligned}\dot{V}_2 &= \boldsymbol{z}_2^\top \boldsymbol{M}\dot{\boldsymbol{z}}_2 \\ &= \boldsymbol{z}_2^\top(\boldsymbol{\tau} - \boldsymbol{C}\boldsymbol{\nu} - \boldsymbol{D}\boldsymbol{\nu} - \boldsymbol{M}\dot{\boldsymbol{\alpha}}),\end{aligned} \tag{3.14}$$

where the following simplifications has and will be applied for the rest of this chapter to achieve notational brevity: $\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}$, $\boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{D}$, $\boldsymbol{R}(\psi) = \boldsymbol{R}$, and $\boldsymbol{S}(r) = \boldsymbol{S}$. Using this, the control input can be chosen as

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\nu} + \boldsymbol{D}\boldsymbol{\nu} - \boldsymbol{\Gamma}_2\boldsymbol{z}_2, \tag{3.15}$$

where $\boldsymbol{\Gamma}_2 > 0$, which results in

$$\dot{V}_2 = -\boldsymbol{z}_2^\top\boldsymbol{\Gamma}_2\boldsymbol{z}_2 < 0, \tag{3.16}$$

which makes the origin of the $\boldsymbol{z}_2$-dynamics

$$\dot{\boldsymbol{z}}_2 = -\boldsymbol{M}^{-1}\boldsymbol{\Gamma}_2\boldsymbol{z}_2 \tag{3.17}$$

UGES.

However, as there are to be experimental tests in the MC-Lab, choosing the control input as

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{\Gamma}_2\boldsymbol{z}_2 \tag{3.18}$$

is beneficial due to possible signal errors on $\boldsymbol{\nu}$. This does however change (3.17) to

$$\dot{\boldsymbol{z}}_2 = -\boldsymbol{M}^{-1}(\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{\Gamma}_2)\boldsymbol{z}_2, \tag{3.19}$$

where the convergence rate of the $\boldsymbol{z}_2$-dynamics becomes influenced by the vessel's $\boldsymbol{C}$ and $\boldsymbol{D}$ matrices, but which also is UGES in the origin.

### Stability analysis

The total closed-loop dynamics become

$$\begin{aligned}\dot{\boldsymbol{z}}_1 &= \boldsymbol{S}^\top\boldsymbol{z}_1 - \boldsymbol{\Gamma}_1\boldsymbol{z}_1 + \boldsymbol{z}_2 \\ \dot{\boldsymbol{z}}_2 &= -\boldsymbol{M}^{-1}(\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{\Gamma}_2)\boldsymbol{z}_2.\end{aligned} \tag{3.20}$$

**Theorem 2** *The origin $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\boldsymbol{0}, \boldsymbol{0})$ of the overall system (3.20) is UGES.*

*Proof (modified from [15]):* Using Theorem 2.1 and Proposition 2.3 from [21], and knowing that both subsystems are separately UGES if the $\boldsymbol{z}_1$-dynamics in (3.20) are unperturbed ($\boldsymbol{z}_2 = \boldsymbol{0}$), and the kinematic control loop has linear growth in the perturbation term $\boldsymbol{z}_2$, it can be concluded that the origin of the total cascade system $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\boldsymbol{0}, \boldsymbol{0})$ is UGES as well.

However, it is important to note that UGES is not physically possible to achieve since the saturation constraints for the actuators limit the achievable control performance. In the following sections, saturation constraints for the actuators will be considered in the control design by introducing nonlinear feedback terms.

### 3.1.2 Nonlinear pose and linear velocity feedback (NP-LV)

By now introducing a nonlinear pose feedback inspired by a constant bearing (CB) guidance concept, which was originally developed for interceptor missiles, but then used for vessel control in [22], we can improve the overall pose control performance. CB guidance is a two-point guidance scheme where the interceptor is supposed to align to the relative interceptor-target velocity along the line-of-sight (LOS) vector between the interceptor and the target.

CB guidance is usually implemented as proportional navigation, where you make the rotation rate of the interceptor velocity directly proportional to the rotation rate of the interceptor target LOS. Another method is to implement CB guidance through the direct velocity assignment

$$\boldsymbol{v}_d = \boldsymbol{v}_t - \kappa(\boldsymbol{p}) \frac{\tilde{\boldsymbol{p}}}{|\tilde{\boldsymbol{p}}|}, \tag{3.21}$$

where $\boldsymbol{v}_t \in \mathbb{R}^2$ is the target velocity and

$$\tilde{\boldsymbol{p}} \triangleq \boldsymbol{p} - \boldsymbol{p}_t \tag{3.22}$$

is the LOS vector between the interceptor position $\boldsymbol{p} = [x, y]^\top \in \mathbb{R}^2$ and the target position $\boldsymbol{p}_t = [x_t, y_t]^\top \in \mathbb{R}^2$, such that

$$|\tilde{\boldsymbol{p}}| \triangleq \sqrt{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{p}}} \geq 0 \tag{3.23}$$

is the Euclidean length of $\tilde{\boldsymbol{p}}$. Furthermore, $\kappa \geq 0$ can be chosen as

$$\kappa(\boldsymbol{p}) = U_{a,max} \frac{|\tilde{\boldsymbol{p}}|}{\sqrt{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{p}} + \Delta_{\tilde{p}}^2}}, \tag{3.24}$$

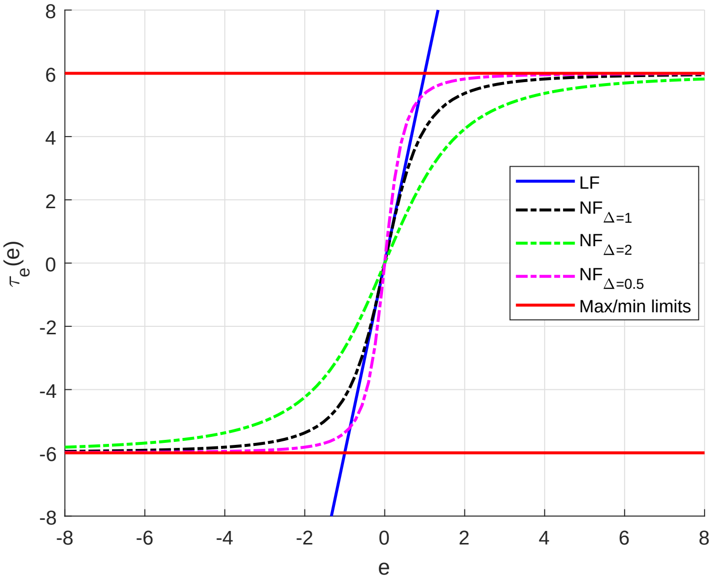


Figure 3.1: LF represents a linear feedback term as a function of the control error $e$, while NF represents a nonlinear feedback term based on a sigmoid function of $e$ using a tuning parameter $\Delta$, such as in (3.21)-(3.24).

where $U_{a,max} > 0$ is the maximum approach speed towards the target and $\Delta_{\tilde{p}} > 0$ is a tuning parameter which affects transient convergence behaviour between the interceptor and target.

The result of using such nonlinear feedback is shown for a scalar error $e \in \mathbb{R}$ in Figure 3.1, where the effects of varying $\Delta$-gains are shown. The linear and nonlinear functions displayed in Figure 3.1 are respectively given as $\tau(e) = \kappa(e)$ and $\tau(e) = \kappa \frac{e}{\sqrt{e^2 + \Delta^2}}$, where $\kappa = 6$.

Introducing the CB guidance nonlinear feedback into the controller, the stabilising function becomes

$$\boldsymbol{\alpha} = \boldsymbol{R}^{\top} \dot{\boldsymbol{\eta}}_t - \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1, \tag{3.25}$$

where

$$\boldsymbol{K}_1(\boldsymbol{z}_1) \triangleq \boldsymbol{\Gamma}_1 \boldsymbol{\Omega}(\boldsymbol{z}_1), \tag{3.26}$$

and

$$\boldsymbol{\Omega}(\boldsymbol{z}_1) = \begin{bmatrix} \dfrac{1}{\sqrt{\boldsymbol{z}_{1,\tilde{p}}^\top \boldsymbol{z}_{1,\tilde{p}} + \Delta_{\tilde{p}}^2}} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \dfrac{1}{\sqrt{z_{1,\tilde{\psi}}^2 + \Delta_{\tilde{\psi}^2}}} \end{bmatrix}, \tag{3.27}$$

and, as before, $\boldsymbol{\Gamma}_1 > 0$, and where $\boldsymbol{z}_{1,\tilde{p}} \triangleq \left[z_{1,1}, z_{1,2}\right]^\top$, $z_{1,\tilde{\psi}} \triangleq z_{1,3}$, $\Delta_{\tilde{p}} > 0$ and $\Delta_{\tilde{\psi}} > 0$. It is possible to choose

$$\boldsymbol{\Omega}(\boldsymbol{z}_1) = \frac{1}{\sqrt{\boldsymbol{z}_1^\top \boldsymbol{z}_1 + \Delta^2}} \mathbf{I}_{3\times 3}, \tag{3.28}$$

if $\Delta_{\tilde{p}} = \Delta_{\tilde{\psi}} = \Delta > 0$, but then it is not possible to define a different transient behaviour for the position and heading.

Choosing (3.27), the derivative of (3.25) becomes

$$\dot{\boldsymbol{\alpha}} = \boldsymbol{R}^\top \ddot{\boldsymbol{\eta}}_t + \boldsymbol{S}^\top \boldsymbol{R}^\top \dot{\boldsymbol{\eta}}_t - \dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1, \tag{3.29}$$

where the derivative of (3.26) is given by

$$\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1) = -\boldsymbol{\Gamma}_1 \dot{\boldsymbol{\Omega}}(\boldsymbol{z}_1) = -\boldsymbol{\Gamma}_1 \begin{bmatrix} \dfrac{\boldsymbol{z}_{1,\tilde{p}}^\top \dot{\boldsymbol{z}}_{1,\tilde{p}}}{(\boldsymbol{z}_{1,\tilde{p}}^\top \dot{\boldsymbol{z}}_{1,\tilde{p}} + \Delta_{\tilde{p}}^2)^{\frac{3}{2}}} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \dfrac{z_{1,\tilde{\psi}}^\top \dot{z}_{1,\tilde{\psi}}}{(z_{1,\tilde{\psi}}^2 + \Delta_{\tilde{\psi}}^2)^{\frac{3}{2}}} \end{bmatrix}. \tag{3.30}$$

**Stability analysis**

The total closed-loop dynamics now become

$$\begin{aligned} \dot{\boldsymbol{z}}_1 &= \boldsymbol{S}^\top \boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 + \boldsymbol{z}_2 \\ \dot{\boldsymbol{z}}_2 &= -\boldsymbol{M}^{-1}(\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{\Gamma}_2)\boldsymbol{z}_2. \end{aligned} \tag{3.31}$$

Here, we have that

$$|\boldsymbol{z}_1| \gg 1 \Rightarrow \dot{\boldsymbol{z}}_1 \approx \boldsymbol{S}^\top \boldsymbol{z}_1 - \boldsymbol{\Gamma}_1 \boldsymbol{\rho}(\boldsymbol{z}_1) + \boldsymbol{z}_2, \tag{3.32}$$

where $\boldsymbol{\rho} = \mathrm{col}(z_{1,\tilde{p}}/|z_{1,\tilde{p}}|, z_{1,\tilde{\psi}}/|z_{1,\tilde{\psi}}|)$ is a vectorial sign-like function that saturates the error $\boldsymbol{z}_1$, and

$$|\boldsymbol{z}_1| \approx 0 \Rightarrow \dot{\boldsymbol{z}}_1 = \boldsymbol{S}^\top \boldsymbol{z}_1 - \boldsymbol{\Gamma}_1 \begin{bmatrix} \dfrac{1}{\Delta_{\tilde{p}}}\mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \dfrac{1}{\Delta_{\tilde{\psi}}} \end{bmatrix} \boldsymbol{z}_1 + \boldsymbol{z}_2, \tag{3.33}$$

where the linear dynamics are recovered, scaled by $\Delta_i$, around the origin.

**Theorem 3** *The origin* $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\mathbf{0}, \mathbf{0})$ *is UGAS, and on each compact set* $B \subset \mathbb{R}^6$ *containing the origin, it is UES.*

*Proof (modified from [15]):* We have that if the $\boldsymbol{z}_2$-dynamics is UGES, the unperturbed $\boldsymbol{z}_1$-dynamics is UGAS, since (3.3) is $\mathcal{C}^1$ and positive definite. $\dot{V}_1$ is negative definite $\forall \boldsymbol{z}_1 \neq 0$ and (3.3) is radially unbounded $\forall \boldsymbol{z}_1 \neq 0$.

Next, it can be shown that the $\boldsymbol{z}_1$-subsystem is growth restricted by satisfying Assumption 7 in [21], where $\alpha_4(s) = s$, $\alpha_5(s) = 1$, $\alpha_1(s) = \frac{1}{2}s^2$, and $\alpha_6(s) = \sqrt{2s}$. Additionally, Assumption 8 in [21] is satisfied by having $\lambda = 2$, $V = \frac{1}{2}\boldsymbol{z}_1^\top \boldsymbol{z}_1$, and $W_1(\boldsymbol{z}_1) = \boldsymbol{z}_1^\top \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1$. With these conditions satisfied, Theorem 2.3 in [21] states that the origin $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\mathbf{0}, \mathbf{0})$ is UGAS.

For $|\boldsymbol{z}(t_0)| \in B$ and the definition of UGAS, there exists $L > 0$ so that $|\boldsymbol{z}(t)| \leq L$, $\forall t \geq t_0 \geq 0$. This also means that $|\boldsymbol{z}_{1,\tilde{p}}| \leq L$ and $|\boldsymbol{z}_{1,\tilde{\psi}}| \leq L$, which further implies that $\exists \gamma > 0$ such that $\boldsymbol{\Gamma}_1 \boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1 > \gamma \mathbf{I}$. Using the quadratic Lyapunov function

$$V(\boldsymbol{z}_1, \boldsymbol{z}_2) = \boldsymbol{z}_1^\top \boldsymbol{z}_1 + \frac{1}{2} b_2 \boldsymbol{z}_2^\top \boldsymbol{M} \boldsymbol{z}_2, \tag{3.34}$$

we get

$$\begin{aligned}
\dot{V} &= 2\boldsymbol{z}_1^\top[-\boldsymbol{S}\boldsymbol{z}_1 - \boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{z}_1 + \boldsymbol{z}_2] + b_2 \boldsymbol{z}_2^\top \boldsymbol{M}[-\boldsymbol{M}^{-1}(\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{\Gamma}_2)\boldsymbol{z}_2] \\
&= -\boldsymbol{z}_1[\boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1]\boldsymbol{z}_1 + 2\boldsymbol{z}_1^\top \boldsymbol{z}_2 - b_2 \boldsymbol{z}_2^\top(\boldsymbol{D} + \boldsymbol{\Gamma}_2)\boldsymbol{z}_2 \\
&= -\boldsymbol{z}_1[\boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1]\boldsymbol{z}_1 + 2\boldsymbol{z}_1^\top \boldsymbol{z}_2 - b_2 \boldsymbol{z}_2^\top(\boldsymbol{\Theta})\boldsymbol{z}_2 \\
&\leq -\gamma|\boldsymbol{z}_1|^2 + 2|\boldsymbol{z}_1||\boldsymbol{z}_2| - b_2\lambda_{\min}(\boldsymbol{\Theta})|\boldsymbol{z}_2|^2 \qquad (3.35)\\
&= -\frac{\gamma}{2}|\boldsymbol{z}_1|^2 - \frac{\gamma}{2}|\boldsymbol{z}_1|^2 + \kappa|\boldsymbol{z}_1|^2 + \frac{1}{\kappa}|\boldsymbol{z}_2|^2 - b_2\lambda_{\min}(\boldsymbol{\Theta})|\boldsymbol{z}_2|^2 \\
&= -\frac{\gamma}{2}|\boldsymbol{z}_1|^2 - \Big(b_2\lambda_{\min}(\boldsymbol{\Theta}) - \frac{2}{\gamma}\Big)|\boldsymbol{z}_2|^2,
\end{aligned}$$

where $\boldsymbol{\Theta} = \boldsymbol{D} + \boldsymbol{\Gamma}_2$, where $\boldsymbol{C}$ is cancelled out because it is skew-symmetric and thus $\boldsymbol{z}_2^\top \boldsymbol{C} \boldsymbol{z}_2 = 0$, and where we used Young's inequality $xy \leq \kappa x^2 + \frac{1}{4\kappa} y^2$, $\kappa = \frac{\gamma}{2} > 0$. Choosing for instance

$$b_2 = \frac{1}{\lambda_{\min}(\boldsymbol{\Theta})} \Big( \frac{\gamma}{2} + \frac{2}{\gamma} \Big), \tag{3.36}$$

gives

$$\dot{V} \leq -\frac{\gamma}{2} |\boldsymbol{z}|^2, \tag{3.37}$$

which proves UES on the compact set $B$.

Note that UGAS shows that the stability of this system is global. The second part of the theorem shows that for any practical set $B$ of initial conditions, the convergence is in fact exponential.

### 3.1.3 Nonlinear pose and velocity feedback (NP-NV)

By now also introducing a nonlinear velocity feedback, the control law (3.18) changes to

$$\boldsymbol{\tau} = \boldsymbol{M} \dot{\boldsymbol{\alpha}} + \boldsymbol{C} \boldsymbol{\alpha} + \boldsymbol{D} \boldsymbol{\alpha} - \boldsymbol{K}_2(\boldsymbol{z}_2) \boldsymbol{z}_2, \tag{3.38}$$

where

$$\boldsymbol{K}_2(\boldsymbol{z}_2) = \boldsymbol{\Gamma}_2 \boldsymbol{\Omega}_2(\boldsymbol{z}_2) = \boldsymbol{\Gamma}_2 \begin{bmatrix} \frac{1}{\sqrt{\boldsymbol{z}_{2,\tilde{v}}^\top \boldsymbol{z}_{2,\tilde{v}} + \Delta_{\tilde{v}}^2}} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \frac{1}{\sqrt{z_{2,\tilde{r}}^2 + \Delta_{\tilde{r}^2}}} \end{bmatrix}, \tag{3.39}$$

and, as before, $\boldsymbol{\Gamma}_2 > 0$, and where $\boldsymbol{z}_{2,\tilde{v}} \triangleq \big[ z_{2,1}, z_{2,2} \big]^\top$, $z_{2,\tilde{r}} \triangleq z_{2,3}$, $\Delta_{\tilde{v}} > 0$ and $\Delta_{\tilde{r}} > 0$.

#### Stability analysis

The total closed-loop dynamics now become

$$\begin{aligned} \dot{\boldsymbol{z}}_1 &= \boldsymbol{S}^\top \boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1) \boldsymbol{z}_1 + \boldsymbol{z}_2 \\ \dot{\boldsymbol{z}}_2 &= -\boldsymbol{M}^{-1} (\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{K}_2(\boldsymbol{z}_2)) \boldsymbol{z}_2. \end{aligned} \tag{3.40}$$

**Theorem 4** *The origin* $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\mathbf{0}, \mathbf{0})$ *is UGAS, and on each compact set* $B \subset \mathbb{R}^6$ *containing the origin, it is UES.*

*Proof (modified from [15]):* The $\boldsymbol{z}_2$-dynamics can be proven to be UGAS using Theorem 4.9 in [10], since (3.13) is $\mathcal{C}^1$ and positive definite, $\dot{V}_2$ is negative definite $\forall \boldsymbol{z}_2 \neq 0$ and (3.13) is radially unbounded $\forall \boldsymbol{z}_2 \neq 0$. A similar conclusion can be made for the unperturbed $\boldsymbol{z}_1$-dynamics.

Next, it can be shown that the $\boldsymbol{z}_1$-subsystem is growth restricted by satisfying Assumption 7 in [21], where $\alpha_4(s) = s$, $\alpha_5(s) = 1$, $\alpha_1(s) = \frac{1}{2}s^2$, and $\alpha_6(s) = \sqrt{2s}$. Additionally, Assumption 8 in [21] is satisfied by having $\lambda = 2$, $V = \frac{1}{2}\boldsymbol{z}_1^\top \boldsymbol{z}_1$, and $W_1(\boldsymbol{z}_1) = \boldsymbol{z}_1^\top \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1$. With these conditions satisfied, Theorem 2.3 in [21] states that the origin $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\boldsymbol{0}, \boldsymbol{0})$ is UGAS.

For $|\boldsymbol{z}(t_0)| \in B$ and the definition of UGAS, there exists $L > 0$ so that $|\boldsymbol{z}(t)| \leq L$, $\forall t \geq t_0 \geq 0$. This also means that $|\boldsymbol{z}_{1,\tilde{p}}| \leq L$, $|\boldsymbol{z}_{1,\tilde{\psi}}| \leq L$, $|\boldsymbol{z}_{2,\tilde{v}}| \leq L$, and $|\boldsymbol{z}_{2,\tilde{r}}| \leq L$, which further implies that $\exists \gamma_i > 0$ such that $\boldsymbol{\Gamma}_i\boldsymbol{\Omega}(\boldsymbol{z}_i) + \boldsymbol{\Omega}(\boldsymbol{z}_i)\boldsymbol{\Gamma}_i > \gamma_i\mathbf{I}$, $i = 1, 2$. Using the quadratic Lyapunov function

$$V(\boldsymbol{z}_1, \boldsymbol{z}_2) = \boldsymbol{z}_1^\top \boldsymbol{z}_1 + b_2\boldsymbol{z}_2^\top \boldsymbol{M}\boldsymbol{z}_2, \tag{3.41}$$

we get

$$\begin{aligned}
\dot{V} &= 2\boldsymbol{z}_1^\top[-\boldsymbol{S}\boldsymbol{z}_1 - \boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{z}_1 + \boldsymbol{z}_2] + 2b_2\boldsymbol{z}_2^\top\boldsymbol{M}[-\boldsymbol{M}^{-1}(\boldsymbol{C} + \boldsymbol{D} + \boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2))\boldsymbol{z}_2] \\
&= -\boldsymbol{z}_1[\boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1]\boldsymbol{z}_1 + 2\boldsymbol{z}_1^\top\boldsymbol{z}_2 - 2b_2\boldsymbol{z}_2^\top[\boldsymbol{D} + \boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2)]\boldsymbol{z}_2 \\
&= -\boldsymbol{z}_1[\boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1]\boldsymbol{z}_1 + 2\boldsymbol{z}_1^\top\boldsymbol{z}_2 - 2b_2\boldsymbol{z}_2^\top\mathbf{I}\boldsymbol{\Lambda}(\boldsymbol{z}_2)\boldsymbol{z}_2 \\
&= -\boldsymbol{z}_1[\boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1) + \boldsymbol{\Omega}(\boldsymbol{z}_1)\boldsymbol{\Gamma}_1]\boldsymbol{z}_1 + 2\boldsymbol{z}_1^\top\boldsymbol{z}_2 - b_2\boldsymbol{z}_2^\top[\mathbf{I}\boldsymbol{\Lambda}(\boldsymbol{z}_2) + \boldsymbol{\Lambda}(\boldsymbol{z}_2)\mathbf{I}]\boldsymbol{z}_2 \\
&\leq -\gamma_1|\boldsymbol{z}_1|^2 + 2|\boldsymbol{z}_1||\boldsymbol{z}_2| - b_2\gamma_2|\boldsymbol{z}_2|^2 \\
&= -\frac{\gamma_1}{2}|\boldsymbol{z}_1|^2 - \frac{\gamma_1}{2}|\boldsymbol{z}_1|^2 + \kappa|\boldsymbol{z}_1|^2 + \frac{1}{\kappa}|\boldsymbol{z}_2|^2 - b_2\gamma_2|\boldsymbol{z}_2|^2 \\
&= -\frac{\gamma_1}{2}|\boldsymbol{z}_1|^2 - \Big(b_2\gamma_2 - \frac{2}{\gamma_1}\Big)|\boldsymbol{z}_2|^2,
\end{aligned} \tag{3.42}$$

where $\mathbf{I}\boldsymbol{\Lambda}(\boldsymbol{z}_2) = \boldsymbol{D} + \boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2)$ and $\boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2)\boldsymbol{z}_2 = \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2$, where $\boldsymbol{C}$ is cancelled out because it is skew-symmetric and thus $\boldsymbol{z}_2^\top\boldsymbol{C}\boldsymbol{z}_2 = 0$, which holds for the proof as stated above because $2\boldsymbol{D} + \boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2) + \boldsymbol{\Omega}(\boldsymbol{z}_2)\boldsymbol{\Gamma}_2 > \boldsymbol{\Gamma}_2\boldsymbol{\Omega}(\boldsymbol{z}_2) + \boldsymbol{\Omega}(\boldsymbol{z}_2)\boldsymbol{\Gamma}_2$ since it is assumed that $\boldsymbol{D} > 0$, and where we used Young's inequality. Choosing for instance

$$b_2 = \frac{1}{\gamma_2}\Big(\frac{\gamma_1}{2} + \frac{2}{\gamma_1}\Big)|\boldsymbol{z}_2|^2, \tag{3.43}$$

gives

$$\dot{V} \leq -\frac{\gamma_1}{2}|\boldsymbol{z}|^2, \tag{3.44}$$

which proves UES on the compact set $B$.

### 3.1.4 Tuning rules

The following tuning rules are suggested to get the desired behaviour [15]. Considering the $\boldsymbol{z}_1$-dynamics in (3.20), (3.31), and (3.40), it can be seen that the choice of the $\boldsymbol{\Gamma}_1$ gain matrix determines the time constant of the $\boldsymbol{z}_1$-dynamics with linear feedback. For simplicity, the time constants matrix for the pose subsystem is defined as

$$\boldsymbol{T}_1 \triangleq \boldsymbol{\Gamma}_1^{-1}, \tag{3.45}$$

where

$$\boldsymbol{T}_1 = \begin{bmatrix} T_p & 0 & 0 \\ 0 & T_p & 0 \\ 0 & 0 & T_\psi \end{bmatrix}, \tag{3.46}$$

where $T_p > 0$ and $T_\psi > 0$ are the time constants for position control and heading control, respectively. A similar observation can be made regarding the $\boldsymbol{z}_2$-dynamics in (3.20), (3.31), and (3.40), where the choice of the $\boldsymbol{\Gamma}_2$ gain matrix determines the time constants of the $\boldsymbol{z}_2$-dynamics with linear feedback. Again, for simplicity, the time constants are defined as

$$\boldsymbol{T}_2 \triangleq \boldsymbol{M}\boldsymbol{\Gamma}_2^{-1}, \tag{3.47}$$

where

$$\boldsymbol{T}_2 = \begin{bmatrix} T_v & 0 & 0 \\ 0 & T_v & 0 \\ 0 & 0 & T_r \end{bmatrix}, \tag{3.48}$$

where $T_v > 0$ and $T_r > 0$ are the time constants for speed control and yaw rate control, respectively.

It is desired that the kinetic subsystem has faster dynamics than the kinematic subsystem. Hence, the kinetic dynamics must have smaller time constants, which means that

$$T_v < T_p \tag{3.49}$$

$$T_r < T_\psi. \tag{3.50}$$

Next, $\Delta_p$, $\Delta_\psi$, $\Delta_v$ and $\Delta_r$, must be chosen. The control parameter $\Delta$ is usually known as the lookahead distance in line of sight guidance [23]. In [29], it is shown that a small $\Delta$-value corresponds to fast convergence to the path, but with a large overshoot, while a large $\Delta$-value reduces overshoot and results in a smooth but slow convergence. Here, the $\Delta$-values scale the linear feedback gains and therefore the time constants of the linear region. If the $\Delta$-values are equal to 1, they will result in the same response as the linear controllers in the linear region. If they are larger than 1, they will give a slower response in this region. If they are chosen smaller than 1, they will give a faster response. Hence, the $\Delta$-values for the NP and NV feedback terms must be chosen such that the conditions in (3.49)-(3.50) are not broken in the linear region, where the time constants will be related to $\mathbf{\Gamma}_1(1,1)/\Delta_p$, $\mathbf{\Gamma}_1(3,3)/\Delta_\psi$, $\mathbf{\Gamma}_2(1,1)/\Delta_v$, and $\mathbf{\Gamma}_2(3,3)/\Delta_r$.

### 3.1.5 Stability in cascade with the MRS model

The MRS model is included as a part of the cascaded control system as shown in Figure 3.2. As the three controllers, LP-LV, NP-LV and NP-NV, are shown to be stable, it is the necessary to prove that the MRS model and its effects will not have an impact on the stability of the system.

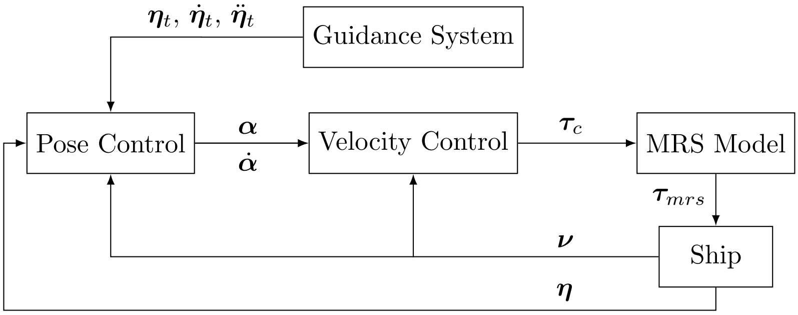


Figure 3.2: Block diagram for the ship control systems as presented above, with the included MRS model.

When investigating the stability analysis of the MRS model in a cascaded system, there are some properties of (2.32) that the following lemma guarantees. These are

- item 1 guarantees that the output of (2.32) always satisfies the magnitude and rate limits;
- item 2 guarantees that if (2.32) is properly initialized, then the output of $\boldsymbol{\tau}_{mrs}$ of (2.32) coincides with its input $\boldsymbol{\tau}_c$ as long as $(\boldsymbol{\tau}_c, \dot{\boldsymbol{\tau}}_c)$ never exceed the magnitude and rate limits;

- item 3 then complements item 2, by specifying that $(\boldsymbol{\tau}_{mrs} - \boldsymbol{\tau}_c)$ will remain an $\mathcal{L}_2$ signal despite the occurrence of either a wrong initialization, and/or a non-zero $\boldsymbol{v} \in \mathcal{L}_2$, and/or a choice of $\boldsymbol{\tau}_c$ which exceeds the magnitude or rate limits (restricted by an arbitrarily small amount) in such a way that the excess of saturation is an $\mathcal{L}_2$ signal.

**Lemma 5** *Modified from [8].*
*Given any signal $\boldsymbol{\tau}_c(\cdot)$ such that $\dot{\boldsymbol{\tau}}_c$ is well defined for almost all $t$, for any diagonal matrix $\boldsymbol{K} > 0$, the MRS model (2.32) satisfies the following:*

*(i) for any measurable $\boldsymbol{v}(\cdot)$, $sat_{\boldsymbol{m}}(\boldsymbol{\tau}_{mrs}(t)) = \boldsymbol{\tau}_{mrs}(t)$ and $sat_{\boldsymbol{r}}(\dot{\boldsymbol{\tau}}_{mrs}(t)) = \dot{\boldsymbol{\tau}}_{mrs}(t)$, for almost all $t \geq 0$;*

*(ii) if $\boldsymbol{\delta}(0) = \boldsymbol{\tau}_c(0)$, $sat_{\boldsymbol{m}}(\boldsymbol{\tau}_c(t)) = \boldsymbol{\tau}_c(t)$, $sat_{\boldsymbol{r}}(\dot{\boldsymbol{\tau}}_c(t)) = \dot{\boldsymbol{\tau}}_c(t)$ and $\boldsymbol{v}(t) = 0$, $\forall t \geq 0$, then $\boldsymbol{\tau}_{mrs}(t) = \boldsymbol{\tau}_c(t)$, $\forall t \geq 0$;*

*(iii) for any $\boldsymbol{\delta}(0)$, if $||\boldsymbol{v}||_2 < \infty$ and $\exists \epsilon > 0$ such that $||\boldsymbol{\tau}_c - sat_{\boldsymbol{m}(1-\epsilon)}(\boldsymbol{\tau}_c)||_2 < \infty$ and $||\dot{\boldsymbol{\tau}}_c - sat_{\boldsymbol{r}(1-\epsilon)}(\dot{\boldsymbol{\tau}}_c)||_2 < \infty$, then $||\boldsymbol{\tau}_{mrs} - \boldsymbol{\tau}_c||_2 < \infty$.*

As the proof of Lemma 5 is quite long and mathematically tedious, for the purpose of readability it is not included here and can be found in [8].

It is then proven that if the input signal to the MRS model, which can be interpreted as a nonlinear filter, is an $\mathcal{L}_2$ signal, the output signal of the MRS model will be an $\mathcal{L}_2$ signal as well, so it can be concluded that the MRS model does not alter the stability properties of the system.

## 3.2 Dynamic window-based controllers

The dynamic window approach was originally developed for robotics motion planning in [3], and is an online collision avoidance strategy for mobile robots. The approach is derived directly from the dynamics of the robot, and is designed to deal with constraints imposed by limited feasible velocities and accelerations. The algorithm consists of two main steps. First, generate a valid search space, then select an optimal solution within the search space which can be reached within a short time interval and are free from collisions. The optimization goal is to select a pose and velocity that brings the robot to the desired position without colliding with any object on the way.

Here, a step-by-step design procedure for a simplified version of the dynamic window (DW) algorithm, where the collision avoidance part of the algorithm is removed, is presented. First, a 2 degrees of freedom version for surge speed and yaw rate, based on the work done in [4], [5] and [17] is explained. Then, it is extended into a full 3 degrees of freedom algorithm to include sway speed as well.

Because our main focus for the controller is pose navigation during lower velocities, it is possible to simplify the design procedure. The dynamic window algorithm and control system in this thesis is therefore based on a linearized dynamic positioning (DP) model, which means that the the nonlinear parts of the damping matrix $\boldsymbol{D}$ and the Coriolis and centripetal matrix $\boldsymbol{C}$ are neglected, since in low-speed manoeuvres the linear damping will dominate over both the nonlinear damping and the Coriolis and centripetal forces. Because of this, the model will only be valid for lower velocities, and will otherwise be underactuated outside of the valid area of the controller. The feasible velocities in which the controller can operate has to be determined through simulations where the linearized model is compared to the full ship model.

This gives the following 3 DOF dynamics [1]:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{3.51}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{D}_L\boldsymbol{\nu} = \boldsymbol{\tau} \tag{3.52}$$

where $\boldsymbol{D}_L$ is given by (2.13).

A 2 DOF controller is usually applied to control surge speed and yaw rate of a vessel which is to operate with higher velocities, while a 3 DOF controller is commonly used for low speed manoeuvring, such as in DP operations. Although the main focus of this thesis is to control vessels in 3 DOF during lower velocities, the 2 DOF DW algorithm and its DWC is presented here because it is crucial to fully understand the implementation of the full 3 DOF DWC. Here, the 2 DOF DWC is based on the linearized ship model in order to make the extension into 3 DOF more understandable for the reader. If the reader wishes to implement the 2 DOF DWC, the presented design should be extended to include the full nonlinear ship model to increase performance.

### 3.2.1 A 2 DOF dynamic window algorithm

Based on the linearized ship model and the actuator magnitude constraints, a set of possible steady-state velocities can be found. This set contains all the velocities the ship can achieve with respect to the actuator magnitude constraints. The possible velocities can be found by computing the steady-state solution of the kinetics (3.52) for all possible control inputs:

$$\boldsymbol{\tau}(\boldsymbol{u}) = \boldsymbol{D}_L\boldsymbol{\nu}_{ss} \tag{3.53}$$

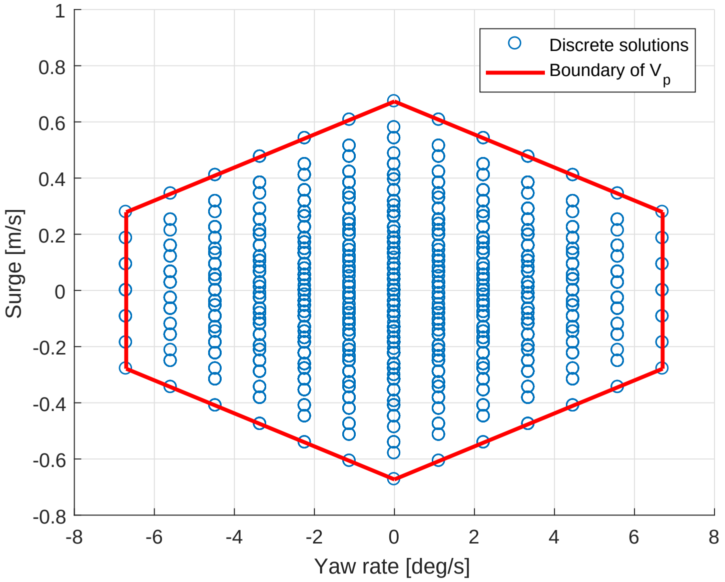


Figure 3.3: Possible combinations of surge speed and yaw rate, with respect to actuator magnitude limits, for the linearized ship model. The boundaries of $V_p$ are shown as the red line.

within the actuator magnitude constraints

$$u_i \in [-0.5, 0.5]. \tag{3.54}$$

The steady-state solutions of (3.53) for a uniformly distributed set of control inputs are shown in Figure 3.3. By designing an approximation of the boundaries, the set of possible velocities can be defined as:

$$V_p = \{(u, r) \in \mathbb{R} \times \mathbb{R} \mid g(u, r) \geq 0\}, \tag{3.55}$$

where $g(u, r)$ is greater than or equal to zero for valid solutions of (3.53)-(3.54), and negative otherwise. Given $m$ approximated boundaries, defined by the functions $h_a(u, r) = 0$, $a \in \{1, 2, ..., m\}$, which are defined by using regression on the boundary lines of $V_p$, where $\nabla h_a(u, r)$ is required to be pointing inwards to the valid solutions, the approximated $g(u, r)$ is given as:

$$g(u, r) = min(h_1(u, r), h_2(u, r), ..., h_m(u, r)). \tag{3.56}$$

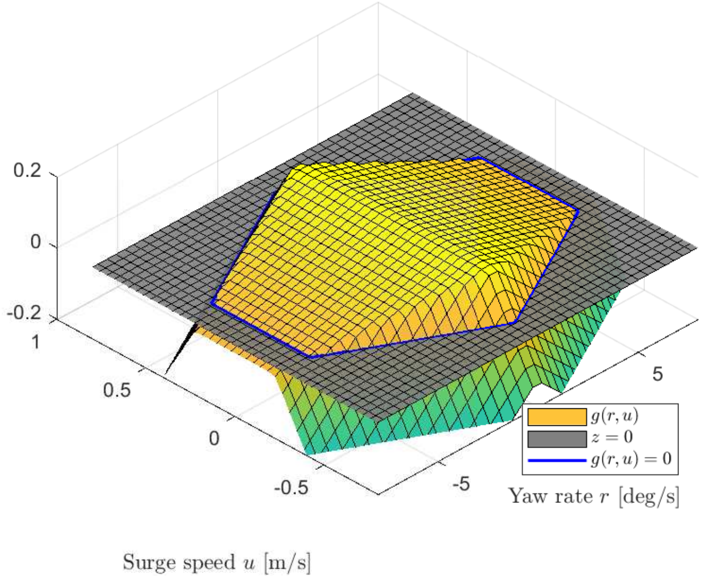


Figure 3.4: Function to find possible velocities for surge speed and yaw rate.

In Figure 3.4, a plot of the function $g(u, r)$ is shown.

Next, the space of reachable points within one time step $T$ needs to be defined. This is done by finding the acceleration limits, and based on these, the set of reachable velocities can be computed. The possible ship acceleration limits can be found by evaluating

$$\dot{\boldsymbol{\nu}} = \boldsymbol{M}^{-1}(\boldsymbol{\tau}(\boldsymbol{u}) - \boldsymbol{D}_L \boldsymbol{\nu}^*), \tag{3.57}$$

for the current velocity $\boldsymbol{\nu}^* = \boldsymbol{\nu} = [u, r]$ and the boundaries of the control input vector. The acceleration limits at the current time step can be computed as:

$$\dot{\boldsymbol{\nu}}_{min} = [\dot{u}_{min}, \dot{r}_{min}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{min}(\boldsymbol{u}) - \boldsymbol{D}_L \boldsymbol{\nu}^*) \tag{3.58}$$

$$\dot{\boldsymbol{\nu}}_{max} = [\dot{u}_{max}, \dot{r}_{max}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{max}(\boldsymbol{u}) - \boldsymbol{D}_L \boldsymbol{\nu}^*), \tag{3.59}$$

where $\boldsymbol{\tau}_{min}$ and $\boldsymbol{\tau}_{max}$ are the minimum and maximum possible control inputs, respectively. The minimum and maximum achievable velocities within the current time step can then be computed as:

$$\boldsymbol{\nu}_{min} = \boldsymbol{\nu} + T\dot{\boldsymbol{\nu}}_{min} \tag{3.60}$$

$$\boldsymbol{\nu}_{max} = \boldsymbol{\nu} + T\dot{\boldsymbol{\nu}}_{max}. \tag{3.61}$$

It should be noted that this method does not consider actuator rate saturations. However, by introducing dynamics to the control input vector $\boldsymbol{\tau}$, the algorithm can also be further developed to handle rate constraints.

Using $T$ as the time allowed for acceleration during the next time step, the dynamic velocity window is then defined using the acceleration limits from (3.58) and (3.59) as

$$\begin{aligned} V_w = \{(u, r) \in \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ \wedge\, r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\}. \end{aligned} \tag{3.62}$$

The set of dynamically feasible velocities is then defined as

$$V_f \triangleq V_p \cap V_w. \tag{3.63}$$

Next, the set of dynamically feasible velocities $V_f$ is discretized uniformly to obtain a discrete set of dynamically feasible velocity pairs.

The desired velocity is defined as

$$\boldsymbol{\nu}_{1d} \triangleq [u_d, r_d]^\top, \tag{3.64}$$

since the focus is on controlling the surge speed and yaw rate. The optimal velocity pair $\boldsymbol{\nu}_{1f} = [u_f, r_f]^\top$ can be selected as

$$\boldsymbol{\nu}_{1f} = \underset{(u,r)\in V_f}{\arg\max}\, G(u, u_d, r, r_d), \tag{3.65}$$

where $G(u, u_d, r, r_d)$ is an objective function, which is defined as

$$G(u, u_d, r, r_d) \triangleq \text{surge}(u, u_d) + \text{yawrate}(r, r_d), \tag{3.66}$$

with

$$\text{surge}(u, u_d) = 1 - \frac{|u_d - u|}{\max\limits_{u' \in V_f}(|u_d - u'|)} \in [0, 1] \tag{3.67}$$

$$\text{yawrate}(r, r_d) = 1 - \frac{|r_d - r|}{\max\limits_{r' \in V_f}(|r_d - r'|)} \in [0, 1]. \tag{3.68}$$

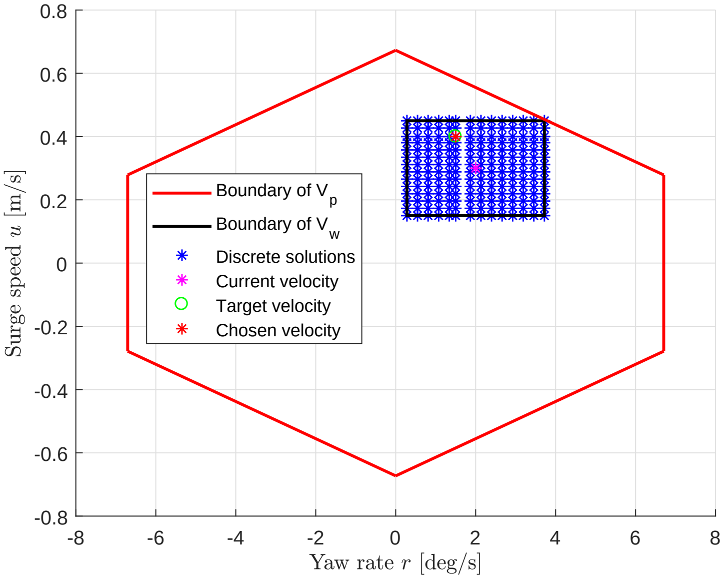


Figure 3.5: The dynamically feasible velocity set, surrounded by the boundaries of the dynamic velocity window and the possible velocity set.

Notice that by using this objective function, we minimize the scaled 1-norm of the entire discrete set of dynamically velocity pairs. In [3], [4] and [28], a distance function and tuning parameters are used to achieve collision avoidance, but this function is removed here since we only focus on handling actuator constraints. As a result, the tuning parameters also become redundant since the remaining two functions are orthogonal to each other. Figure 3.5 illustrates $V_p$, $V_w$, $V_f$ and $\boldsymbol{\nu}_{1d} = [0.4\,\mathrm{m/s}, 0.0262\,\mathrm{rad/s}]^\top$ given a current velocity pair of $\boldsymbol{\nu} = [0.3\,\mathrm{m/s}, 0.0349\,\mathrm{rad/s}]^\top$. Note that this figure is not to scale, and is used for illustrative purposes only. In reality, the window is much smaller due to the acceleration limits per time step $T$.

### 3.2.2 A 2 DOF dynamic window-based controller

In order to use the 2 DOF dynamic window algorithm presented above as a part of a control system, several changes has to be done to the controller presented in Section 3.1. This control system will not be used in either simulations or laboratory experiments, but will be presented here for the purpose of widening the reader's understanding on how a dynamic window-based controller is designed.

Since the focus is on controlling the surge speed and yaw rate, the control objective is modified to $\tilde{\psi} \triangleq \psi(t) - \psi_t(t) \to 0$ as $t \to \infty$ and $\tilde{u} \triangleq u(t) - u_t(t) \to 0$ as $t \to \infty$, where $\psi_t(t) \in \mathbb{S}$ represents the heading associated with a target ship and $u_t$ is the target surge speed. Furthermore, $\psi_t(t)$ is $\mathcal{C}^2$ and bounded.

Using a combination of a cascaded feedback controller, as presented in Section 3.1, and a dynamic feedback controller where the dynamics of the uncontrolled sway mode enters the yaw control law [18], the control input can be chosen as

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{D}_L\boldsymbol{\alpha} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2. \tag{3.69}$$

Here the NP-NV controller from Section 3.1.3 which is modified to be used on the linearized shop model is chosen.

Now, the error variables $z_1$ and $\boldsymbol{z}_2 = [z_{2,u}, z_{2,v}, z_{2,r}]^\top$ are defined as

$$z_1 \triangleq \psi - \psi_t \tag{3.70}$$

$$\boldsymbol{z}_2 \triangleq \boldsymbol{\nu} - \boldsymbol{\alpha}, \tag{3.71}$$

where, as before, $\boldsymbol{\alpha} = [\alpha_u, \alpha_v, \alpha_r] \in \mathbb{R}^3$ is a vector of stabilising functions, which can be interpreted as a desired velocity, such that

$$\alpha_u = u_t \tag{3.72}$$

$$\alpha_r = \dot{\psi}_t - K_1(z_1)z_1, \tag{3.73}$$

where

$$K_1(z_1) = \Gamma_1 \frac{1}{\sqrt{z_1^2 + \Delta_{\tilde{\psi}}^2}} \tag{3.74}$$

represents a nonlinear control gain with $\Gamma_1 > 0$ and $\Delta_{\tilde{\psi}} > 0$. The nonlinear feedback term in (3.69) is, as before, given as

$$\boldsymbol{K}_2(\boldsymbol{z}_2) = \boldsymbol{\Gamma}_2 \begin{bmatrix} \frac{1}{\sqrt{\boldsymbol{z}_{2,\tilde{v}}^\top \boldsymbol{z}_{2,\tilde{v}} + \Delta_{\tilde{v}}^2}} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \frac{1}{\sqrt{z_{2,\tilde{r}}^2 + \Delta_{\tilde{r}^2}}} \end{bmatrix}, \tag{3.75}$$

with the control gain $\boldsymbol{\Gamma}_2 > 0$, where $\boldsymbol{z}_{2,\tilde{v}}$ is defined as $\boldsymbol{z}_{2,\tilde{v}} \triangleq [z_{2,u}, z_{2,v}]^\top$, $\Delta_{\tilde{v}} > 0$ and $\Delta_{\tilde{r}} > 0$. The time derivative of the vector of stabilising functions then becomes

$$\dot{\boldsymbol{\alpha}} = [\dot{u_t}, \dot{\alpha_v}, \dot{\alpha_r}]^\top, \tag{3.76}$$

where $\dot{u_t}$ is the target surge acceleration, and

$$\dot{\alpha_r} = \ddot{\psi_t} - \dot{K}_1(z_1)z_1 - K_1(z_1)\dot{z}_1, \tag{3.77}$$

with

$$\dot{z}_1 = -K_1(z_1)z_1 + z_{2,\tilde{r}} \tag{3.78}$$

and

$$\dot{K}_1(z_1) = \Gamma_1 \frac{z_1\dot{z}_1}{\left(z_1^2 + \Delta_{\tilde{\psi}}^2\right)^{\frac{3}{2}}}. \tag{3.79}$$

Based on the design of the dynamics of the uncontrolled sway mode in [18], the variable $\alpha_v$ is a dynamic state of the controller, and is given by

$$m_{22}\dot{\alpha}_v = -d_{22}(\boldsymbol{\nu})\alpha_v + \gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2), \tag{3.80}$$

where

$$\gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2) = K_{2,22}(\boldsymbol{z}_2)z_{2,v} - m_{23}\dot{\alpha}_r - d_{23}(\boldsymbol{\nu})\alpha_r, \tag{3.81}$$

and $m_{ij}$, $d_{ij}(\boldsymbol{\nu})$ and $K_{2,ij}(\boldsymbol{z}_2)$ are components at the $i$th row and $j$th column of the matrices $\boldsymbol{M}$, $\boldsymbol{D}_L$ and $\boldsymbol{K}_2(\boldsymbol{z}_2)$, while

$$\alpha_v = \int_0^t \dot{\alpha}_v(\sigma)d\sigma + \alpha_v(0), \qquad \alpha_v(0) = v(0). \tag{3.82}$$

Combining this with the simplified 2 DOF dynamic window algorithm, we can develop a dynamic window-based controller. In this setup, the simplified dynamic window algorithm will use $\boldsymbol{\alpha}_1 = [\alpha_u, \alpha_r]^\top$ as an input such that $\boldsymbol{\nu}_{1d} = \boldsymbol{\alpha}_1$. In the case where $\boldsymbol{\alpha}_1$ is an infeasible velocity, the algorithm will modify $\boldsymbol{\alpha}_1$ to a feasible velocity $\boldsymbol{\alpha}_{1f} = [\alpha_{f,u}, \alpha_{f,r}]^\top$, otherwise $\boldsymbol{\alpha}_{1f} = \boldsymbol{\alpha}$. A pseudocode of the simplified DW algorithm is shown in Algorithm 1.

After the optimal velocity pair $\boldsymbol{\alpha}_{1f}$ is found, the vector of stabilising functions is given as

$$\boldsymbol{\alpha}_f = [\alpha_{f,u}, \alpha_{f,v}, \alpha_{f,r}]^\top, \tag{3.83}$$

where $\alpha_{f,v}$ is, as in (3.82), given as

$$\alpha_{f,v} = \int_0^t \dot{\alpha}_{f,v}(\sigma)d\sigma + \alpha_{f,v}(0), \qquad \alpha_{f,v}(0) = v(0), \tag{3.84}$$

**Algorithm 1** Pseudocode of the simplified DW algorithm

```
$V_w$ is calculated using (3.58),(3.59) and (3.62) and discretized uniformly
if the desired velocity vector $\boldsymbol{\alpha}_1 \in V_f$ then
    The closest reachable velocity row and column to $\boldsymbol{\alpha}_1$ is shifted such that $\boldsymbol{\alpha}_1$ is one of the reachable velocity pairs in $V_w$
end if
Remove all the reachable velocity pairs in $V_w$ which are outside of the $g(u, r)$ boundaries to describe the set of dynamically feasible velocities $V_f$
Select the optimal velocity pair $\boldsymbol{\alpha}_{1f}$ through maximizing the objective function (3.66) over the discrete feasible search space $V_f = V_p \cap V_w$
```

where, as in (3.80),

$$m_{22}\dot{\alpha}_{f,v} = -d_{22}(\boldsymbol{\nu})\alpha_{f,v} + \gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2), \tag{3.85}$$

with, based on (3.81),

$$\gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2) = -m_{23}\dot{\alpha}_r - d_{23}(\boldsymbol{\nu})\alpha_r. \tag{3.86}$$

We want the ship to reach $\boldsymbol{\alpha}_f$ after time $T$, hence the desired acceleration is chosen to be

$$\dot{\boldsymbol{\alpha}}_{DWC} = \frac{\boldsymbol{\alpha}_f - \boldsymbol{\nu}}{T}, \tag{3.87}$$

which means that

$$\boldsymbol{\alpha}_{DWC} = \int_0^t \dot{\boldsymbol{\alpha}}_{DWC} d\sigma + \boldsymbol{\alpha}_{DWC}(0). \tag{3.88}$$

Both $\boldsymbol{\alpha}_{DWC}$ and $\dot{\boldsymbol{\alpha}}_{DWC}$ are used in the kinetic controller which is modified to

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC}, \tag{3.89}$$

for the case of the linearized ship model.

The DWC uses the heading controller given in (3.73) together with the target speed $u_t$ as inputs to the simplified DW algorithm, which is described in Algorithm 1, in order to determine the vector of stabilizing functions given in (3.83)-(3.86). Based on (3.83), the desired acceleration and velocity vectors are found using (3.87)-(3.88), which are used to construct the control input (3.89).

When comparing the control law in (3.89) against (3.69), it can be seen that the feedback term $-\boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2$ in (3.69) is not included in (3.89) since the DWC makes the optimal velocity pair track the target velocity by using (3.87)-(3.88). For a more detailed discussing regarding this topic, see the end of Section 3.2.4.

**Stability analysis**

The dynamic feedback controller is proven stable in [18]. There, through the use of Lyapunov augments, the equilibrium $(z_1, \boldsymbol{z}_2) = (0, \mathbf{0})$ of the $\boldsymbol{z}$-subsystem is proved UGAS. The unforced $\alpha_v$-subsystem ($\gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2) = 0$) is clearly exponentially stable, and since $(z_1, \boldsymbol{z}_2) \in \mathcal{L}_\infty$ and $(\alpha_r, \dot{\alpha}_r) \in \mathcal{L}_\infty$, then $\gamma(\alpha_r, \dot{\alpha}_r, \boldsymbol{z}_2) \in \mathcal{L}_\infty$. This then implies that the $\alpha_v$-subsystem is input-to-state stable from $\gamma(\cdot)$ to $\alpha_v$. This is proven in greater detail in [18].

To achieve a stability proof of the dynamic window algorithm, and then thus the full 2 DOF dynamic window-based controller, is rather difficult and tedious work. In [17], a dynamic window algorithm used for horizontal collision avoidance is proved stable based on the work done in [31]. There, the controller is found to be UGAS and ULES in an obstacle-free environment, and, based on this result and cascade theory, such as in [11] and [21], it is then possible to conclude that the 2 DOF dynamic window-based controller based on a dynamic feedback controller is stable and UGAS as well, as it is an algorithm based on the dynamic window algorithm where the collision avoidance part is removed, and where the dynamic feedback controller clearly is stable, as proven above.

### 3.2.3 Extending the 2 DOF DW algorithm into 3 DOF

To extend the above algorithm into 3 DOF, it is required to include sway speed. Thus, it is required to recompute the steady-state kinetics in (3.52) with the included sway motion. Simulating (3.53) with sway speed included, will yield the steady-state velocities needed to create the dynamic window.

Based on the linearized ship model and its actuator constraints, a set of possible velocities can be found. This set contains all the velocities the ship can achieve with respect to the actuator magnitude constraints. The steady-state solutions of (3.53) for a uniformly distributed set of control inputs, are shown in Figure 3.6. By designing an approximation of the boundaries, the set of possible velocities can be defined as

$$V_p = \{(u, v, r) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R} \mid g(u, v, r) \geq 0\}, \tag{3.90}$$

where $g(u, v, r)$ is a positive semidefinite function for feasible velocities with respect to the actuator constraints.

However, it is possible to split the 3 DOF problem into three 2 DOF problems. Here, an approximation of the 3 DOF set is done by projecting the set into three 2 DOF sets to simplify calculations [6]. Justifying the approximation

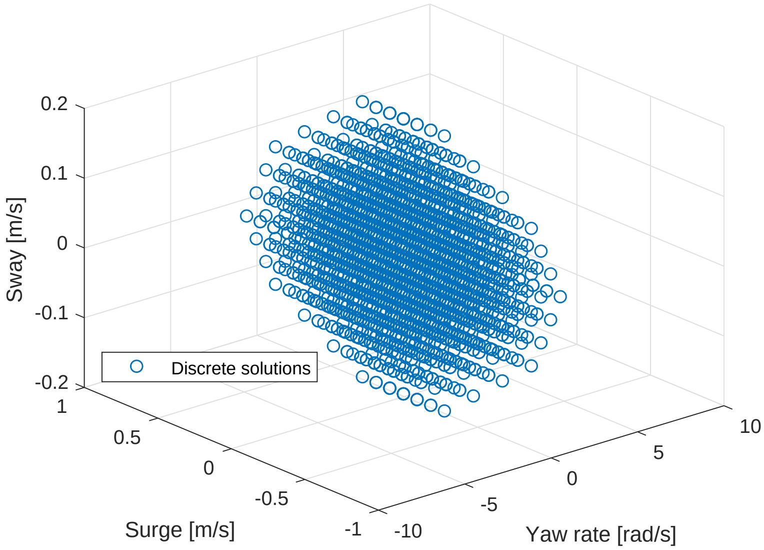


Figure 3.6: Steady-state solutions of (3.53) for a uniformly distributed set of control inputs.

by noting that each of the steady-state solution boundary faces are almost parallel with one axis. Following this, faces that are parallel with one axis can be parametrized by the remaining two variables. Notice, however, that we lose information where all three variables are correlated, and can therefore not model faces which are not parallel with one of the axis. The result of the approximation is the following three sets of possible velocities:

$$V_{p,(u,r)} = \{(u, r) \in \mathbb{R} \times \mathbb{R} \mid g_{(u,r)}(u, r) \geq 0\} \tag{3.91}$$

$$V_{p,(v,r)} = \{(v, r) \in \mathbb{R} \times \mathbb{R} \mid g_{(v,r)}(v, r) \geq 0\} \tag{3.92}$$

$$V_{p,(u,v)} = \{(u, v) \in \mathbb{R} \times \mathbb{R} \mid g_{(u,v)}(u, v) \geq 0\}, \tag{3.93}$$

where $g_{(u,r)}(u, r)$, $g_{(v,r)}(v, r)$ and $g_{(u,v)}(u, v)$ are greater than or equal to zero for velocities inside the corresponding boundaries. The three sets are shown in Figure 3.7, 3.8 and 3.9, where the discrete solutions show the steady-state solutions for the respective 2 DOF window, and the red lines show the boundaries

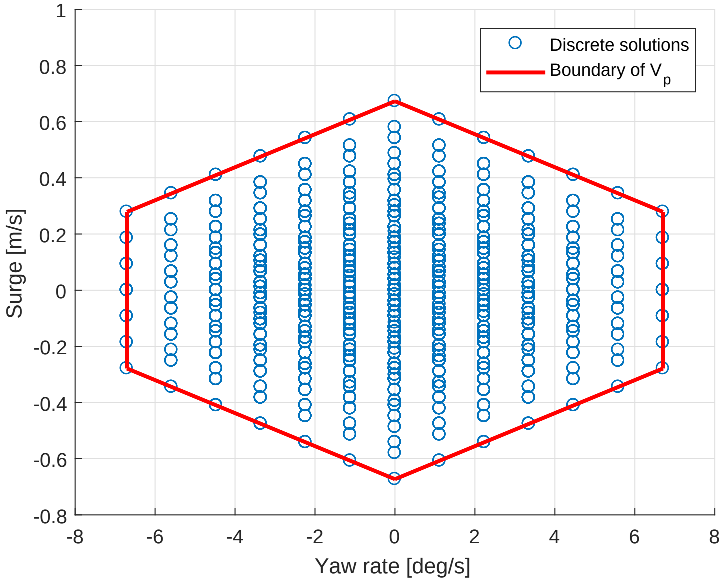


Figure 3.7: Possible combinations of surge speed and yaw rate, with respect to actuator magnitude limits, for the linearized ship model. The boundaries of $V_p$ are shown as the red line.

of the respective set of possible velocities. An alternative to this method, which possibly would result in a more correct 3 DOF set of possible velocities, would be to compute the set analytically. However, this would be more challenging and time consuming, and would be most beneficial when using the full nonlinear model together with the dynamic window algorithm.

Given $m$, $n$ and $k$ approximated boundaries, defined by the functions $h_{a,(u,r)}(u,r) = h_{b,(v,r)}(v,r) = h_{c,(u,v)}(u,v) = 0$, $a \in \{1,2,...,m\}$, $b \in \{1,2,...,n\}$ and $c \in \{1,2,...,k\}$, the approximated functions are given as:

$$g_{(u,r)}(u,r) = min(h_{1,(u,r)}(u,r), h_{2,(u,r)}(u,r), ..., h_{m,(u,r)}(u,r)) \tag{3.94}$$
$$g_{(v,r)}(v,r) = min(h_{1,(v,r)}(v,r), h_{2,(v,r)}(v,r), ..., h_{n,(v,r)}(v,r)) \tag{3.95}$$
$$g_{(u,v)}(u,v) = min(h_{1,(u,v)}(u,v), h_{2,(u,v)}(u,v), ..., h_{k,(u,v)}(u,v)). \tag{3.96}$$

Here, the functions $h_{a,(u,r)}(u,r) = h_{b,(v,r)}(v,r) = h_{c,(u,v)}(u,v) = 0$ are defined

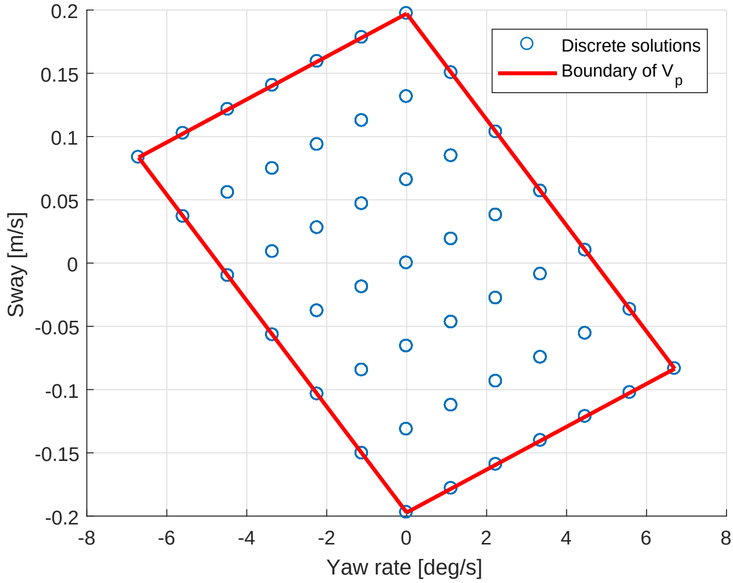


Figure 3.8: Possible combinations of sway speed and yaw rate, with respect to actuator magnitude limits, for the linearized ship model. The boundaries of $V_p$ are shown as the red line.

by using regression on the boundary of the sets $V_{p_{(u,r)}}$, $V_{p_{(v,r)}}$ and $V_{p_{(u,v)}}$, where $\nabla h_{a,(u,r)}(u,r)$, $\nabla h_{b,(v,r)}(v,r)$ and $\nabla h_{c,(u,v)}(u,v)$ are required to be pointing inwards to the valid solutions. In Figure 3.10, 3.11 and 3.12, plots for the functions (3.94)-(3.96) are shown.

Next, the space of reachable points within one time step $T$ needs to be defined. Using

$$\dot{\boldsymbol{\nu}}_{min} = [\dot{u}_{min}, \dot{v}_{min}, \dot{r}_{min}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{min}(\boldsymbol{u}) - \boldsymbol{D}_L\boldsymbol{\nu}^*) \tag{3.97}$$

$$\dot{\boldsymbol{\nu}}_{max} = [\dot{u}_{max}, \dot{v}_{max}, \dot{r}_{max}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{max}(\boldsymbol{u}) - \boldsymbol{D}_L\boldsymbol{\nu}^*), \tag{3.98}$$

where $\boldsymbol{\tau}_{min}$ and $\boldsymbol{\tau}_{max}$ are the minimum and maximum possible control inputs, respectively, and $\boldsymbol{\nu}^*$ is the current velocity $\boldsymbol{\nu}$, we find the acceleration limits and the reachable velocities for the current time step. The minimum and maximum

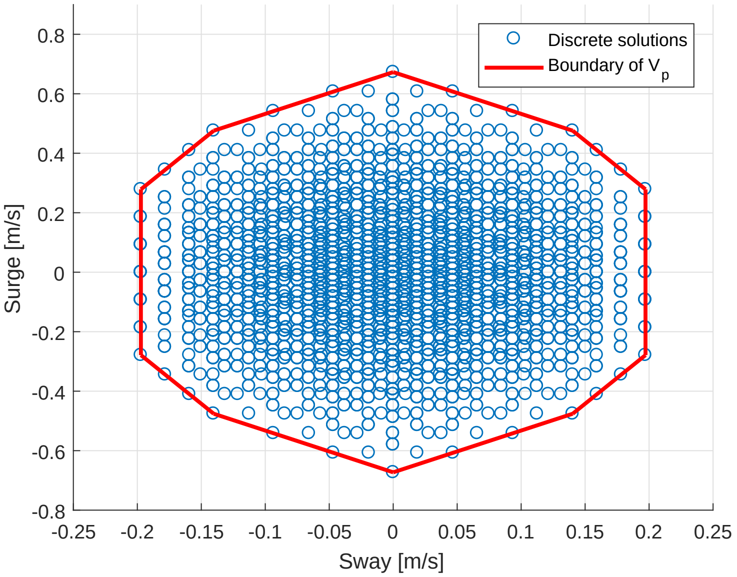


Figure 3.9: Possible combinations of surge and sway speed, with respect to actuator magnitude limits, for the linearized ship model. The boundaries of $V_p$ are shown as the red line.

achievable velocities within the current time step can then be computed as:

$$\boldsymbol{\nu}_{min} = \boldsymbol{\nu} + T\dot{\boldsymbol{\nu}}_{min} \tag{3.99}$$

$$\boldsymbol{\nu}_{max} = \boldsymbol{\nu} + T\dot{\boldsymbol{\nu}}_{max}. \tag{3.100}$$

Using $T$ as the time allowed for acceleration during the next time step, the dynamic velocity window is then defined using the acceleration limits from (3.97) and (3.98) as

$$\begin{aligned} V_w = \{(u, v, r) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ \wedge\, v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T] \wedge r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\}, \end{aligned} \tag{3.101}$$

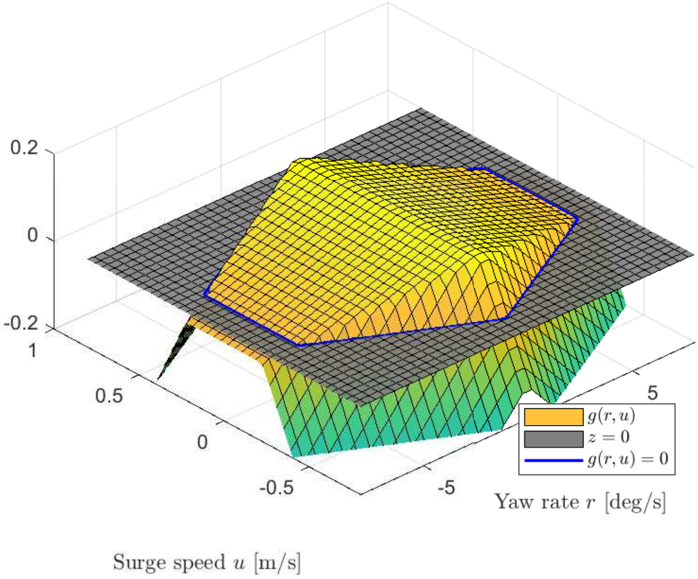


Figure 3.10: Function to find possible velocities for surge and yaw rate.

which we project into the three cases

$$V_{w,(u,r)} = \{(u,r) \in \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ \wedge\, r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\} \tag{3.102}$$

$$V_{w,(v,r)} = \{(v,r) \in \mathbb{R} \times \mathbb{R} \mid v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T] \\ \wedge\, r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\} \tag{3.103}$$

$$V_{w,(u,v)} = \{(u,v) \in \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ \wedge\, v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T]\}. \tag{3.104}$$

This defines the sets of dynamically feasible velocities as

$$V_{f,(u,r)} \triangleq V_{p,(u,r)} \cap V_{w,(u,r)} \tag{3.105}$$

$$V_{f,(v,r)} \triangleq V_{p,(v,r)} \cap V_{w,(v,r)} \tag{3.106}$$

$$V_{f,(u,v)} \triangleq V_{p,(u,v)} \cap V_{w,(u,v)}, \tag{3.107}$$

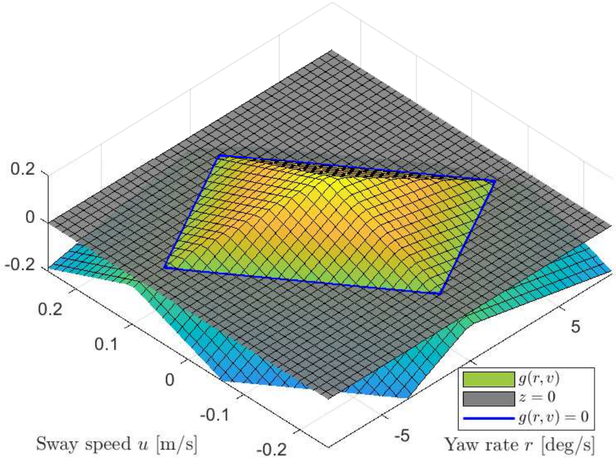


Figure 3.11: Function to find possible velocities for sway and yaw rate.

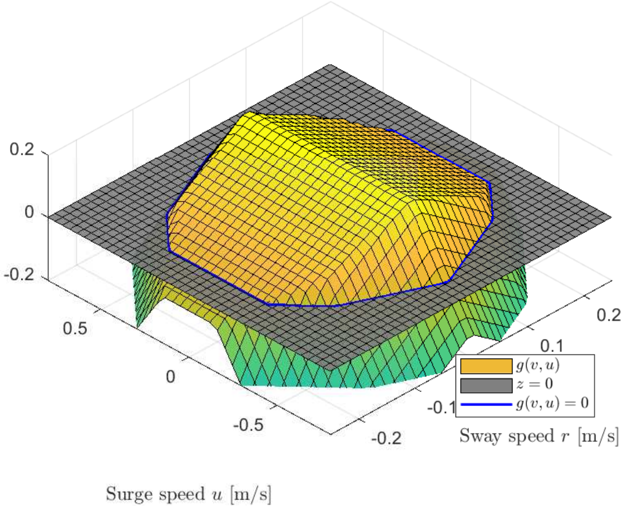


Figure 3.12: Function to find possible velocities for surge and sway.

which for the whole window can be expressed as

$$V_f \triangleq \left(V_{p,(u,r)} \cup V_{p,(v,r)} \cup V_{p,(u,v)}\right) \cap \left(V_{w,(u,r)} \cup V_{w,(v,r)} \cup V_{w,(u,v)}\right). \tag{3.108}$$

Next, the sets of dynamically feasible velocities are discretized uniformly to obtain discrete sets of dynamically feasible velocities. For the 3 DOF case, the desired velocity is defined as

$$\boldsymbol{\nu}_d \triangleq [u_d, v_d, r_d]^\top. \tag{3.109}$$

Given $\boldsymbol{\nu}_d$, the optimal dynamically feasible velocity $\boldsymbol{\nu}_f = [u_f, v_f, r_f]^\top$ can be selected as

$$\boldsymbol{\nu}_f = \underset{(u,v,r)\in V_f}{\arg\max}\ G(\boldsymbol{\nu}, \boldsymbol{\nu}_d), \tag{3.110}$$

where $V_f$ is the general 3 DOF solution and $G(\boldsymbol{\nu}, \boldsymbol{\nu}_d)$ is an objective function, which is defined as

$$G(\boldsymbol{\nu}, \boldsymbol{\nu}_d) \triangleq \mathrm{surge}(u, u_d) + \mathrm{sway}(v, v_d) + \mathrm{yawrate}(r, r_d), \tag{3.111}$$

with

$$\mathrm{surge}(u, u_d) = 1 - \frac{|u_d - u|}{\max\limits_{u'\in V_f}(|u_d - u'|)} \in [0,1] \tag{3.112}$$

$$\mathrm{sway}(v, v_d) = 1 - \frac{|v_d - u|}{\max\limits_{v'\in V_f}(|v_d - v'|)} \in [0,1] \tag{3.113}$$

$$\mathrm{yawrate}(r, r_d) = 1 - \frac{|r_d - r|}{\max\limits_{r'\in V_f}(|r_d - r'|)} \in [0,1]. \tag{3.114}$$

Notice that by using this objective function, we minimise the scaled 1-norm of the entire discrete set of dynamically feasible velocities. For the three 2 DOF case, this algorithm is modified to fit 2 DOF and run once for each velocity pair scenario; surge speed and yaw rate, sway speed and yaw rate, and surge and sway speed. Resulting in the sets of dynamically feasible velocities

$$\boldsymbol{\nu}_{f,(u,r)} = [\nu_{f,u}, 0, \nu_{f,r}]^\top \tag{3.115}$$

$$\boldsymbol{\nu}_{f,(v,r)} = [0, \nu_{f,v}, \nu_{f,r}]^\top \tag{3.116}$$

$$\boldsymbol{\nu}_{f,(u,r)} = [\nu_{f,u}, \nu_{f,v}, 0]^\top, \tag{3.117}$$

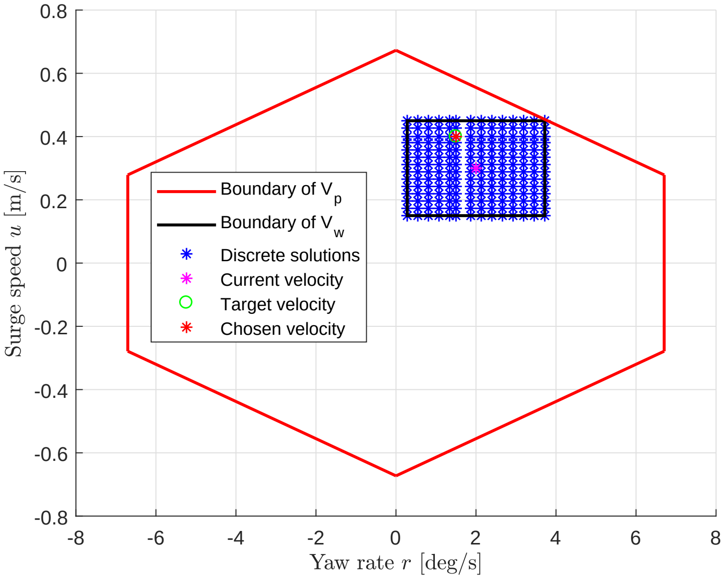


Figure 3.13: The dynamically feasible velocity set for surge speed and yaw rate, surrounded by the boundaries of the dynamic velocity window and the possible velocity set.

which combines into

$$\boldsymbol{\nu}_f = \frac{\boldsymbol{\nu}_{f,(u,r)} + \boldsymbol{\nu}_{f,(v,r)} + \boldsymbol{\nu}_{f,(u,v)}}{2} \tag{3.118}$$

for the full 3 DOF case.

Figures 3.13, 3.14 and 3.15 illustrate the sets $V_p$, $V_w$ and $V_f$ for the three 2 DOF cases, where $\boldsymbol{\nu}_d = [0.4\,\text{m/s}, 0.13\,\text{m/s}, 0.0262\,\text{rad/s}]^\top$ given a current velocity $\boldsymbol{\nu} = [0.3\,\text{m/s}, 0.1\,\text{m/s}, 0.0349\,\text{rad/s}]^\top$. Note that this is not to scale, and is used for illustrative purposes only. In reality, the window is smaller due to the acceleration limits per time step $T$.

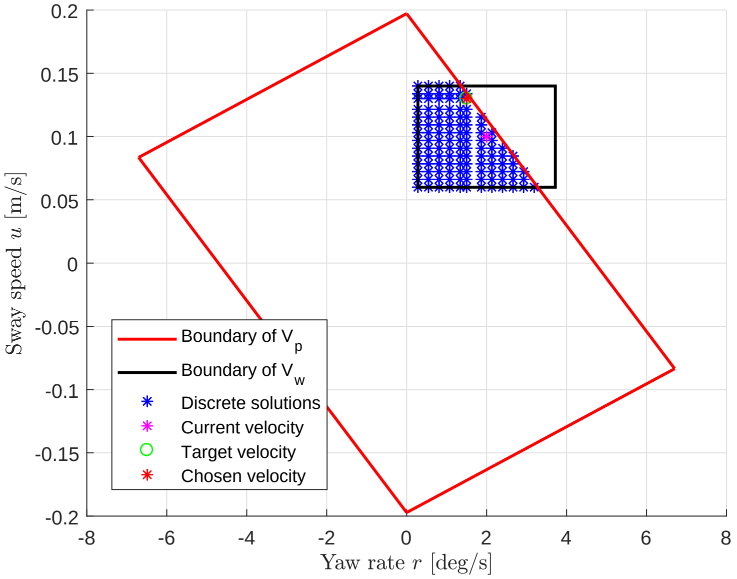


Figure 3.14: The dynamically feasible velocity set for sway speed and yaw rate, surrounded by the boundaries of the dynamic velocity window and the possible velocity set.

### 3.2.4 A 3 DOF dynamic window-based controller

In order to use the above result as a dynamic window-based controller, we combine the previous traditional control design with the simplified dynamic window algorithm.

In this setup, the simplified DW algorithm will use $\boldsymbol{\alpha} = [\alpha_u, \alpha_v, \alpha_r]^\top$ as an input such that $\boldsymbol{\nu}_d = \boldsymbol{\alpha}$. In the case where $\boldsymbol{\alpha}$ is an infeasible velocity, the simplified DW algorithm will modify $\boldsymbol{\alpha}$ to a feasible velocity $\boldsymbol{\alpha}_f = [\alpha_{f,u}, \alpha_{f,v}, \alpha_{f,r}]^\top$, otherwise $\boldsymbol{\alpha}_f = \boldsymbol{\alpha}$. A pseudo-code of the simplified DW algorithm is shown in Algorithm 2.

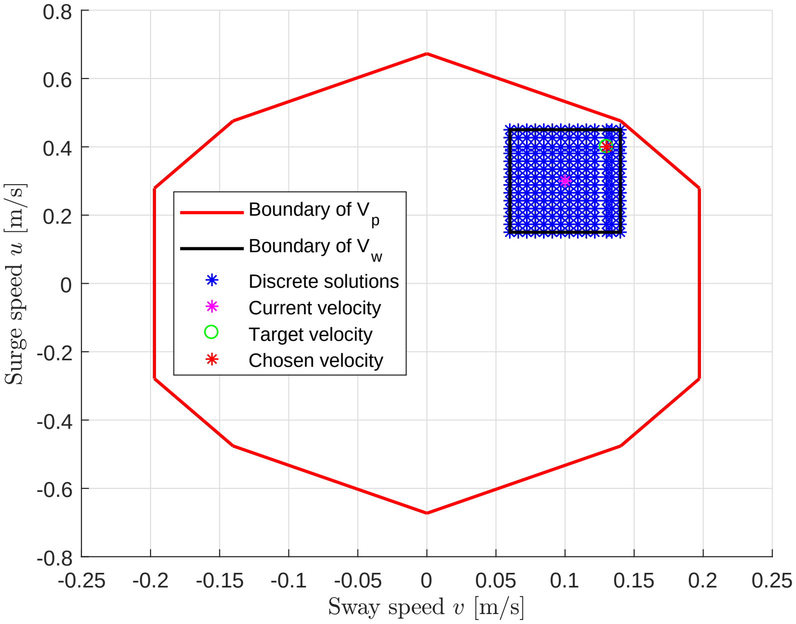


Figure 3.15: The dynamically feasible velocity set for surge and sway speed, surrounded by the boundaries of the dynamic velocity window and the possible velocity set.

**Algorithm 2** Pseudocode of the simplified DW algorithm for 3 DOF

```
$V_w$ is calculated using (3.97),(3.98) and (3.101) and discretized uniformly
if the desired velocity vector $\boldsymbol{\alpha} \in V_f$ then
    The closest reachable velocity row and column to $\boldsymbol{\alpha}$ is shifted such that $\boldsymbol{\alpha}$ is one of the reachable velocities in $V_w$
end if
Remove all the reachable velocities in $V_w$ which are outside of the $g(u, v, r)$ boundaries to describe the set of dynamically feasible velocities $V_f$
Select the optimal velocity pair $\boldsymbol{\alpha}_f$ through maximizing the objective function (3.111) over the discrete feasible search space $V_f = V_p \cap V_w$
```

For the three 2 DOF case, this algorithm is modified to fit 2 DOF and run once for each velocity pair scenario; surge speed and yaw rate, sway speed and yaw rate, and surge and sway speed. Resulting in the vectors of stabilising functions:

$$\boldsymbol{\alpha}_{f,(u,r)} = [\alpha_{f,u}, 0, \alpha_{f,r}]^\top \tag{3.119}$$

$$\boldsymbol{\alpha}_{f,(v,r)} = [0, \alpha_{f,v}, \alpha_{f,r}]^\top \tag{3.120}$$

$$\boldsymbol{\alpha}_{f,(u,v)} = [\alpha_{f,u}, \alpha_{f,v}, 0]^\top, \tag{3.121}$$

which, as in (3.118), combines into

$$\boldsymbol{\alpha}_f = \frac{\boldsymbol{\alpha}_{f,(u,r)} + \boldsymbol{\alpha}_{f,(v,r)} + \boldsymbol{\alpha}_{f,(u,v)}}{2} \tag{3.122}$$

for the full 3 DOF case.

We want the ship to reach $\boldsymbol{\alpha}_f$ after the time $T$, hence the desired acceleration is chosen to be

$$\dot{\boldsymbol{\alpha}}_{DWC} = \frac{\boldsymbol{\alpha}_f - \boldsymbol{\nu}}{T}, \tag{3.123}$$

which means that

$$\boldsymbol{\alpha}_{DWC} = \int_0^t \dot{\boldsymbol{\alpha}}_{DWC} d\sigma + \boldsymbol{\alpha}_{DWC}(0). \tag{3.124}$$

Both $\boldsymbol{\alpha}_{DWC}$ and $\dot{\boldsymbol{\alpha}}_{DWC}$ are used in the linear kinetic controller which is modified to

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC}. \tag{3.125}$$

The 3 DOF DWC uses the desired velocity given in (3.25) together with the current velocity $\boldsymbol{\nu}$ as inputs to the simplified DW algorithm, which is described in Algorithm 2, in order to determine the vector of stabilizing functions given in (3.119)-(3.122). Based on (3.122), the desired acceleration and velocity vectors are found using (3.123)-(3.124), which are used to construct the control input (3.125). The block diagram for the DWC is shown in Figure 3.16.

When comparing the control law in (3.125) against (3.38), it can be seen that the feedback term $-\boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2$ in (3.38) is not included in (3.125) since the DWC makes the optimal velocity track the target velocity by using (3.123)-(3.124). However, the DWC is a feedforward-based control algorthim which has some weaknesses against uncertainties and disturbances since the DWC is heavily model-dependent, and (3.125) can only fulfil the control objective when the DWC model is correct. In practice, when the model is not perfectly known, the control law (3.125) should also include a feedback term. This feedback term will also accommodate for internal uncertainties and external disturbances, instead of just controlling the surge speed, sway speed and yaw rate. When the system is affected by internal uncertainties and external disturbances, the DWC will attempt to compensate for them since the DWC tries to find the optimal velocity. However, the performance in terms of robustness for the DWC is limited by the actuator constraints, which give a maximum bound on the uncertainties and disturbances which the controller can compensate for. If it is necessary to compensate for model uncertainties or external disturbances, the kinetic controller can be modified to include the feedback term $-\boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2$:

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2. \tag{3.126}$$

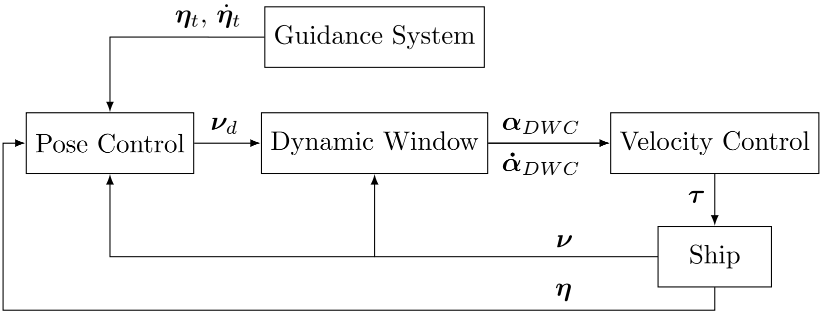


Figure 3.16: Block diagram for the dynamic window-based controller without feedback.

### Stability analysis

To achieve a stability proof of the dynamic window algorithm, and then thus the full 3 DOF dynamic window-based controller, is rather difficult and tedious work. In [17], a dynamic window algorithm used for horizontal collision avoidance is proved stable based on the work done in [31]. There, the controller is found to be UGAS and ULES in an obstacle-free environment. Combining this result with the stability analysis done in Section 3.1.3 and by using cascade theory, such as in [11] and [21], one can conclude that the 3 DOF dynamic window-based controller, with or without the feedback term, is stable and ULES, as it is an algorithm based on the dynamic window algorithm where the collision avoidance part is removed, and where the nonlinear cascaded feedback is UES, as proven in Section 3.1.3.

### 3.2.5 Model comparison

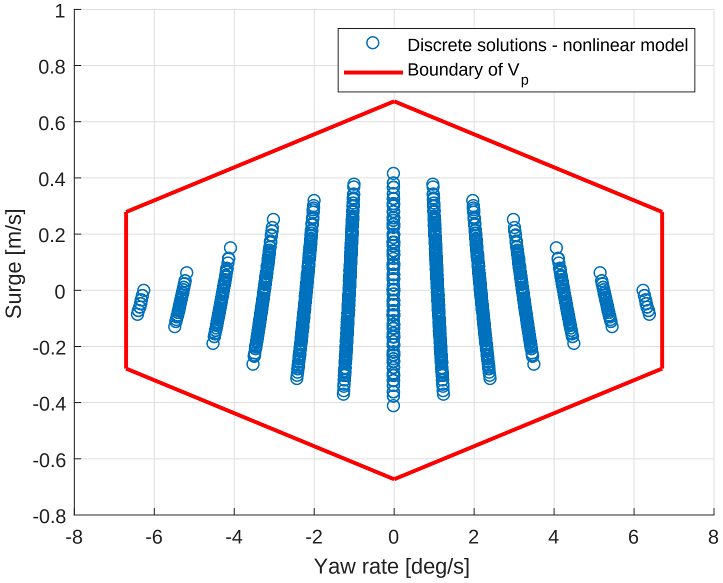


Figure 3.17: Possible combinations of surge speed and yaw rate, with respect to actuator magnitude limits, for the full nonlinear ship model. The boundaries of $V_p$ based on the linear model are shown as the red line.

To get an understanding of where the linear model is valid, a comparison between the linear and nonlinear model is needed. In Section 3.2.3, the possible velocities for the linear model were found by computing the steady-state velocities for all possible control inputs. For the nonlinear model, the calculation is modified to

$$\boldsymbol{\tau}(\boldsymbol{u}) = \boldsymbol{C}\boldsymbol{\nu}_{ss} + \boldsymbol{D}\boldsymbol{\nu}_{ss}, \tag{3.127}$$

where (3.53) is modified to include the Coriolis and centripetal matrix $\boldsymbol{C}$ from (2.7) and the full nonlinear damping matrix $\boldsymbol{D}$ in (2.12).

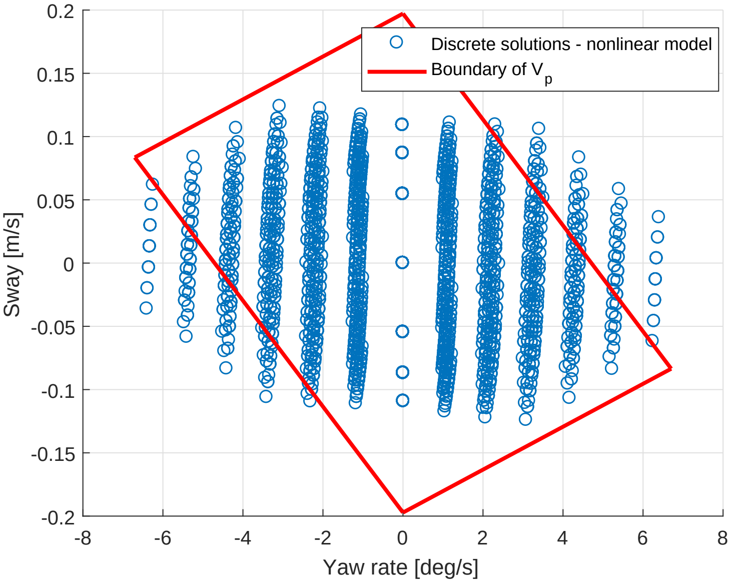


Figure 3.18: Possible combinations of sway speed and yaw rate, with respect to actuator magnitude limits, for the full nonlinear ship model. The boundaries of $V_p$ based on the linear model are shown as the red line.

The comparison between the linear and nonlinear model is shown in Figure 3.17, 3.18 and 3.19, where the discrete solutions of (3.127) are shown by the blue circular outline, and the boundaries for the set of possible velocities for the linear model are shown by the red outline.

As can be seen in Figure 3.17, including the nonlinearities in the ship model for the yaw rate-surge speed window will decrease the maximum steady-state velocities, effectively reducing the maximum velocities the ship can achieve within actuator magnitude constraints given in (3.54). This means that, for the case of the yaw rate-surge speed window, one can safely use the linear DWC, as the ship physically cannot achieve velocities that lie outside of the linear window, effectively being limited by the boundaries of the possible velocities set for the nonlinear model, which is limited by the discrete solutions as shown in Figure 3.17. The same can be seen for the surge-sway speed window in Figure 3.19.

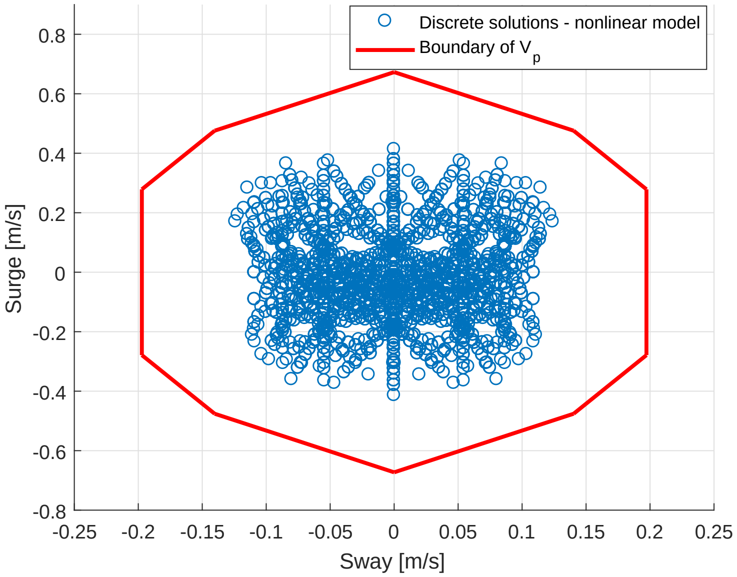


Figure 3.19: Possible combinations of surge and sway, with respect to actuator magnitude limits, for the full nonlinear ship model. The boundaries of $V_p$ based on the linear model are shown as the red line.

However, this would not lead to an energy efficient behaviour, as the controller would assume that the ship could reach higher velocities, and thus command maximum thrust without actually achieving a higher velocity, effectively wasting energy.

For the yaw rate-sway speed window shown in Figure 3.18, it is shown that the nonlinear model has discrete solutions that lie outside the boundaries for the set of possible velocities for the linear model, meaning that the actual ship can achieve greater velocities than what the linear window allows. This means that if the linear DWC commanded a maximum thrust to achieve a velocity on the border of its set of possible velocities, it would exceed the linear limitations, resulting in an empty set of feasible velocities, which potentially would end up with the algorithm choosing $\boldsymbol{\alpha}_{f,(v,r)} = 0$, effectively affecting the choice of the optimal velocity par $\boldsymbol{\alpha}_f$ in (3.122), and thus also the commanded control input.

Based on this, it can then be considered smart to use the linear DWC for what it is designed for; pose navigation during lower velocities, where one keeps the velocities well within the boundaries of the full nonlinear model. Suggested operational space for the velocities are approximately: $[-0.2, 0.2]\,(m/s)$ for surge speed, $[-0.1, 0.1]\,(m/s)$ for sway speed, and $[-3.5, 3.5]\,(deg/s)$ for yaw rate. If one wishes to violate the suggested limits, one should consider implementing a dynamic window-based controller based on the full nonlinear ship model instead.

## 3.3 Chapter summary

In this chapter, cascaded feedback and dynamic window-based controllers were discussed. First, a linear cascaded feedback controller was implemented, and then step-by-step developed into a nonlinear cascaded feedback controller. The stability of the different feedback controllers were analysed, and tuning rules were presented. In addition, it was investigated if the MRS model would impact the stability of the controllers when added to the control system in cascade. Then, a linear 2 DOF dynamic window algorithm and its correlated dynamic window-based controller based on a dynamic feedback controller, was presented. The 2 DOF dynamic window algorithm was extended into 3 DOF, and a 3 DOF dynamic window-based controller was developed. The dynamic window-based controllers were presented along with a brief stability analysis. Then, a comparison between the linear and nonlinear ship model for 3 DOF was discussed, in order to find out for which velocities the linear model would be valid.

# Chapter 4

# Simulation results

This chapter contains the results from numerical simulations of the different control algorithms, with and without constraint handling methods. The simulation details are presented, and then some performance metrics are discussed. The simulation results are split into two parts; main simulation results, where the control algorithms are simulated without external disturbances or uncertainties, and extended simulation results, where disturbances and uncertainties are introduced in order to analyse how the control schemes perform during non-optimal scenarios, like in the laboratory.

## 4.1 Simulation scenario

A so-called 4-corner test, as shown in Figure 4.1, is used for the simulations and experiments. This test is used in [25] as a way to compare the performance of dynamic positioning control algorithms. The ship is first initialized in dynamic positioning to point straight North at heading $0\,(deg)$. Then the following setpoint changes are commanded:

1. Position change $2\,[m]$ straight North: Tests a pure surge movement ahead.

2. Position change $2\,[m]$ straight East: Tests a pure sway movement in the starboard direction.

3. Heading change $45\,[deg]$ clockwise: Tests a pure yaw motion while keeping the position steady.

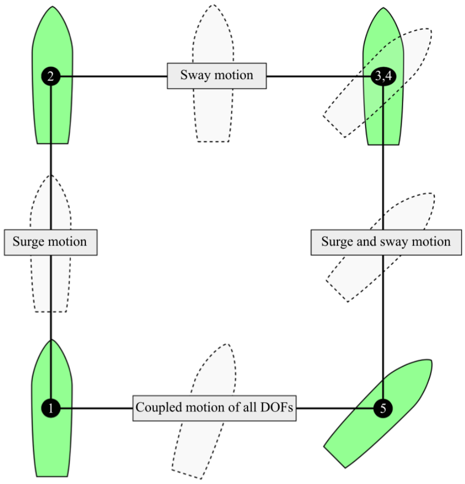


Figure 4.1: The 4-corner dynamic positioning test. Modified from [25].

4. Position change $2\,[m]$ straight South: Tests a combined surge-sway movement while keeping heading steady.

5. Position change $2\,[m]$ straight West and heading change $45\,[deg]$ counterclockwise: Tests a combined surge-sway-yaw movement.

The system is implemented such that the target will automatically change setpoint when the ship is within $0.005\,[m]$ from the target in both $x$ and $y$ position and $0.3\,[deg]$ from the target heading. When the 4-corner test is completed, the ship will have returned accurately to its initial position and heading, ready for a new test at the same pose and along the same track.

### Trajectory tracking

Note that for such a 4-corner test, the target motion is simplified to a setpoint, hence $\dot{\boldsymbol{\eta}}_t = \ddot{\boldsymbol{\eta}}_t = 0$, unless a trajectory tracking module is used. The simulations and most laboratory experiments will be conducted with setpoint tracking, but some experiments are conducted using trajectory tracking.

Trajectory tracking is defined as a control system that forces the system output $\boldsymbol{y}(t) \in \mathbb{R}^m$ to track a desired output $\boldsymbol{y}_d(t) \in \mathbb{R}^m$, and is commonly implemented through low-pass filters for the generation of position, velocity and acceleration trajectories, as explained in [1].

## 4.2 Controller and actuator constraint parameters

The controller gains and the actuator constraints for the different numerical simulations will be presented along with the results. For convenience, the control laws for the different controllers are repeated here:

**LP-LV**:

$$\boldsymbol{\alpha} = -\boldsymbol{\Gamma}_1 \boldsymbol{z}_1 \tag{4.1}$$

$$\dot{\boldsymbol{\alpha}} = -\boldsymbol{\Gamma}_1 \dot{\boldsymbol{z}}_1 \tag{4.2}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{\Gamma}_2 \boldsymbol{z}_2, \tag{4.3}$$

**NP-LV**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{4.4}$$

$$\dot{\boldsymbol{\alpha}} = -\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1 \tag{4.5}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{\Gamma}_2 \boldsymbol{z}_2, \tag{4.6}$$

**NP-NV**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{4.7}$$

$$\dot{\boldsymbol{\alpha}} = -\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1 \tag{4.8}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2, \tag{4.9}$$

**DWC**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{4.10}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC}, \tag{4.11}$$

**DWC-FB**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{4.12}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2, \tag{4.13}$$

where $\boldsymbol{\alpha}_{DWC}$ and $\dot{\boldsymbol{\alpha}}_{DWC}$ are chosen by the dynamic window algorithm, and DWC-FB refers to the dynamic window-based controller with a nonlinear feedback term included. Note that these are modified for setpoint tracking where $\ddot{\boldsymbol{\eta}}_t = \dot{\boldsymbol{\eta}}_t = 0$.

In order to make the results as comparable as possible to the experimental results, the simulations are run using MATLAB/Simulink with the numerical solver ode4 (Runge-Kutta) with a fixed-step size of 0.01[s], which is the same solver and sampling time used in the MC-Lab.

## 4.3 Performance metrics

To evaluate and objectively compare the performance of the different controllers, performance metrics are used. Here, we define

$$\begin{aligned} \bar{e}_1(t) &\triangleq \sqrt{\bar{\boldsymbol{z}}_1^\top \bar{\boldsymbol{z}}_1} \\ \bar{e}_2(t) &\triangleq \sqrt{\bar{\boldsymbol{z}}_2^\top \bar{\boldsymbol{z}}_2}, \end{aligned}$$

as the error inputs for the performance metrics, where $\boldsymbol{z}_1$ is the pose error and defined in (3.1), and $\boldsymbol{z}_2$ is the velocity error and defined in (3.2), and where $\bar{\boldsymbol{z}}_1$ and $\bar{\boldsymbol{z}}_2$ are the normalized error signals in the intervals $[-0.5, 0.5]$ in the expected operational space of the ship [26]. It is necessary to normalize the metrics because they contain measurements in different units, and it is not desirable that one measurement impacts the metric more than the others. To obtain this normalization, the position errors are divided by $4\,[m]$ and the heading error by $\frac{\pi}{2}\,[rad]$, since the position errors will be in the intervals $[-2, 2]\,[m]$ and the heading error in the interval $[-\frac{\pi}{4}, \frac{\pi}{4}]\,[rad]$. The velocity errors are divided by $2\boldsymbol{\nu}_{max}$, where the maximum surge and sway speeds and yaw rate are described in Section 2.1.2, since, e.g. for surge speed $u$, the velocity error will be in the interval $[-u_{max}, u_{max}]$, where $u_{max}$ is the maximum steady-state surge velocity. These signals represent the instantaneous control errors, while it is wanted to consider the accumulated errors over time. Therefore, the performance metric integral of the absolute error (IAE)

$$IAE(t) \triangleq \int_0^t |\bar{e}(\sigma)| d\sigma, \tag{4.14}$$

which integrates the temporal evolution of the absolute error, is used. Also, the integral of the absolute error multiplied by the energy consumption (IAEW) [16]

$$IAEW(t) \triangleq \int_0^t |\bar{e}(\sigma)| d\sigma \int_0^t P(\sigma) d\sigma, \tag{4.15}$$

with

$$P(t) = |\boldsymbol{\nu}(t)^\top \boldsymbol{\tau}|, \tag{4.16}$$

which represents the mechanical power, is applied. IAEW thus indicates which controller has the best combined control accuracy and energy use in one single metric. Then, using the integral of absolute differentiated control (IADC) [14], [26], defined as

$$IADC(t) \triangleq \int_0^t |\dot{\bar{\tau}}(\sigma)| d\sigma, \tag{4.17}$$

where

$$\bar{\tau} = \sqrt{\boldsymbol{\tau}^\top \boldsymbol{\tau}}, \tag{4.18}$$

and $\dot{\bar{\tau}}$ is computed using numerical derivation, which penalizes actuator changes, and thus a measurement for actuator wear and tear is obtained.

## 4.4 Main simulation results

### 4.4.1 Comparison of the cascaded feedback controllers

Here, a comparison of the unconstrained cascaded feedback controllers are shown. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.1. These gains have been chosen such that the LP-LV time constants for the kinematic and kinetic controllers are $T_p = 12.5 > T_v = 5$ and $T_\psi = 14.32 > T_r = 5.73$, which satisfy the tuning rules. Furthermore, $\Delta_p$, $\Delta_\psi$, $\Delta_v$ and $\Delta_r$ are chosen such that the time constant with the kinematic control loop do not become smaller than that of the kinetic control loop. With the chosen $\Delta$-values, the time constants in the linear region associated with the NP-LV controller are $T_p = 6.25 > T_v = 5$ and $T_\psi = 7.16 > T_r = 5.73$, while the time constants in the linear region associated with the NP-NV controller are $T_p = 6.25 > T_v = 3.5$ and $T_\psi = 7.16 > T_r = 5.73$.

In Figure 4.2, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained LP-LV-controlled ship, the dash-dotted black outline represents the

| | **LP-LV** | **NP-LV** | **NP-NV** |
|---|---|---|---|
| $\boldsymbol{\Gamma_1}$ | diag([0.08, 0.08, 0.0698]) | –\|\|– | –\|\|– |
| $\boldsymbol{\Gamma_2}$ | diag([0.2, 0.2, 0.1745])$\boldsymbol{M}$ | –\|\|– | –\|\|– |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | – | [0.5, 0.5] | –\|\|– |
| $\Delta_{\tilde{v},\tilde{r}}$ | – | – | [0.7, 1.0] |
| $\boldsymbol{m}$ | – | – | – |
| $\boldsymbol{r}$ | – | – | – |
| $\boldsymbol{K}$ | – | – | – |

Table 4.1: Controller gains and actuator constraint parameters for the simulations with the cascaded feedback controllers.

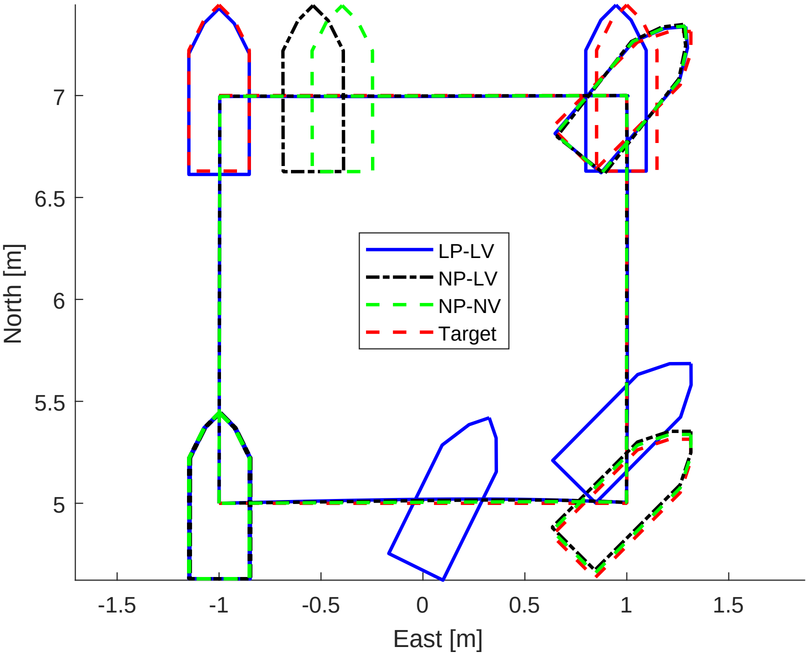


Figure 4.2: 4-corner path plot for the cascaded feedback controllers.

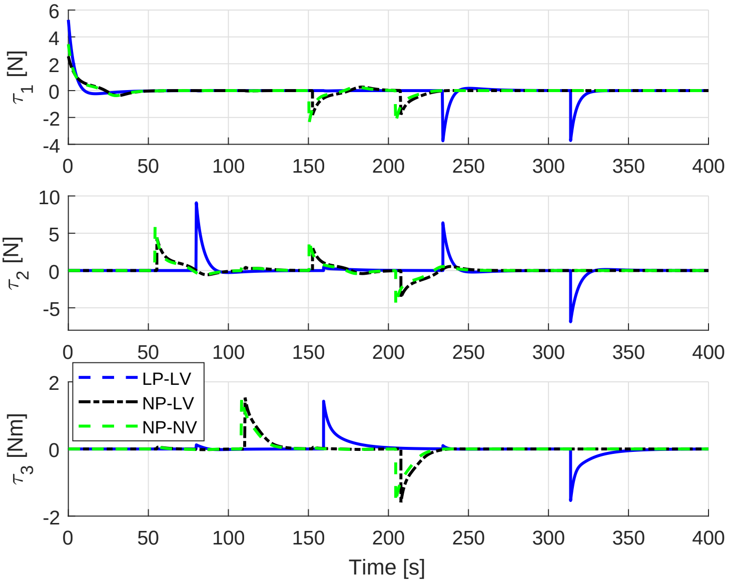


Figure 4.3: Commanded actuator inputs for the cascaded feedback controllers.

unconstrained NP-LV-controlled ship, the dashed green outline represents the unconstrained NP-NV-controlled ship, while the dashed red outline represents the 4-corner target reference. It should be noted that the ship heading is not properly portrayed in these plots, but that they are used to display the trajectory of each controller. To get a better understanding of the best performing controller in terms of position and heading, the performance metrics should be used.

Figure 4.3 shows the commanded control inputs for the three different controllers. It can be seen that the unconstrained LP-LV controller greatly exceeds the actuator constraints of CSAD presented in Section 2.3, while the two other controllers have a more conservative performance. This plot also shows that the LP-LV controller is slower than the other two controllers, as it uses a longer time to stabilize around the setpoint limits for the 4-corner test presented in Section 4.1. The NP-LV and NP-NV controllers have similar performance, but where the NP-NV controller is slightly faster.

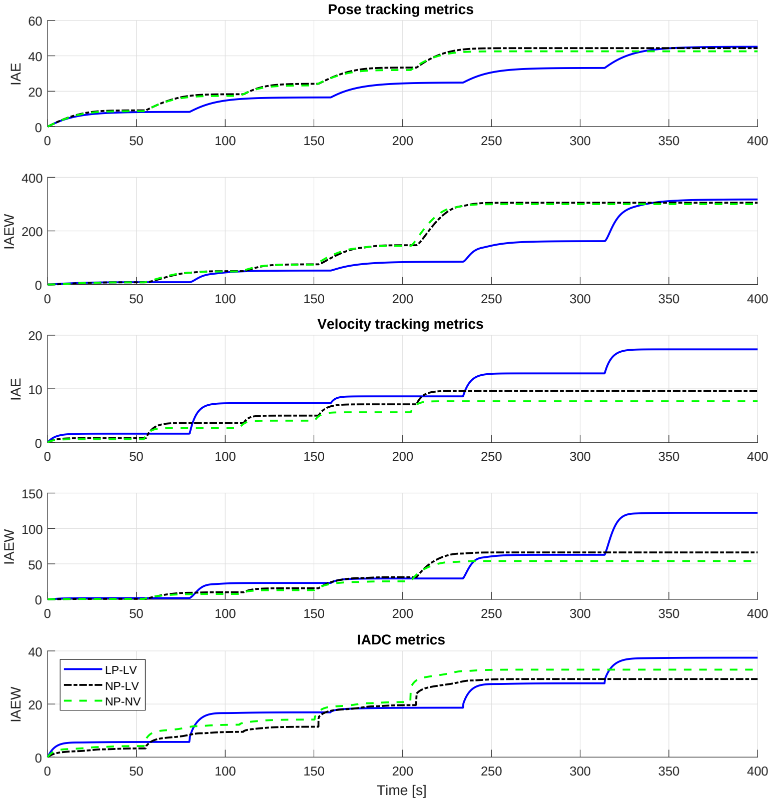


Figure 4.4: IAE, IAEW and IADC for the cascaded feedback controllers.

| | **LP-LV** | **NP-LV** | **NP-NV** |
|---|---|---|---|
| **IAE** $e_1$ | 45.112 | 44.338 | **42.548** |
| **IAEW** $e_1$ | 317.450 | 305.267 | **299.787** |
| **IAE** $e_2$ | 17.356 | 9.608 | **7.670** |
| **IAEW** $e_2$ | 122.133 | 66.154 | **54.039** |
| **IADC** | 37.458 | **29.422** | 32.958 |

Table 4.2: Performance metrics for the cascaded feedback controllers.

Figure 4.4 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The IAE and IAEW metrics for pose tracking show that the three controllers have similar performance in the sense of accuracy and energy usage, but where the LP-LV controller has a slower transient performance. However, the IAE and IAEW metrics for velocity tracking show that the LP-LV controller is less accurate and energy efficient compared to its nonlinear counterparts, where the NP-NV controller comes out as the best performing controller. The IADC metrics show that, in the sense of actuator usage and wear and tear, the NP-LV controller has the best performance. The final values for the performance metrics are summarized in Table 4.2, where the best performing controller is marked in bold, and where it is shown that the NP-NV controller has the best overall performance of the cascaded feedback controllers.

### 4.4.2 LP-LV with constraint handling methods

Here, simulation results for the LP-LV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.3.

In Figure 4.5, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained LP-LV-controlled ship, the dash-dotted black outline represents the magnitude constrained LP-LV-controlled ship, the dashed green outline represents the LP-LV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

Figure 4.6 shows the commanded control inputs for the three different controllers. It can be seen that the unconstrained LP-LV controller greatly exceeds the actuator constraints of CSAD presented in Section 2.3, while the two other

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\vert\vert-$ | $-\vert\vert-$ |
| $\boldsymbol{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\vert\vert-$ | $-\vert\vert-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $-$ | $-$ | $-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $-$ | $-$ |
| $\boldsymbol{m}$ | $-$ | $[2.88, 1.60, 1.36]$ | $-\vert\vert-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[2.88, 1.60, 1.36]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[5, 2.78, 2.36]$ |

Table 4.3: Controller gains and actuator constraint parameters for the simulations with the LP-LV controller with and without actuator constraint handling methods.

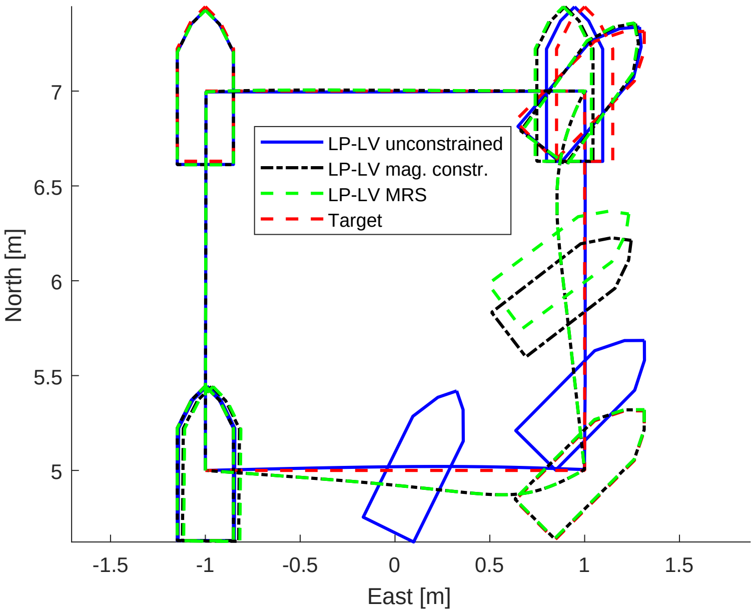


Figure 4.5: 4-corner path plot for the LP-LV controller with and without actuator constraint handling methods.

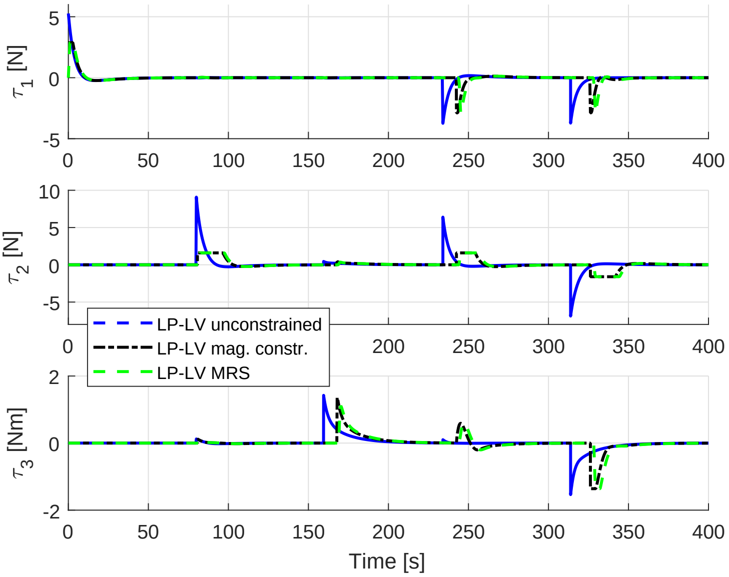


Figure 4.6: Commanded actuator inputs for the LP-LV controller with and without actuator constraint handling methods.

controllers have a more conservative performance, limited by the designed constraints. It is important to remember that the unconstrained case is unrealistic, because the simulated ship operates with no upper and lower limits on its actuators, and thus the result is only used as a benchmark in order to see how the different actuator constraint models affect performance.

Figure 4.7 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The IAE and IAEW metrics for pose tracking show that introducing actuator constraints decreases the overall pose tracking accuracy, but has little impact on energy use. The IAE and IAEW metrics for velocity tracking show that actuator constraints results in a negative impact on performance in velocity tracking. However, the IADC metrics show that actuator constraints can have a positive impact on overall actuator usage and wear and tear. The final values for the performance metrics are summarized in Table 4.4, where the best performing

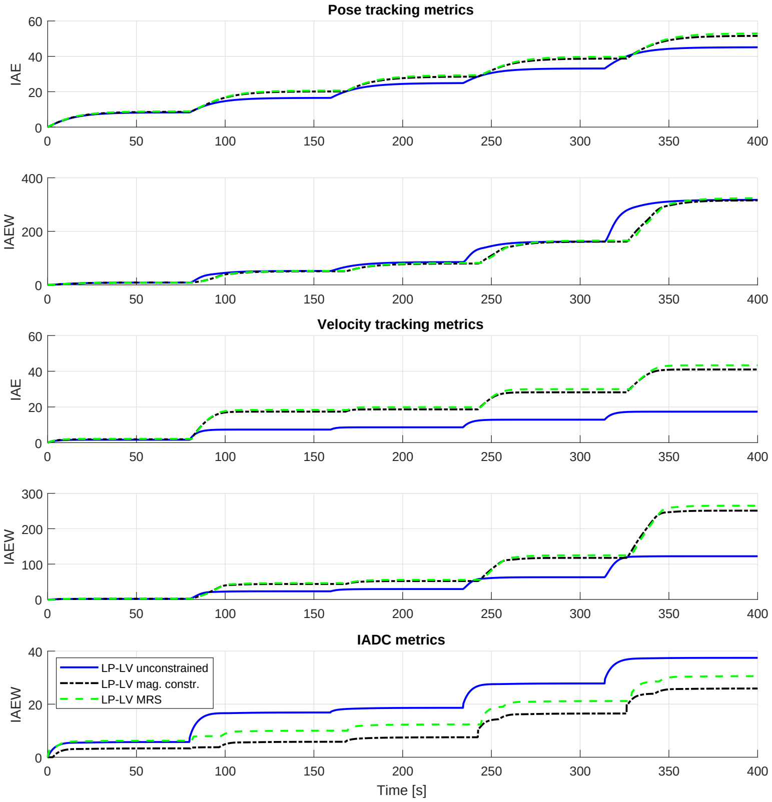


Figure 4.7: IAE, IAEW and IADC for the LP-LV controller with and without actuator constraint handling methods.

controller is marked in bold.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | **45.112** | 51.558 | 52.938 |
| **IAEW** $e_1$ | 317.450 | **315.666** | 323.815 |
| **IAE** $e_2$ | **17.356** | 40.990 | 43.326 |
| **IAEW** $e_2$ | **122.133** | 250.962 | 265.020 |
| **IADC** | 37.458 | **35.901** | 30.550 |

Table 4.4: Performance metrics for the LP-LV controller with and without constraint handling methods.

### 4.4.3 NP-LV with constraint handling methods

Here, simulation results for the NP-LV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.5.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\mathbf{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ | $-\|\|-$ |
| $\mathbf{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $-$ | $-$ |
| $\boldsymbol{m}$ | $-$ | $[2.88, 1.60, 1.36]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[2.88, 1.60, 1.36]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[5, 2.78, 2.36]$ |

Table 4.5: Controller gains and actuator constraint parameters for the simulations with the NP-LV controller with and without actuator constraint handling methods.

In Figure 4.8, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained NP-LV-controlled ship, the dash-dotted black outline represents the magnitude constrained NP-LV-controlled ship, the dashed green outline represents the NP-LV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

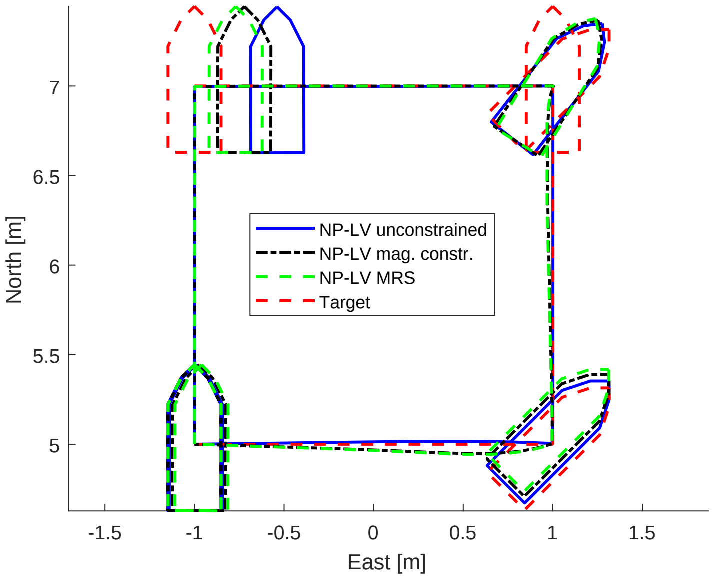


Figure 4.8: 4-corner path plot for the NP-LV controller with and without actuator constraint handling methods.

Figure 4.9 shows the commanded control inputs for the three different controllers. It can be seen that the unconstrained NP-LV controller greatly exceeds the actuator constraints in of CSAD in sway motion, as presented in Section 2.3, while the two other controllers have a more conservative performance, limited by the designed constraints. It is important to remember that the unconstrained case is unrealistic, because the simulated ship operates with no upper and lower limits on its actuators, and thus the result is only used as a benchmark in order to see how the different actuator constraint models affect performance.

Figure 4.10 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The IAE and IAEW metrics for pose tracking show that introducing actuator constraints decreases the overall pose tracking accuracy by a small amount, but has little impact on energy use. The IAE and IAEW metrics for velocity tracking show that actuator constraints results in a negative impact on performance in

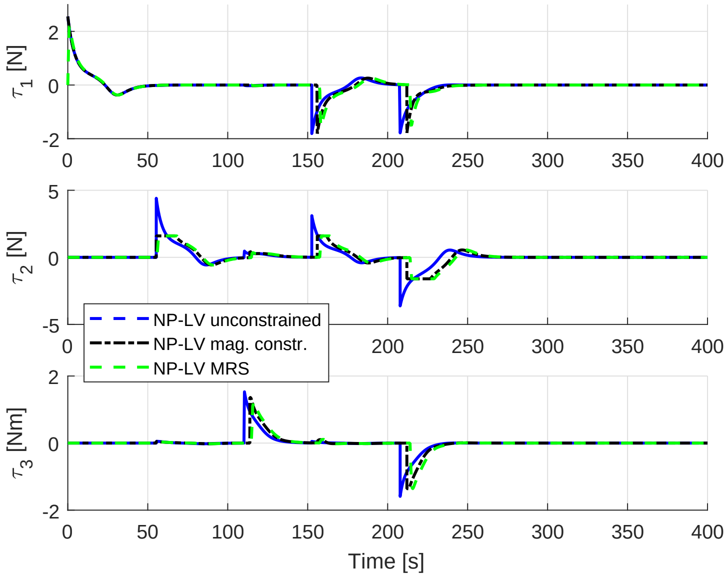


Figure 4.9: Commanded actuator inputs for the NP-LV controller with and without actuator constraint handling methods.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | **44.338** | 47.999 | 48.788 |
| **IAEW** $e_1$ | **305.267** | 314.623 | 322.975 |
| **IAE** $e_2$ | **9.608** | 15.185 | 16.569 |
| **IAEW** $e_2$ | **66.154** | 100.583 | 109.653 |
| **IADC** | 29.422 | **21.269** | 25.622 |

Table 4.6: Performance metrics for the NP-LV controller with and without constraint handling methods.

velocity tracking. However, the IADC metrics show that actuator constraints can have a positive impact on overall actuator usage and wear and tear. The final values for the performance metrics are summarized in Table 4.6, where the best performing controller is marked in bold.

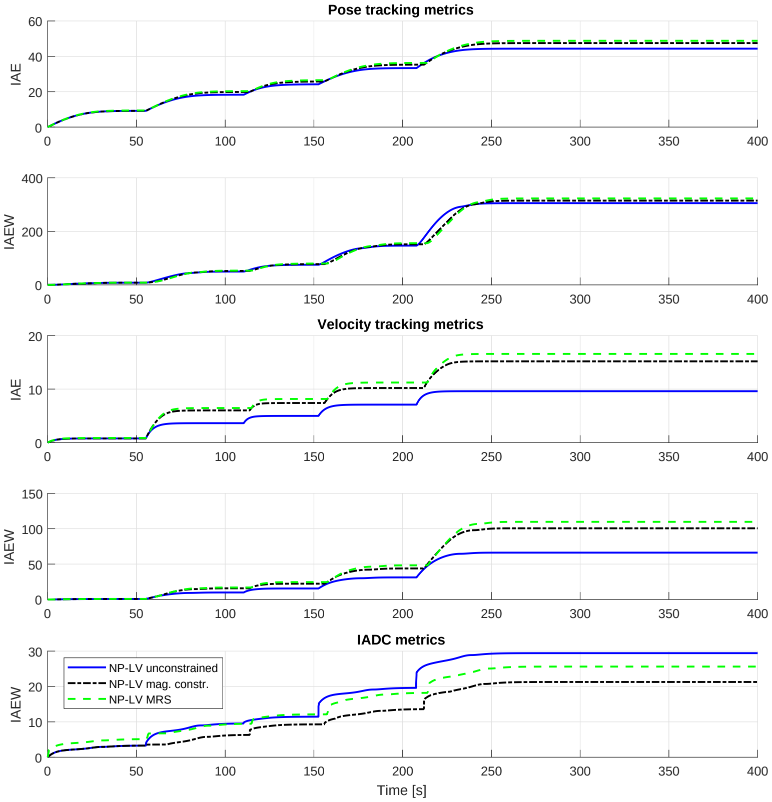


Figure 4.10: IAE, IAEW and IADC for the NP-LV controller with and without actuator constraint handling methods.

### 4.4.4 NP-NV with constraint handling methods

Here, simulation results for the NP-NV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.7.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\mathbf{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\Vert-$ | $-\Vert-$ |
| $\mathbf{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\Vert-$ | $-\Vert-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\Vert-$ | $-\Vert-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-\Vert-$ | $-\Vert-$ |
| $\boldsymbol{m}$ | $-$ | $[2.88, 1.60, 1.36]$ | $-\Vert-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[2.88, 1.60, 1.36]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[5, 2.78, 2.36]$ |

Table 4.7: Controller gains and actuator constraint parameters for the simulations with the NP-NV controller with and without actuator constraint handling methods.

In Figure 4.11, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained NP-NV-controlled ship, the dash-dotted black outline represents the magnitude constrained NP-NV-controlled ship, the dashed green outline represents the NP-NV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

Figure 4.12 shows the commanded control inputs for the three different controllers. It can be seen that the unconstrained NP-NV controller greatly exceeds the actuator constraints in of CSAD in sway motion, as presented in Section 2.3, while the two other controllers have a more conservative performance, limited by the designed constraints. It is important to remember that the unconstrained case is unrealistic, because the simulated ship operates with no upper and lower limits on its actuators, and thus the result is only used as a benchmark in order to see how the different actuator constraint models affect performance.

Figure 4.13 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The IAE and IAEW metrics for pose tracking show that introducing actuator constraints decreases the overall pose tracking accuracy by a small amount, but has little impact on energy use. The IAE and IAEW metrics for velocity tracking

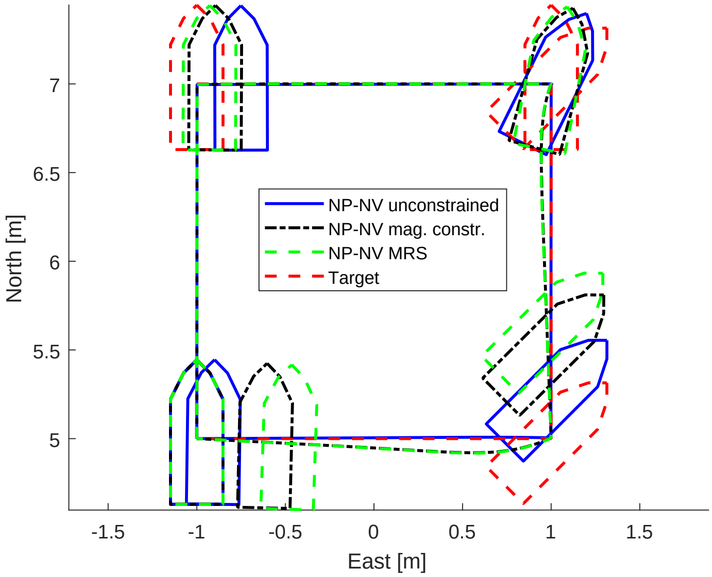


Figure 4.11: 4-corner path plot for the NP-NV controller with and without actuator constraint handling methods.

show that actuator constraints results in a negative impact on performance in velocity tracking. However, the IADC metrics show that actuator constraints can have a positive impact on overall actuator usage and wear and tear. The final values for the performance metrics are summarized in Table 4.8, where the best performing controller is marked in bold.

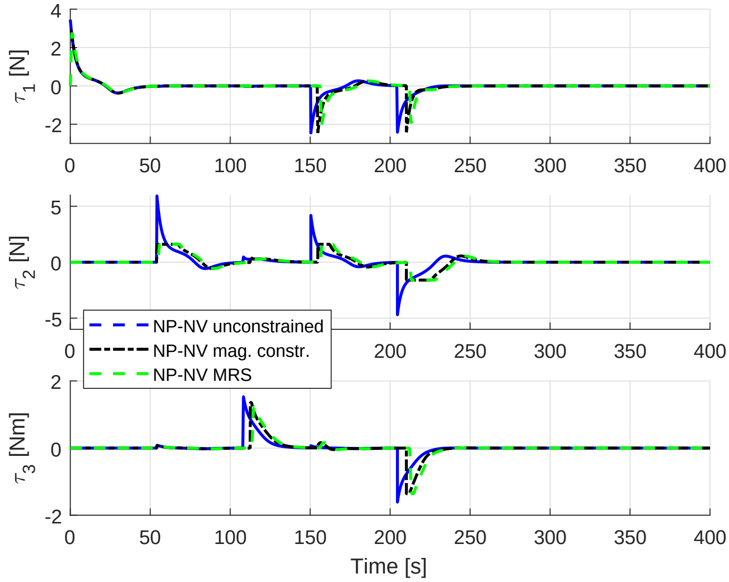


Figure 4.12: Commanded actuator inputs for the NP-NV controller with and without actuator constraint handling methods.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | **42.548** | 46.519 | 47.874 |
| **IAEW** $e_1$ | **299.787** | 312.903 | 321.806 |
| **IAE** $e_2$ | **7.670** | 14.718 | 16.108 |
| **IAEW** $e_2$ | **54.039** | 98.996 | 108.277 |
| **IADC** | 32.958 | **21.114** | 28.553 |

Table 4.8: Performance metrics for the NP-NV controller with and without constraint handling methods.

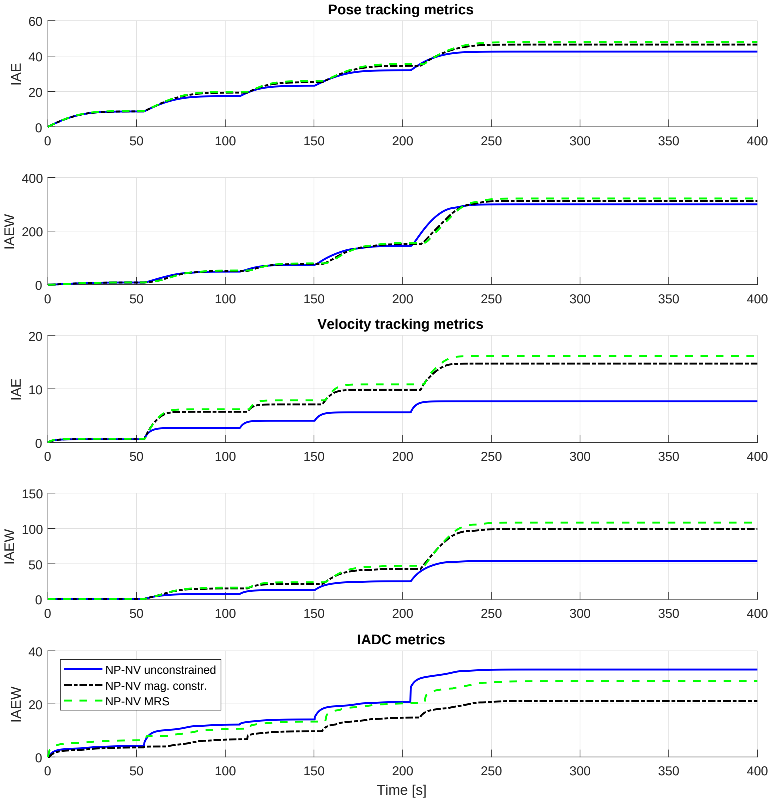


Figure 4.13: IAE, IAEW and IADC for the NP-NV controller with and without actuator constraint handling methods.

### 4.4.5 DWC

Here, simulation results for the dynamic window-based controller will be presented. It will be tested against a magnitude constrained NP-NV, where the magntiude constraints are equal to the constraint the DWC is designed with. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.9. The DWC is simulated without the feedback, as it has no impact on the control performance where no uncertainties or disturbances are involved.

| | **NP-NV** | **DWC** |
|---|---|---|
| $\boldsymbol{\Gamma_1}$ | diag$([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | diag$([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-$ |
| $\boldsymbol{m}$ | $[3.5996, 2.0003, 1.7027]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $-$ |
| $\boldsymbol{K}$ | $-$ | $-$ |

Table 4.9: Controller gains and actuator constraint parameters for the simulation with the DWC.

In Figure 4.14, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner target reference.

Figure 4.15 shows the commanded control inputs for the two different controllers. It can be seen that the DWC keeps the control inputs higher longer than the NP-NV controller, since the DWC tracks the optimal velocity $\boldsymbol{\alpha}_f$ which is on the boundaries of the window of possible velocities $V_p$, unless the target velocity $\boldsymbol{\alpha}$ is inside the velocity window. It can then be seen that the NP-NV controller has a more conservative behaviour.

Figure 4.16, 4.17 and 4.18 illustrates how the surge speed, sway speed and yaw rate moves in the velocity space in order to track the target through the 4-corner test. Here, the blue dash-dotted outline represents the DWC, while the NP-NV controller, represented by the black dashed outline, is included for comparative purposes.

Figure 4.19 illustrates closer how the DWC tracks the desired velocity. The

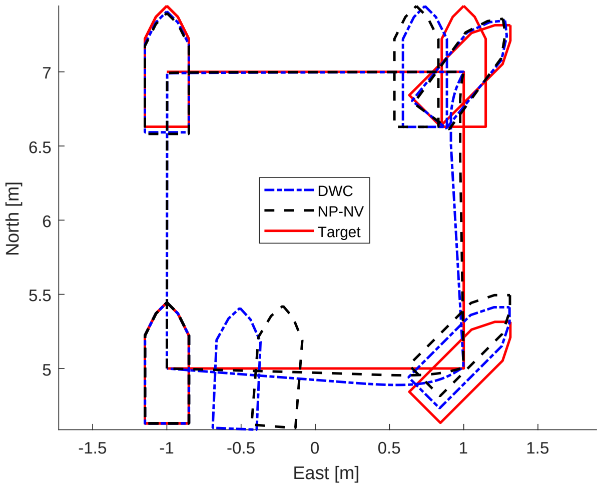


Figure 4.14: 4-corner path plot for the DWC and the NP-NV controller.

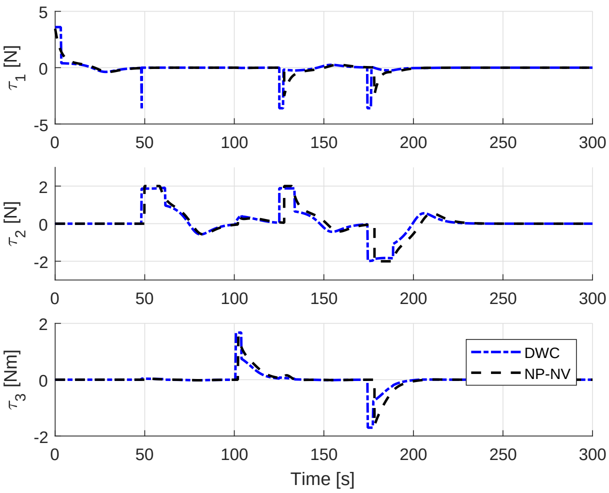


Figure 4.15: Commanded actuator inputs for the DWC and the NP-NV controller.

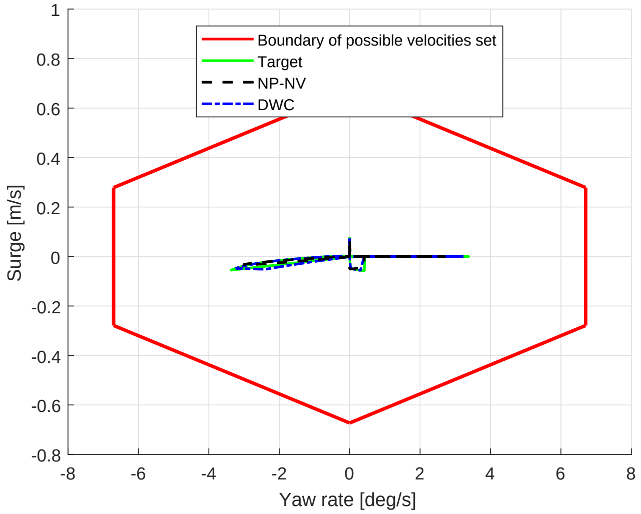


Figure 4.16: Yaw rate-surge window for the 3 DOF DWC.

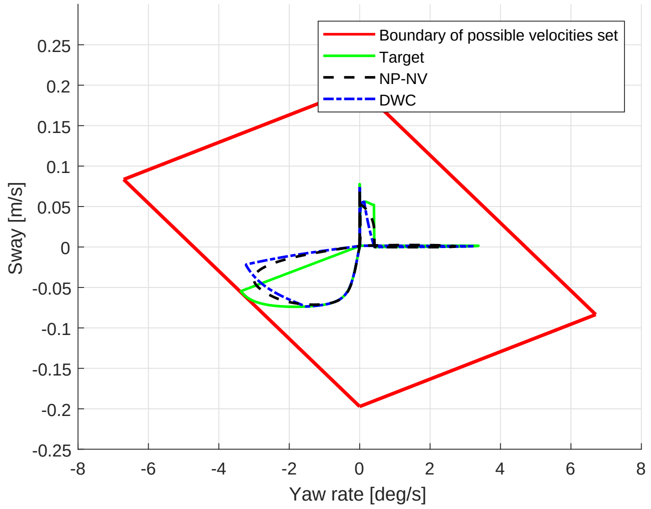


Figure 4.17: Yaw rate-sway window for the 3 DOF DWC.

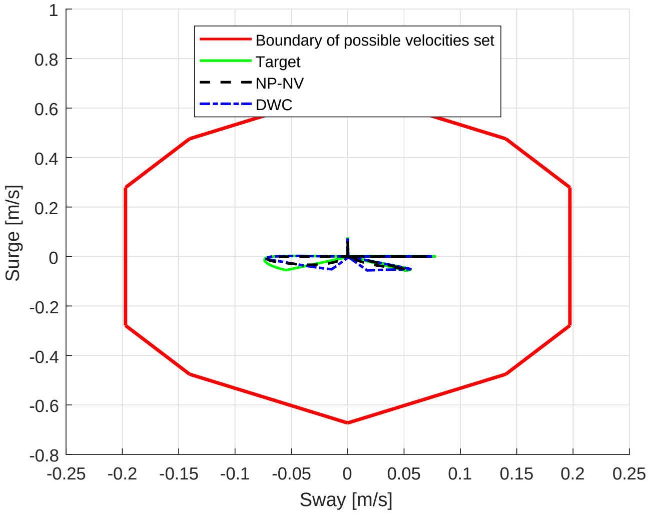


Figure 4.18: Sway-surge window for the 3 DOF DWC.

tracking of surge speed is used for this illustration. Here the upper and lower limits of the velocity window, given by (3.100) and (3.99), respectively, are represented by the blue and red outline, respectively, while the green outline represents the current velocity of CSAD, the dashed magenta outline the desired velocity $\alpha_u$, and the dash-dotted black outline the chosen velocity $\alpha_{f,u}$. It is shown how the DW algorithm chooses a chosen velocity $\alpha_{f,u}$ on the border of its window in order to track the desired velocity $\alpha_u$, where the size of the window satisfies the actuator constraints. In Figure 4.20, for illustrative purposes the same result is plotted for the NP-NV controller, where it can be seen that the NP-NV controller has a slower convergence towards the desired velocity $\alpha_u$.

Figure 4.21 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The IAE and IAEW metrics for pose tracking show that the DWC has a more accurate and energy efficient pose tracking performance. The IAE and IAEW metrics for velocity tracking show that the DWC has a more accurate and more energy efficient velocity tracking performance as well. It can also be seen that the DWC is slightly faster at performing the 4-corner test than the NP-NV controller. The IADC metrics show that the DWC has a bigger usage of the

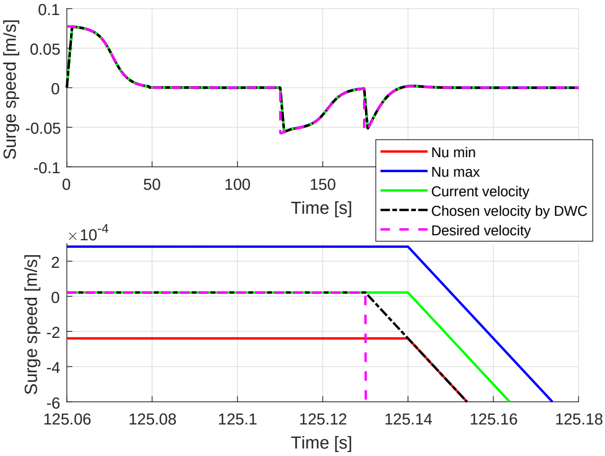


Figure 4.19: Tracking of the desired surge speed $\alpha_u$ for the DWC, where the upper and lower limits of the velocity window are included. The top figure represents the whole 4-corner test, while the bottom is zoomed in for illustrative purposes.

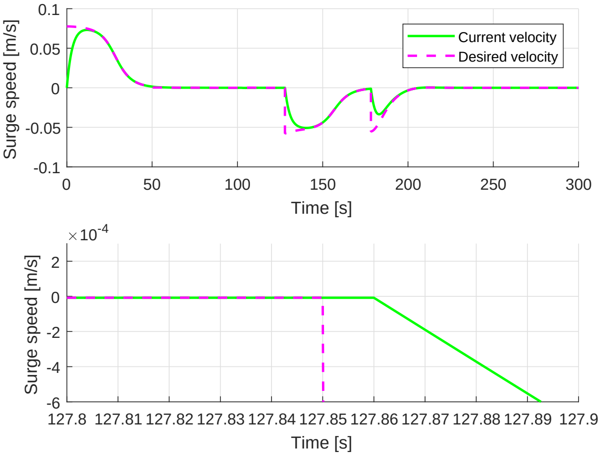


Figure 4.20: Tracking of the desired surge speed $\alpha_u$ for the NP-NV controller.

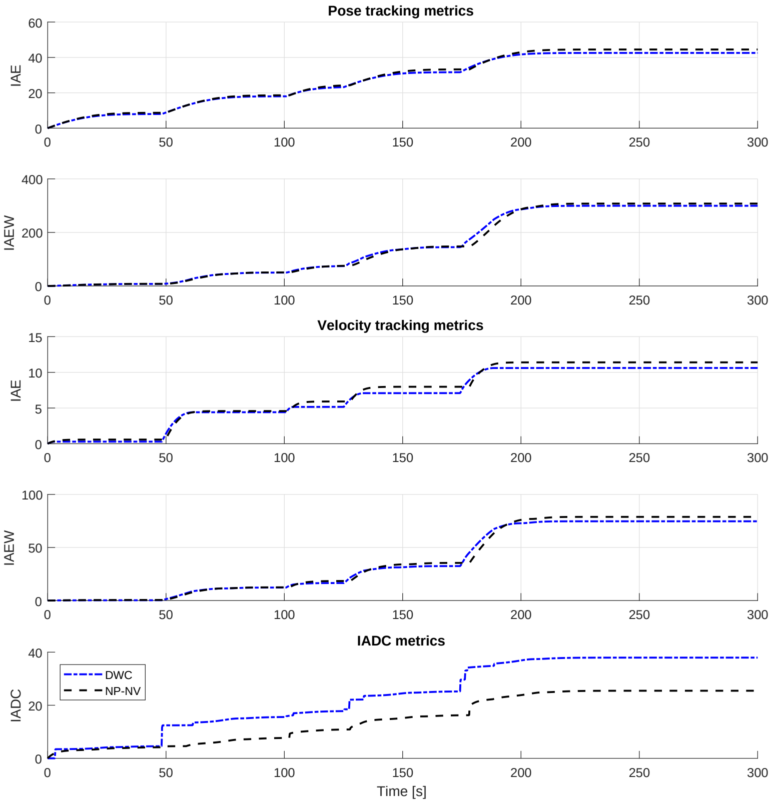


Figure 4.21: IAE, IAEW and IADC for pose tracking for the DWC and the NP-NV controller.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 44.487 | **42.541** |
| **IAEW** $e_1$ | 307.659 | **299.399** |
| **IAE** $e_2$ | 11.396 | **10.610** |
| **IAEW** $e_2$ | 78.813 | **74.669** |
| **IADC** | **25.464** | 37.950 |

Table 4.10: Performance metrics for the simulation with the DWC and the NP-NV controller.

actuators, which is to be expected by the result shown in Figure 4.15, where the commanded actuator inputs have a higher rate of change than for the NP-NV controller. The final values for the performance metrics are summarized in Table 4.10, where the best performing controller is marked in bold, and it is shown that the DWC has the overall best performance.

### DWC - Less aggressive pose tracking

It was desired to test how the DWC would perform when the pose tracking-gains were lowered. Lowering $\mathbf{\Gamma}_2$ to diag([0.03, 0.03, 0.349]) for both controllers, while the rest of the parameters were kept as in Table 4.9, the two controllers were simulated conducting the 4-corner test again.

In Figure 4.22, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner target reference. Notice how the simulated ships follow much closer to target reference, than when the pose tracking-gain was higher.

In Table 4.11, the final values for the performance metrics for this simulation are summarized. Here, the best performing controller is marked in bold, and it is shown that the DWC again has the overall best performance, beating NP-NV in all metrics except for IADC. Notice the substantial difference in velocity tracking performance. Comparing this result to the main result in Table 4.10, we can see that the energy use for both pose and velocity tracking has been severely reduced. By reducing the pose tracking-gain, it is desired to follow a lower velocity, and thus making the IAE metrics for velocity tracking lower. The IAE metrics for pose tracking has about doubled, due to the usage of setpoint tracking and a longer simulation time for the 4-corner test, but the overall accuracy of the ship performance has increased. Based on this, one can conclude that if accuracy is more important than the time used for transit, lower pose tracking-gains are beneficial. This statement is valid for all the controllers presented in this thesis.

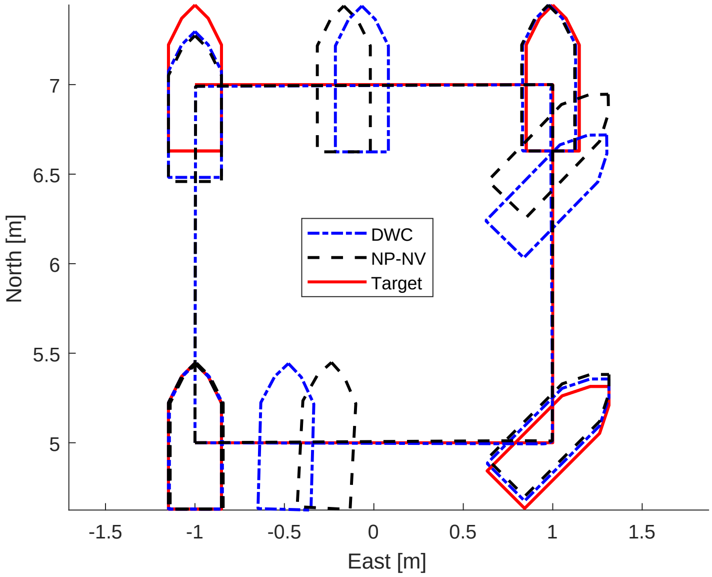


Figure 4.22: 4-corner path plot for the DWC and the NP-NV controller with lowered pose tracking-gains.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 96.698 | **91.644** |
| **IAEW** $e_1$ | 207.050 | **201.850** |
| **IAE** $e_2$ | 3.156 | **1.324** |
| **IAEW** $e_2$ | 6.757 | **2.916** |
| **IADC** | **10.946** | 26.559 |

Table 4.11: Performance metrics for the simulation with the DWC and the NP-NV controller with lowered pose tracking-gains.

## 4.5 Extended simulation results

In this section, simulations which are exposed to disturbances, uncertainties and time delays are presented. The purpose is to analyse how the different controllers perform when exposed to such effects, and to try to mimic the laboratory environment to have a better comparison between the simulations and the experiments done in Chapter 5. The results from this section is meant to be used as a part of the discussion in Chapter 6.

### 4.5.1 DWC with model uncertainties

Here, it is desired to analyse how model uncertainties impact the performance of the DWC and the NP-NV controller. Because of this, simulation results for different amounts of model uncertainties are presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.12. The DWC is simulated with a nonlinear feedback, such as in (3.126) and discussed in Section 3.2.4.

| | **NP-NV** | **DWC** |
|---|---|---|
| $\boldsymbol{\Gamma_1}$ | diag([0.08, 0.08, 0.0698]) | −\|\|− |
| $\boldsymbol{\Gamma_2}$ | diag([0.2, 0.2, 0.1745])$\boldsymbol{M}$ | −\|\|− |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | [0.5, 0.5] | −\|\|− |
| $\Delta_{\tilde{v},\tilde{r}}$ | [0.7, 1.0] | −\|\|− |
| $\boldsymbol{m}$ | [3.5996, 2.0003, 1.7027] | −\|\|− |
| $\boldsymbol{r}$ | − | − |
| $\boldsymbol{K}$ | − | − |

Table 4.12: Controller gains and actuator constraint parameters for the simulation with the DWC with model uncertainties.

The simulations are done with 0%, +15%, -30% and +50% model uncertainties, where the matrices $\boldsymbol{M}$ in (2.4), $\boldsymbol{C}$ in (2.7). and $\boldsymbol{D}$ in (2.12), are adjusted according to the percentages. Here, the figures show the effects on performance on the extreme case of +50% model uncertainties, while the rest of the results are summed up in Table 4.13.

In Figure 4.23, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner

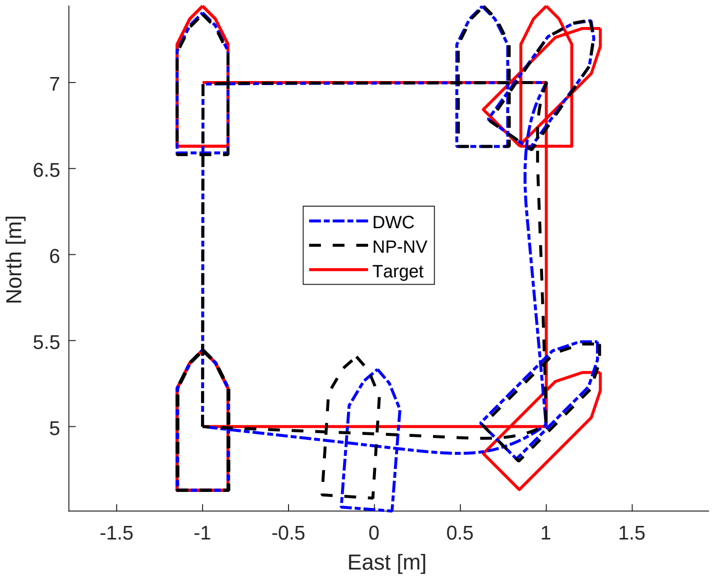


Figure 4.23: 4-corner path plot for the DWC and the NP-NV controller, here for the case of +50% model uncertainties.

target reference.

Figure 4.24 shows the commanded control inputs for the two different controllers. It can be seen that the model uncertainties have not created stability issues for either of the controllers, and that they both have a similar performance as in the simulation conducted without model uncertainties, as shown in Figure 4.15, but where the usage of the actuators in $\tau_2$ is slightly higher.

As an example of velocity tracking performance, Figure 4.25 illustrates how the sway speed and yaw rate moves in the velocity space in order to track the target through the 4-corner test. Here, the blue dash-dotted outline represents the DWC, while the NP-NV controller, represented by the black dashed outline, is included for comparative purposes. When comparing Figure 4.25 to Figure 4.17, it can be seen that the model uncertainties have had some negative performance impacts on the velocity tracking performance of both of the controllers.

This is illustrated further in Figure 4.26, where it is shown how the DWC tracks the surge speed. Here the upper and lower limits of the velocity window,

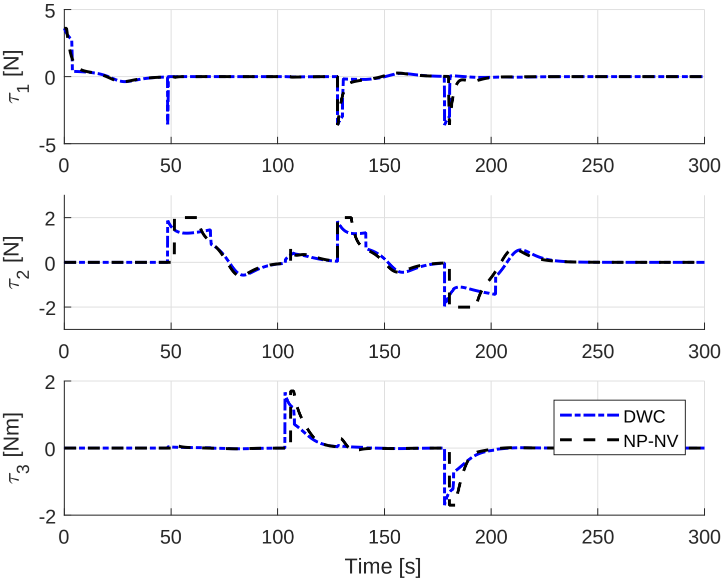


Figure 4.24: Commanded actuator inputs for the DWC and the NP-NV controller, here for the case of +50% model uncertainties.

given by (3.100) and (3.99), respectively, are represented by the blue and red outline, respectively, while the green outline represents the current velocity of CSAD, the dashed magenta outline the desired velocity $\alpha_u$, and the dash-dotted black outline the chosen velocity $\alpha_{f,u}$. It is shown how the DWC chooses a chosen velocity $\alpha_{f,u}$ on the border of its window when tracking the desired velocity, but that the window size is smaller compared to the one in Figure 4.19 because the increase in the model parameters have reduced the size of the dynamic velocity window.

Figure 4.27 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The IAE and IAEW metrics for pose tracking show that the DWC still have a more energy efficient performance than the NP-NV controller, but is now less accurate. The IAE and IAEW metrics for velocity tracking show that the NP-NV controller has a more accurate and more energy efficient velocity

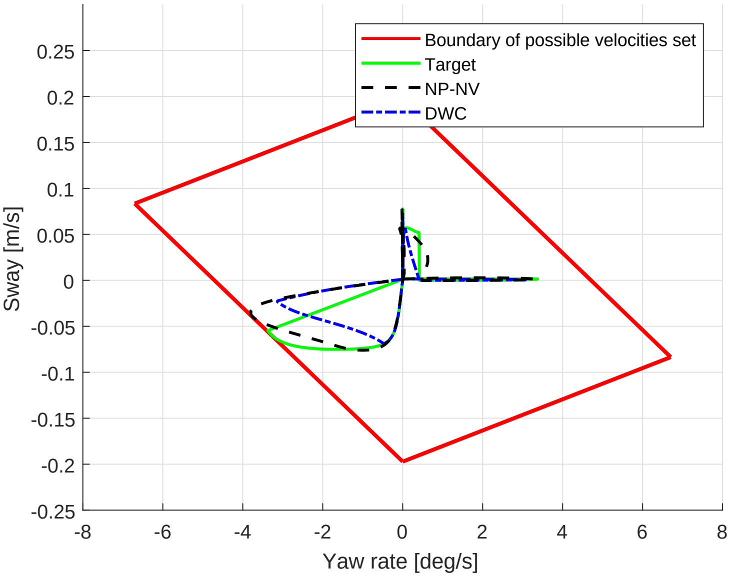


Figure 4.25: Yaw rate-sway window for the 3 DOF DWC, here for case of +50% model uncertainties.

tracking performance as well. It can also be seen that the DWC still is slightly faster at performing the 4-corner test than the NP-NV controller. The IADC metrics show that the DWC has a bigger usage of the actuators, which is to be expected by the result shown in Figure 4.24, where the commanded actuator inputs have a higher rate of change than for the NP-NV controller. The final values for the performance metrics are summarized in Table 4.13, where the result for different model uncertainties are displayed. For every simulation, the best performing controller is marked in bold. It can be seen that when no model uncertainties are introduced, the DWC with a nonlinear feedback is clearly the best performing controller, as was shown in Section 4.4.5. When the values of the model parameters are reduced, the DWC still performs better than the NP-NV controller, but when the parameters are increased, the NP-NV has a better overall performance. It can be concluded that in terms of controller robustness, the NP-NV controller wins because the DWC is a highly model-based controller, and will require a significantly accurate model to work optimally in real life

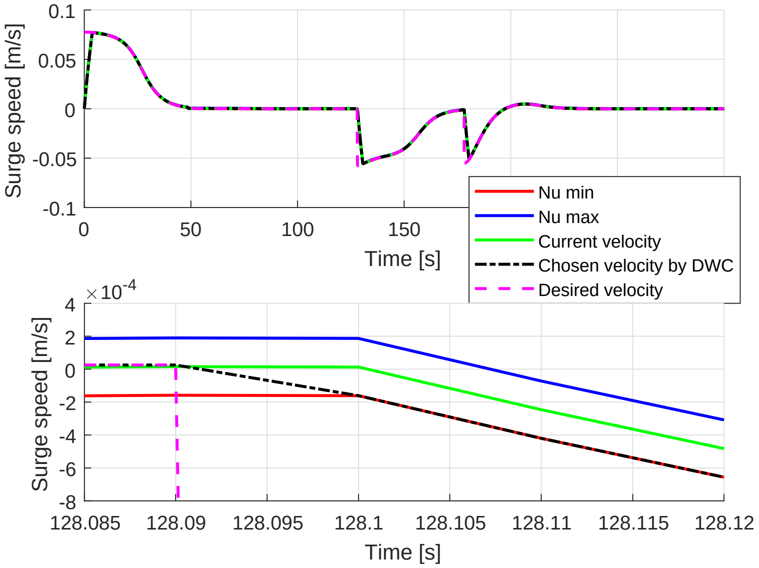


Figure 4.26: Tracking of the desired surge speed $\alpha_u$ for the DWC, here for case of $+50\%$ model uncertainties, where the upper and lower limits of the velocity window are included. The top figure represents the whole 4-corner test, while the bottom is zoomed in for illustrative purposes.

scenarios. However, the most important result from this simulation is that introducing model uncertainties into the different control laws do not create instabilities in the performance, meaning that they are both feasible to use in real life operations, even if the ship model they are based on is not perfectly modeled.

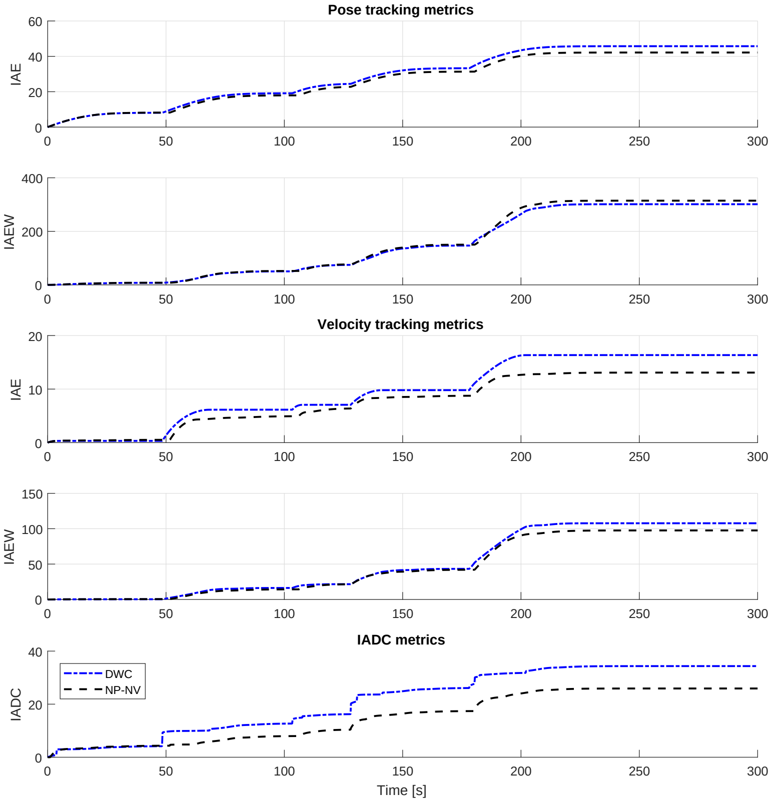


Figure 4.27: IAE, IAEW and IADC for pose tracking for the DWC and the NP-NV controller, here for the case of +50% model uncertainties.

| **0%** | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 44.487 | **42.660** |
| **IAEW** $e_1$ | 307.659 | **299.346** |
| **IAE** $e_2$ | 11.396 | **10.854** |
| **IAEW** $e_2$ | 78.813 | **76.163** |
| **IADC** | **25.464** | 38.430 |

| **+15%** | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | **43.488** | 43.524 |
| **IAEW** $e_1$ | 309.897 | **299.999** |
| **IAE** $e_2$ | **11.569** | 12.354 |
| **IAEW** $e_2$ | **82.443** | 85.152 |
| **IADC** | **24.435** | 40.963 |

| **-30%** | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 48.139 | **42.139** |
| **IAEW** $e_1$ | 300.999 | **298.870** |
| **IAE** $e_2$ | 17.049 | **9.889** |
| **IAEW** $e_2$ | 106.601 | **69.978** |
| **IADC** | **17.601** | 35.947 |

| **+50%** | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | **42.162** | 45.735 |
| **IAEW** $e_1$ | 314.442 | **301.348** |
| **IAE** $e_2$ | **13.081** | 16.350 |
| **IAEW** $e_2$ | **97.558** | 107.695 |
| **IADC** | **25.464** | 34.342 |

Table 4.13: Performance metrics for the simulation with the DWC and the NP-NV controller with model uncertainties. The actual percentage of model uncertainties are shown in the upper-left corner of each table.

### 4.5.2 DWC with estimation noise

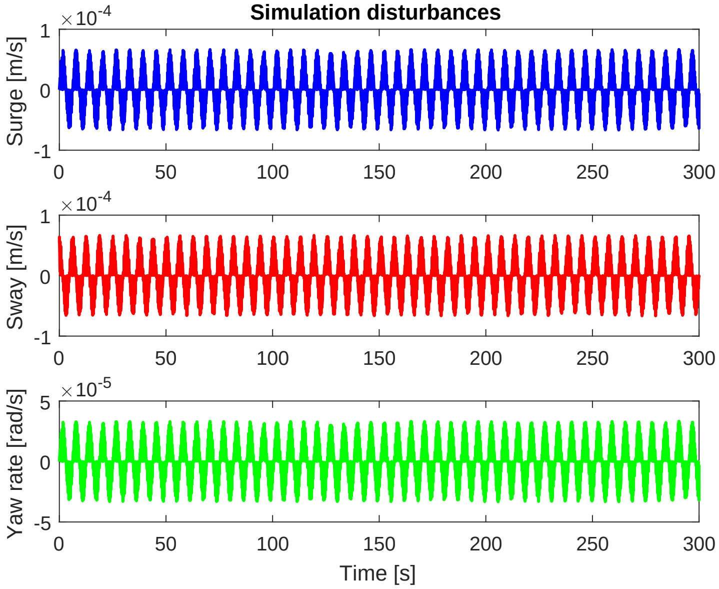


Figure 4.28: The noise which is added to the velocity to mimic estimation noise.

Here, it is desired to analyse how estimation noise impact the performance of the DWC and the NP-NV controller. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are the same as in the previous simulation, and are shown in Table 4.12. The DWC is simulated with a nonlinear feedback, such as in (3.126) and discussed in Section 3.2.4.

The simulations are done with noise which is meant to represent a small amount of noise on the velocity estimate, causing small twitches and thus increasing the total use of the actuators in order to compensate for the velocity measurement noise. It was desired to closely analyse how noise like this would impact the overall control performance, and especially the commanded actuator input and energy efficiency. The noise added is meant to only create small twitches, and is represented by a cosine, in which the amplitude is $|A_u| = |A_v| < 10^{-5}$ [m/s] for surge speed and sway speed, respectively, and

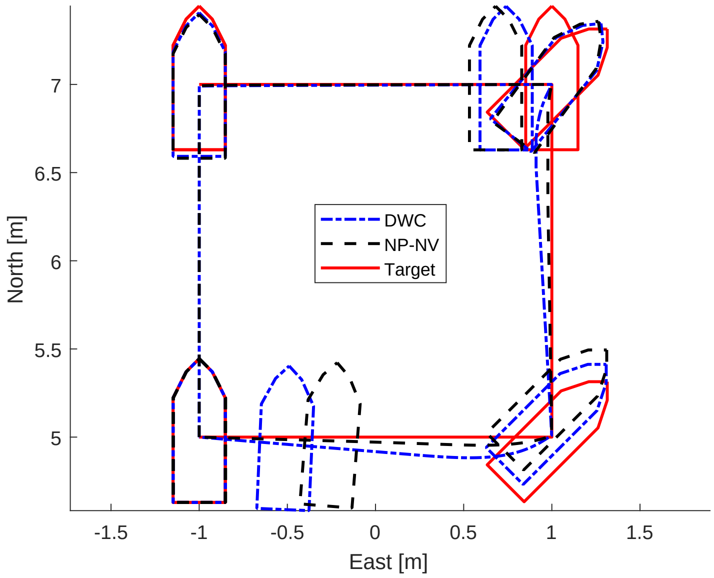


Figure 4.29: 4-corner path plot for the DWC and the NP-NV controller with velocity estimation noise.

$|A_r| < 10^{-6}$ [rad/s] for yaw rate, and is shown in Figure 4.28.

In Figure 4.29, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner target reference.

Figure 4.30 shows the commanded control inputs for the two different controllers. It can be seen that the presence of velocity estimation noise has caused a severe actuator twitching for the DWC, while the NP-NV controller is more or less unaffected. The loss of performance for the DWC can be explained by the fact that the DWC is a model predictive controller, and when the velocity at the next time step is not how it predicted, it will try to compensate for this by either increasing or decreasing thrust, resulting in actuator twitching. It can also be seen that adding a nonlinear feedback was not enough to compensate

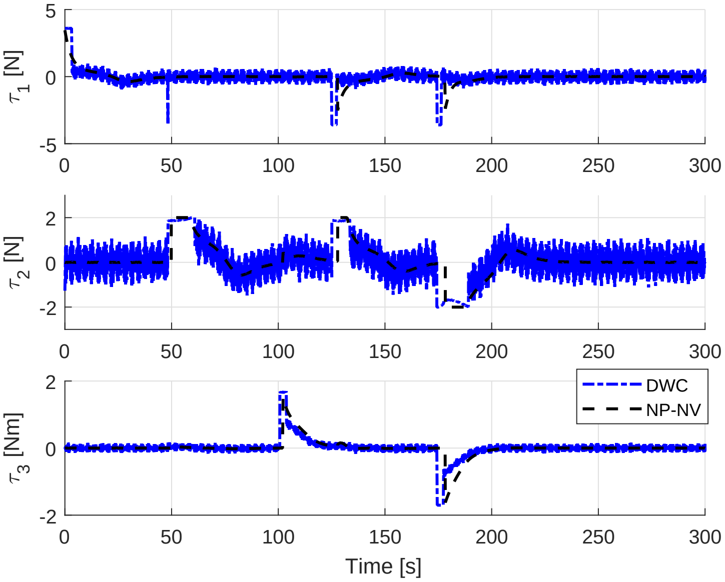


Figure 4.30: Commanded actuator inputs for the DWC and the NP-NV controller with estimation noise.

for such noise. As can be seen in the path plot in Figure 4.29, the DWC still navigates the 4-corner in more or less the same way as before, but compared to the NP-NV controller, it has a performance loss due to how it selects the desired acceleration, given by $\dot{\boldsymbol{\alpha}}_{DWC}$ in (3.123). The small measurement noise is enough to make instabilities for the DW algorithm, creating severe twitching in the choice of $\dot{\boldsymbol{\alpha}}_{DWC}$ as shown for desired surge acceleration in Figure 4.31, and thus affecting performance through the commanded control input.

Figure 4.32 illustrates how the DWC tracks the target surge speed. Here the upper and lower limits of the velocity window, given by (3.100) and (3.99), respectively, are represented by the blue and red outline, respectively, while the green outline represents the current velocity of CSAD, the dashed magenta outline the desired velocity $\alpha_u$, and the dash-dotted black outline the chosen velocity $\alpha_{f,u}$. It is shown that the DWC chooses a desired velocity at the border of its velocity window, even when in the presence of noise. However, the small vibrations in velocity $\boldsymbol{\nu}$ caused by the estimation noise, causes the desired

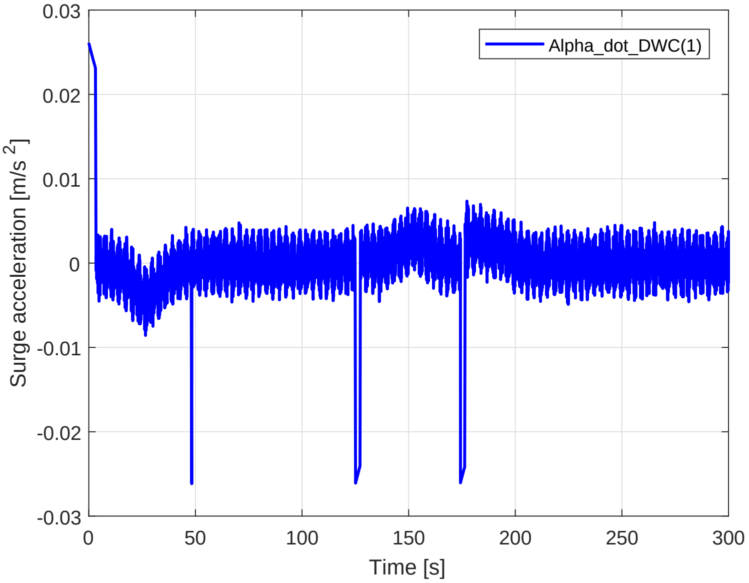


Figure 4.31: The desired acceleration $\dot{\boldsymbol{\alpha}}_{DWC}$ in surge when measurement noise is present.

acceleration to oscillate, which in the end causes severe actuator twitching. It is then clear that the DW algorithm is very sensitive in its prediction, and we see that even when the predicted velocity is just a few $\mu m/s$ off at every time step, it can cause problems, which indicates that the DWC is in need of very accurate measurements to be feasible in real life operations.

Figure 4.33 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The IAE and IAEW metrics for pose tracking show that the DWC still has a better accuracy, while NP-NV has a more energy efficient behaviour. The same result is repeated in the IAE and IAEW metrics for velocity tracking. It can be seen that the DWC still is slightly faster at performing the 4-corner test. However, one should remember that this result is not completely comparable to real life experiments as no rate constraints are presents in this simulation environment, and thus one can expect the controllers to behave differently in a laboratory

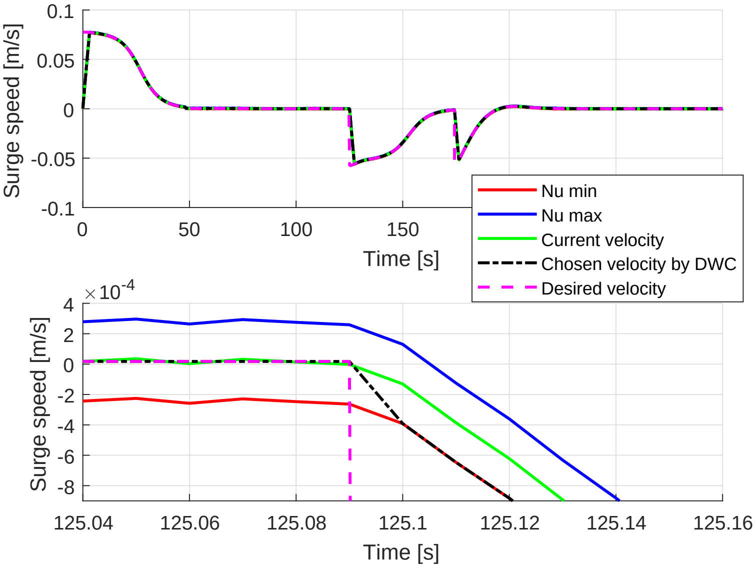


Figure 4.32: Tracking of the desired surge speed $\alpha_u$ for the DWC with estimation noise, where the upper and lower limits of the velocity window are included. The top figure represents the whole 4-corner test, while the bottom is zoomed in for illustrative purposes.

environment, and that the experimental performance of DWC would decrease if it was tested through laboratory experiments were such noise were present. The final values for the performance metrics are summarized in Table 4.14, where the best performing controller is marked in bold. Comparing the values to the ones in Table 4.13 for the case of 0% model uncertainties, it can be seen that the performance of the NP-NV controller is almost unaffected by the velocity measurement noise, except for in the IADC metrics. It can be concluded that the NP-NV controller is the most robust controller of the two, and that it has the overall best performance and ability to handle small amounts of noise in the measurement signal.

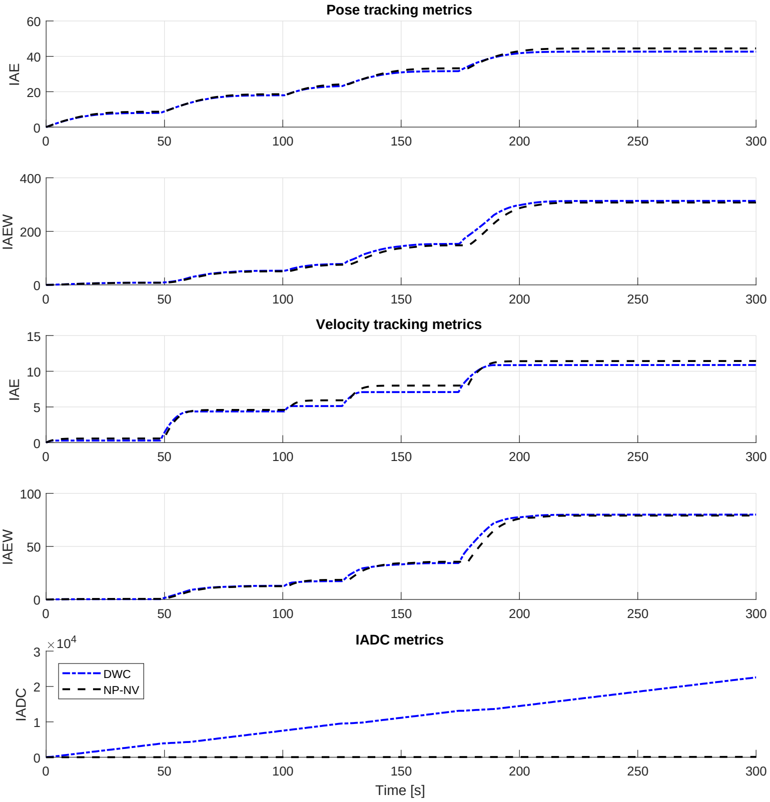


Figure 4.33: IAE, IAEW and IADC for pose tracking for the DWC and the NP-NV controller when measurement noise is present.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 44.487 | **42.661** |
| **IAEW** $e_1$ | **307.641** | 313.597 |
| **IAE** $e_2$ | 11.435 | **10.876** |
| **IAEW** $e_2$ | **79.529** | 79.951 |
| **IADC** | **95.529** | 22568 |

Table 4.14: Performance metrics for the simulation with the the DWC and the NP-NV controller with estimation noise.

### 4.5.3 DWC with a system time delay

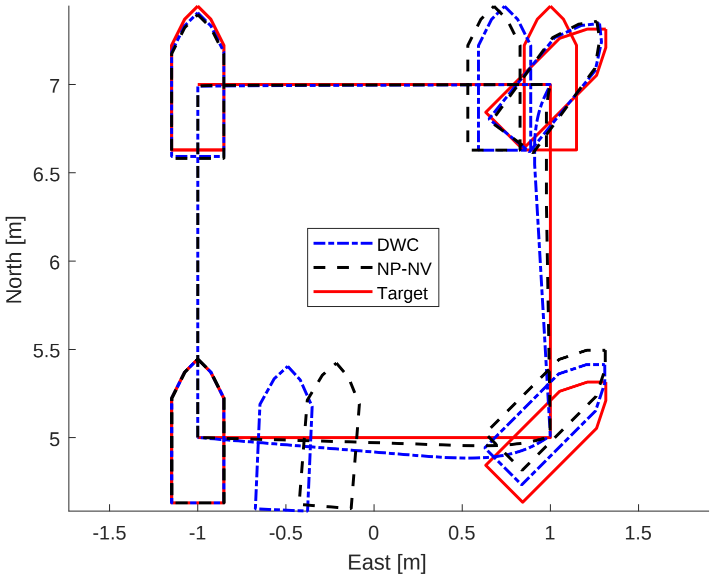


Figure 4.34: 4-corner path plot for the DWC and the NP-NV controller with a system time delay.

Here, it is desired to analyse how a system time delay impacts the performance of the DWC and the NP-NV controller. The time delay $t_d = 0.02$ [s] is

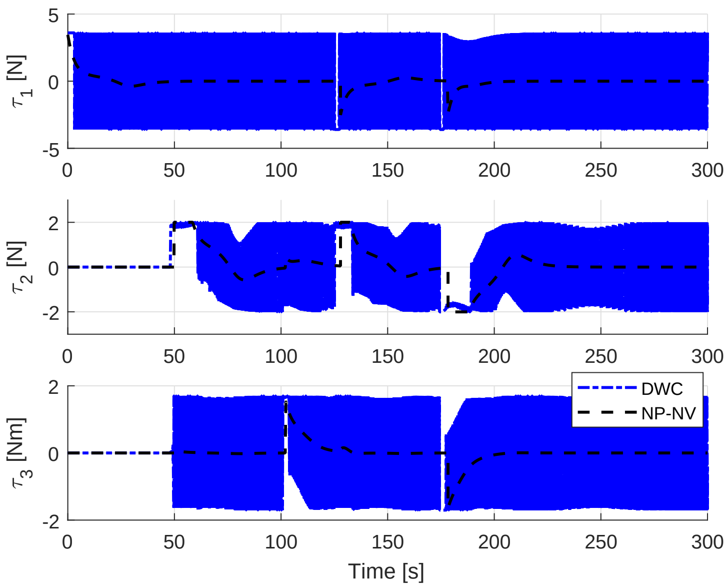


Figure 4.35: Commanded actuator inputs for the DWC and the NP-NV controller with a system time delay.

added to the commanded actuator inputs during simulation. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are the same as in the previous simulation, and shown in Table 4.12. The DWC is simulated with a nonlinear feedback, such as in (3.126) and discussed in Section 3.2.4.

In Figure 4.34, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner target reference.

Figure 4.35 shows the commanded control inputs for the two different controllers. It can be seen that the presence of a system time delay has caused the actuators to oscillate for the DWC, while the NP-NV controller is more or less unaffected. As in Section 4.5.2, the current velocity does not match the

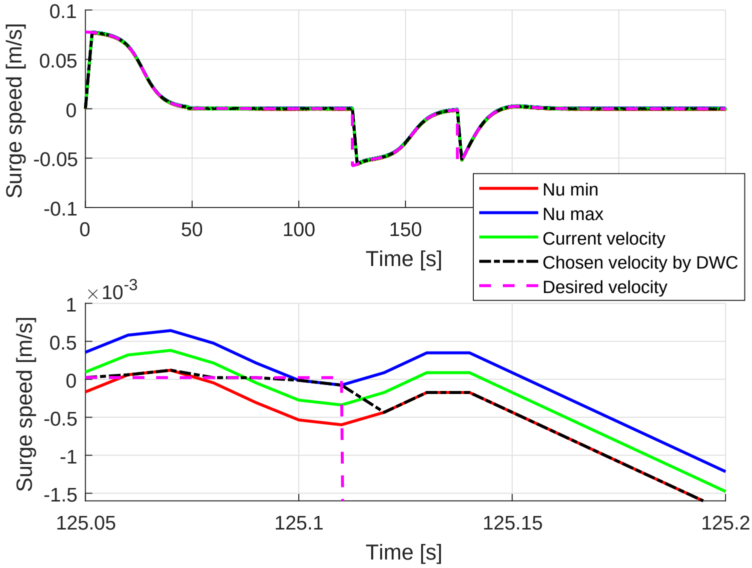


Figure 4.36: Tracking of the desired surge speed $\alpha_u$ for the DWC with a system time delay, where the upper and lower limits of the velocity window are included. The top figure represents the whole 4-corner test, while the bottom is zoomed in for illustrative purposes.

predicted velocity at the next time step, causing severe actuator twitching and performance loss. As can be seen in the path plot in Figure 4.34, the DWC still navigates the 4-corner in more or less the same way as before, but compared to the NP-NV controller, it has a performance loss due to how it selects the desired acceleration, given by $\dot{\boldsymbol{\alpha}}_{DWC}$ in (3.123).

Figure 4.36 illustrates how the DWC tracks the target surge speed. Here the upper and lower limits of the velocity window, given by (3.100) and (3.99), respectively, are represented by the blue and red outline, respectively, while the green outline represents the current velocity of CSAD, the magenta outline the desired velocity $\alpha_u$, and the dash-dotted black outline the chosen velocity $\alpha_{f,u}$. Notice how the current velocity oscillates around the desired velocity.

Figure 4.37 shows the performance metrics IAE and IAEW for pose tracking

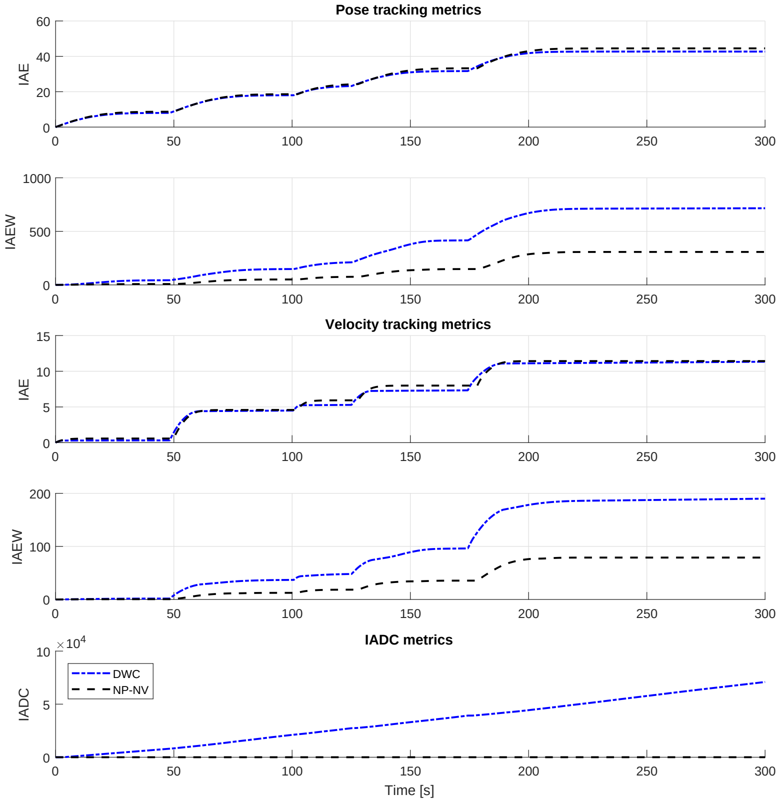


Figure 4.37: IAE, IAEW and IADC for pose tracking for the DWC and the NP-NV controller, where a system time delay is present.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | 44.507 | **42.698** |
| **IAEW** $e_1$ | **307.460** | 713.507 |
| **IAE** $e_2$ | 11.425 | **11.333** |
| **IAEW** $e_2$ | **78.926** | 189.910 |
| **IADC** | **25.491** | 70901 |

Table 4.15: Performance metrics for the simulation with the DWC and the NP-NV controller with a system time delay.

and velocity tracking, along with the IADC metric, for the two controllers. The IAE and IAEW metrics show that the DWC has a better accuracy, but that the NP-NV controller is much more energy efficient. It can be seen that the DWC still is slightly faster at performing the 4-corner test. However, one should remember that this result is not completely comparable to real life experiments as no rate constraints are presents in this simulation environment, and thus one can expect the controllers to behave differently in a laboratory environment, and that the experimental performance of DWC would decrease if it was tested through laboratory experiments were such a time delay was present. The final values for the performance metrics are summarized in Table 4.15, where the best performing controller is marked in bold. Comparing the values to the ones in Table 4.13 for the case of 0% model uncertainties, it can be seen that the performance of the NP-NV controller is almost unaffected by the system time delay. It can then be concluded that, once again, the NP-NV controller is the most robust controller of the two, and that it has the best overall performance and ability to handle system time delays.

### 4.5.4 DWC with all disturbances

The previous sections have analysed how different types of disturbances affect the performance of the DWC and the NP-NV controller. It is now desired to analyse how a combination of all of the different disturbances will affect the overall performance. This combination can represent a simulated laboratory environment, where different types of disturbances will be present. Here, the simulations are done with similar estimation noise as in Section 4.5.2, with +20% model uncertainties, and with a system time delay $t_d = 0.09$ [s], which is supposed to mimic the time delay caused by the different operational frequencies for the different subsystems in the laboratory setup, see Chapter 5. In addition

| | **NP-NV** | **DWC** |
|---|---|---|
| $\mathbf{\Gamma_1}$ | $\text{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\mathbf{\Gamma_2}$ | $\text{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-\|\|-$ |
| $\boldsymbol{m}$ | $[3.5996, 2.0003, 1.7027]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $[5.76, 3.20, 2.72]$ | $-\|\|-$ |
| $\boldsymbol{K}$ | $[5, 2.78, 2.37]$ | $-\|\|-$ |

Table 4.16: Controller gains and actuator constraint parameters for the simulation with the DWC with all disturbances.

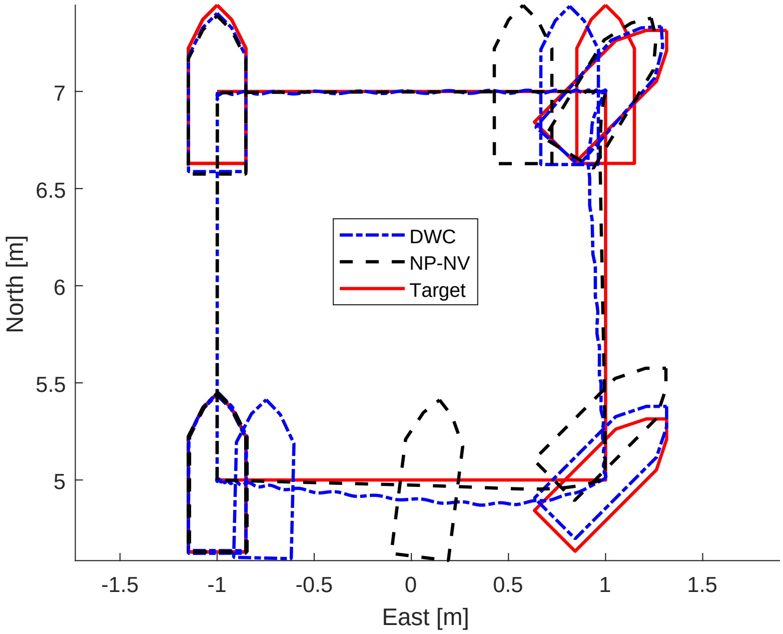


Figure 4.38: 4-corner path plot for the DWC and the NP-NV controller with all disturbances.

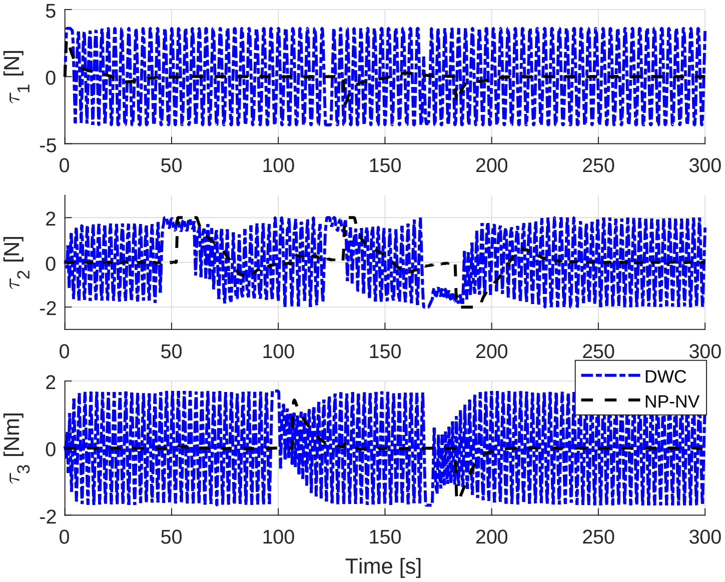


Figure 4.39: Commanded actuator inputs for the DWC and the NP-NV controller with all disturbances.

to this, in order to include rate constraints, the controllers are simulated with an MRS model. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this simulation are shown in Table 4.16, where the rate constraints are set sufficiently high so that they would not be too limiting on the performance. The DWC is simulated with a nonlinear feedback, such as in (3.126) and discussed in Section 3.2.4.

In Figure 4.38, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted blue outline represents the DWC, the dashed black outline represents the magnitude constrained NP-NV-controlled ship, while the red outline represents the 4-corner target reference.

Figure 4.39 shows the commanded control inputs for the two different controllers. It is shown that the combination of the disturbances have resulted in severe actuator twitching where the control input signal oscillates during the whole simulation, but where the oscillating is limited by the rate constraints in

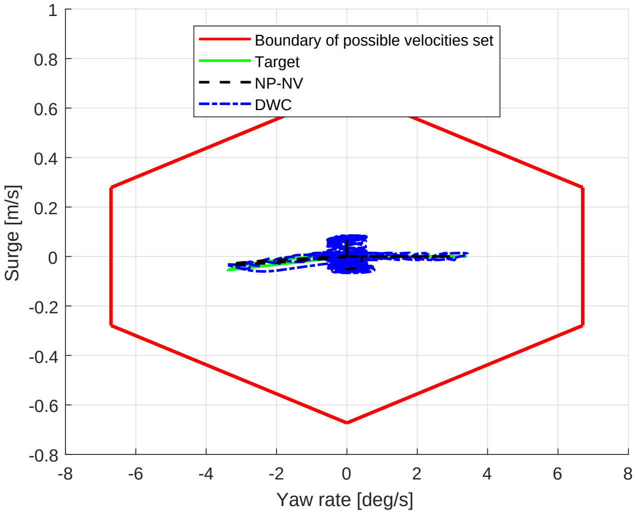


Figure 4.40: Yaw rate-surge window for the 3 DOF DWC.

the MRS model. We see that the combination of a noisy velocity estimate and a time delay has further decreased overall performance.

Figure 4.40, 4.41 and 4.42 illustrate how the surge speed, sway speed and yaw rate moves in the velocity space in order to track the target through the 4-corner test. Here, the blue dash-dotted outline represents the DWC, while the NP-NV controller, represented by the black dashed outline, is included for comparative purposes. It can be seen that the disturbance effects, as discussed in the previous sections, have a huge impact on the DWC's ability to track the desired velocity.

Figure 4.43 illustrates more closely how the DWC tracks the target surge speed. Here the upper and lower limits of the velocity window, given by (3.100) and (3.99), respectively, are represented by the blue and red outline, respectively, while the green outline represents the current velocity of CSAD, the dashed magenta outline the desired velocity $\alpha_u$, and the dash-dotted black outline the chosen velocity $\alpha_{f,u}$. Notice how the current velocity oscillates around the desired velocity.

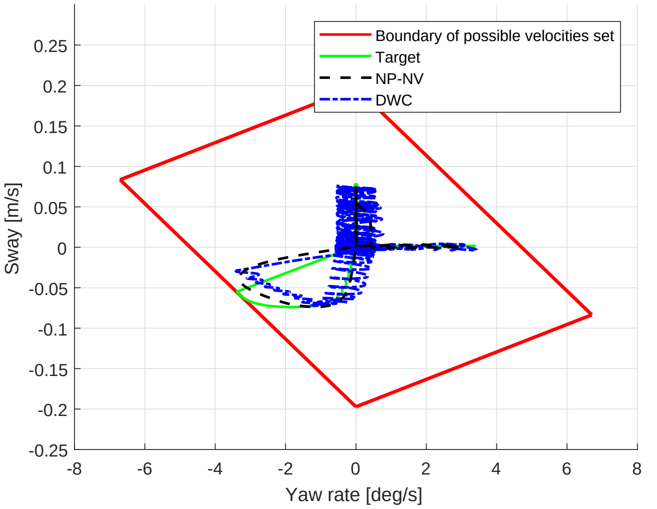


Figure 4.41: Yaw rate-sway window for the 3 DOF DWC.

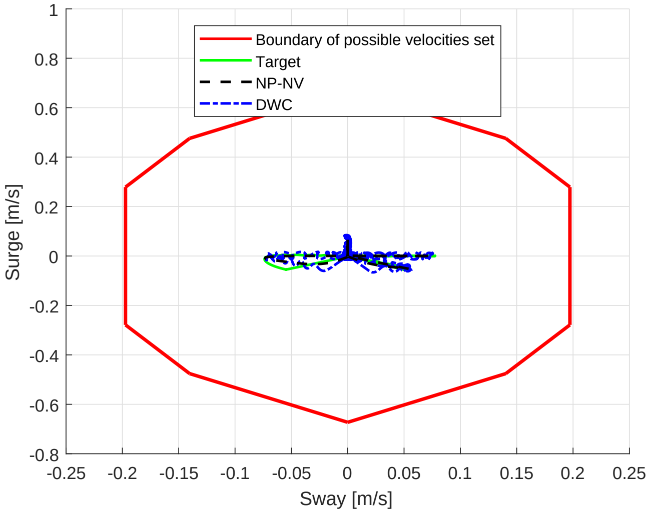


Figure 4.42: Sway-surge window for the 3 DOF DWC.

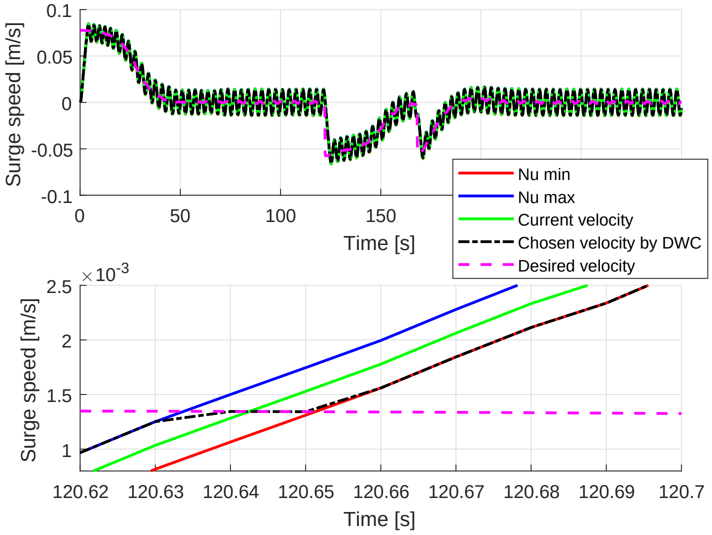


Figure 4.43: Tracking of the desired surge speed $\alpha_u$ for the DWC with a system time delay, where the upper and lower limits of the velocity window are included. The top figure represents the whole 4-corner test, while the bottom is zoomed in for illustrative purposes.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | **44.720** | 44.892 |
| **IAEW** $e_1$ | **321.865** | 702.865 |
| **IAE** $e_2$ | **13.202** | 28.236 |
| **IAEW** $e_2$ | **95.015** | 442.094 |
| **IADC** | **80.921** | 1757.4 |

Table 4.17: Performance metrics for the simulation with the DWC and the NP-NV controller with estimation noise.

Figure 4.44 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. Here it is shown that, when under the effect of a combination of disturbances, the NP-NV controller is both more accurate and energy efficient than the DWC,

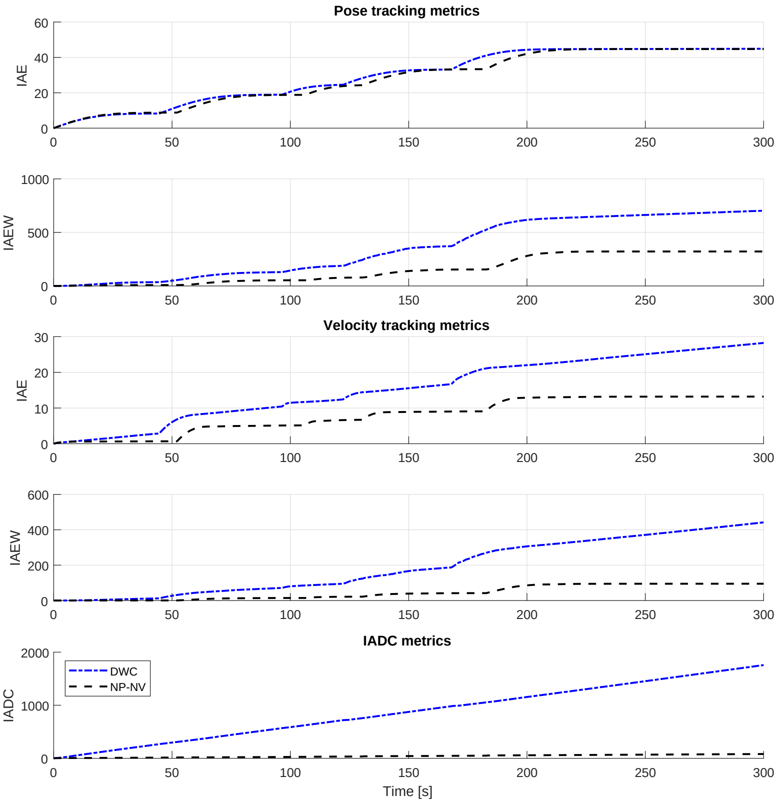


Figure 4.44: IAE, IAEW and IADC for pose tracking for the DWC and the NP-NV controller with all disturbances.

but that the DWC still has a faster transient response. The final values for the performance metrics are summarized in Table 4.17, where the best performing controller is marked in bold. Comparing the values to the ones in Table 4.13 for the case of 0% model uncertainties, it can be seen that the performance of the NP-NV controller has not been severely reduced, except for in the IADC metric. It can be concluded that the NP-NV controller is the most robust controller of the two, and that it has the overall best performance and ability to handle a combination of disturbances, which one can expect to be present in real life environments.

## 4.6 Comparison

The different control algorithms were tested in simulations and evaluated by different performance metrics. In the main simulation results, where no disturbances or uncertainties were present, it was shown that the best performing cascaded feedback controller was the NP-NV-controller, proving that the included nonlinearities indeed have a positive impact on the overall performance. The different cascaded feedback controllers were also tested with constraint handling methods, but in a simulation environment, which can be thought of as an optimal environment, the included saturation effects only slowed the vessel down in its 4-corner test. It is therefore necessary to test the performance of the different controllers in the MC-Lab in order to get a better picture on how including the different constraint handling method will impact the performance.

The extended simulation results, which were included in order to investigate how different types of disturbances could affect the performance of the DWC and the cascaded feedback controllers, showed that its expectable that the DWC will have performance issues in laboratory experiments as it is not currently designed to handle uncertain noise and time delays while running at such a high frequency. The NP-NV controller proved to be a much more robust controller, handling the simulated disturbances without much change in overall performance.

## 4.7 Chapter summary

In this chapter, the 4-corner test was presented along with three different performance metrics. The different control algorithms were tested with and without constraint handling methods, and in environments with and without external disturbances, time delays and model uncertainties. A further discussion about the combination of the simulation results and the experimental results presented in Chapter 5, is presented in Chapter 6.

# Chapter 5

# Experimental results

In this chapter, experimental results from the Marine Cybernetics Laboratory will be presented. During the project, two full weeks were spent in the laboratory, in addition to several other shorter visits used for testing of new implementations between the two main laboratory weeks. As a part of the process of the work done in this thesis, the results from the two weeks will be presented and discussed separately. Many important changes were done to the design and implementations of the different controllers between the laboratory weeks. These changes were based on the results from the first week, and includes changes done to both the laboratory implementation, the ship model and the controllers.

## 5.1 Marine Cybernetics Laboratory

The Marine Cybernetics laboratory is a small ocean basin laboratory at the Department of Marine Technology at NTNU. Due to its relatively small size and advanced instrumental package, the facility is especially suited for tests of motion control system for model-scale surface vessels, but is also suitable for more specialized hydrodynamic tests due to the advanced towing carriage, which has capability for precise movement of models up to six degrees of freedom for both surface ships and submersibles. The basin measures $40\,[m] \times 6.45\,[m] \times 1.5\,[m]$ in length, breadth and depth, respectively, and is displayed in Figure 5.1.

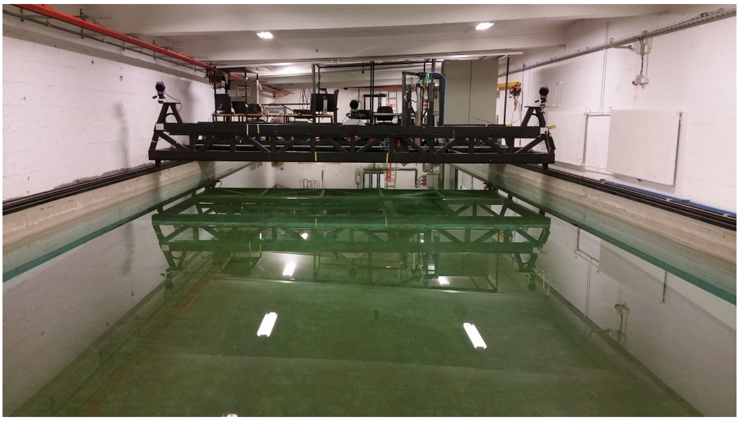


Figure 5.1: MC-Lab basin [32].

### 5.1.1 Laboratory hardware

The lab is equipped with a Qualisys Track Manager (QTM) system for motion capture, which is used for position measurements for the on-board control system. The inputs to the Qualisys system come from three Oqus high-speed infrared (IR) cameras, which tracks the IR reflector orbs fitted on the model-scale ships. The QTM system is installed on a dedicated workstation, using P2P communication with the Oqus cameras.

Experiments can be fully supervised from a control room equipped with a dedicated computer for the QTM system and a TV connected to 2 high-resolution video cameras. The internal communications in the lab are done over IP on a dedicated WLAN network, allowing for wireless control of the model-scale ships, as well as transfer of experimental data from the on-board computer. The ship is equipped with a National Instrument CompactRIO (cRIO) embedded computer system for control computation.

### 5.1.2 Laboratory software

To communicate with the ship, the dedicated laptops are fitted with a substantial software suite, which includes a LabVIEW Full Development System, MATLAB with Simulink package, as well as the National Instruments complete Veristand software package. The full list of software is listed in [32].

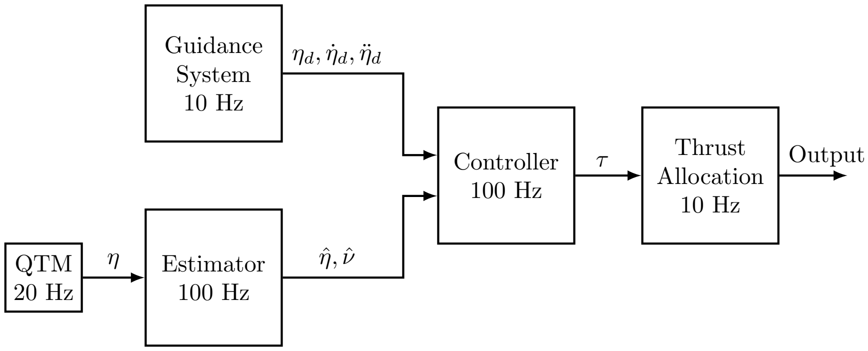


Figure 5.2: MC-Lab block diagram.

While the Qualisys system supplies position measurements, it does not compute velocities. Instead, using the position measurements, the on-board computer estimates body-fixed velocities for control feedback with an applied derivative filter, as seen in Figure 5.2, where the block diagram for the ship system used for the laboratory experiments is shown. Here, it can be seen that the different parts of the control system run at different frequencies. The main system runs at 100 Hz, which includes the controller and the estimator, while the QTM transmits updated position measurements to the control system at 20 Hz. The guidance system and the thrust allocation update the reference target and the commanded actuator inputs, respectively, at a frequency of 10 Hz. This is the main setup for the experiments, and will be used throughout the experiments unless anything else is stated.

## 5.2 Estimator

During laboratory week 1, it became clear that the estimator was not designed to handle a poorly calibrated camera system, giving spikes in the estimated velocity signal, and thus affecting the controller performance. This can be observed for all the experiments throughout laboratory week 1. The poor calibration resulted in that the position measurement occasionally froze, leading the control system to believe that the ship was stationary. When the signal freeze was over, the measured position would be updated, effectively giving a huge rise in the estimated velocity for this time step, which affected the overall performance. Before laboratory week 2, it was then necessary to redesign the estimator to be fault tolerant to improve performance.

### 5.2.1 Fault tolerant estimator design

The initial velocity estimator for the laboratory implementation of CSAD was implemented as a derivative filter. The first step towards handling the signal spikes was to limit the allowed accelerations in the estimator. The allowed accelerations were determined through velocity tests with CSAD in the MC-Lab basin, conducted between the two main laboratory weeks. Maximum thrust in each DOF was used in the experiments, which provided the acceleration limits for CSAD:

$$|a_{u,max}| = 0.13\, m/s^2 \tag{5.1}$$

$$|a_{v,max}| = 0.0267\, m/s^2 \tag{5.2}$$

$$|a_{r,max}| = 0.0052\, rad/s^2. \tag{5.3}$$

Since the estimator is running at 100Hz, these limits were then scaled to fit the acceleration limits per time step, $t_s = 0.01\,[s]$, effectively preventing a higher acceleration than what is physically possible for CSAD to achieve. Note that these acceleration limits are based on the average acceleration from zero to maximum velocity, and thus some tuning of allowed accelerations per time step was needed, where data from the velocity experiments were used as input to an offline simulator used for testing the new estimator design. Suitable values for the acceleration limits used in the estimator were then found to be $|\boldsymbol{a}_{max}| = [0.0011\, m/s^2, 0.0004539\, m/s^2, 0.0007098\, rad/s^2]$.

This change to the estimator was tested, and proved not to affect the overall tracking accuracy of the controller, while at the same time reducing spikes in the estimator, efficiently reducing actuator wear and tear. However, it deemed necessary to further improve upon the estimator's fault tolerance, as it was desired to reduce the estimator spikes as much as possible, and this was not an efficient enough method on its own. Thus, more effort was put into further improving the fault tolerance of the estimator.

The spikes occur when the Qualisys camera system loses the ship, and then transmits the last measured position until the ship is detected again, which usually happens within a few samples. When this happens, the control system will assume that the ship has a constant pose, while the ship is actually moving about in the basin. When the Qualisys system then detects the ship again and updates the measured pose, the estimated pose jumps to the updated pose, resulting in a sudden change in the velocity estimate. This phenomenon is shown

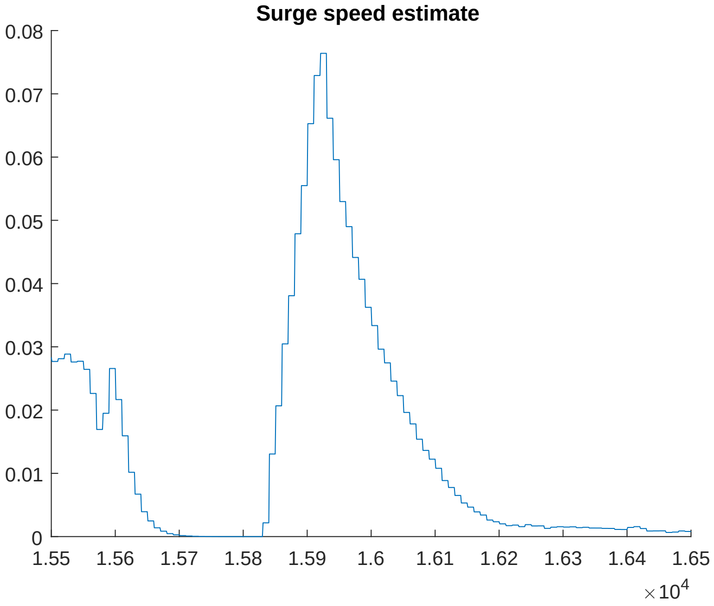


Figure 5.3: A spike in the estimated surge speed, cause by the Qualisys camera system.

in Figure 5.3. It is strongly desirable to avoid the velocity spikes to reduce noise on the commanded actuator inputs.

To handle the camera system's faults, a redesign of the estimator was implemented. Here, the estimator is designed to have two main states of operation; normal operation and lost measurement signal operation. Comparing each new sample to the previous sample, with precision down to the 8th decimal, we can identify frozen measurement signals from the Qualisys system, which runs at 20 Hz. This means that, since the computer on-board the ship, including the estimator, runs at 100 Hz, that every five consecutive samples from the Qualisys system will be identical. However, if the following sample is identical to the previous five, the signal can be assumed to be frozen and lost. A boolean signal is then sent to a switch in the estimator, telling the estimator to switch state until the lost measurement signal is restored. The two estimator states are explained below.

### Normal operation

The velocity estimator is implemented as a derivative filter using the pose measurements $\boldsymbol{\eta}_m = [x_m, y_m, \psi_m]$ as input. For normal operation, this is implemented as

$$\begin{aligned}
\boldsymbol{\eta}_{f,n} &= a\boldsymbol{\eta}_{m,n} + (1-a)\boldsymbol{\eta}_{f,n-1} && (5.4)\\
\dot{\boldsymbol{\eta}}_{\boldsymbol{m,n}} &= \frac{\boldsymbol{\eta}_{m,n} - \boldsymbol{\eta}_{m,n-1}}{t_s} && (5.5)\\
\dot{\boldsymbol{\eta}}_{f,n} &= b\dot{\boldsymbol{\eta}}_{m,n} + (1-b)\dot{\boldsymbol{\eta}}_{f,n-1} && (5.6)\\
\hat{\boldsymbol{\nu}}_{f,n} &= \boldsymbol{R}^{\top}(\psi_{m,n})\dot{\boldsymbol{\eta}}_{f,n}, && (5.7)
\end{aligned}$$

where $\boldsymbol{\eta}_f$ is the estimated pose, $\dot{\boldsymbol{\eta}}_m$ is the derivative of the measured pose calculated using numerical derivation, $\dot{\boldsymbol{\eta}}_f$ is the derivative of the estimated pose, $\boldsymbol{R}$ is a rotation matrix as in (2.3), $\psi_m$ is the measured heading, $\hat{\boldsymbol{\nu}}_f$ is the body-fixed estimated velocity, $t_s$ is the sample time, and $a, b \in \mathbb{R}$ are the cut-off parameters for the filter. Here, $a = 0.08$ and $b = 0.05$. When the boolean signal indicate that the Qualisys measurement signal is lost, it switches state to the lost measurement signal operation.

### Lost measurement signal operation

Here, the estimator keeps the last estimated velocity as $\hat{\boldsymbol{\nu}}_{fl} = \hat{\boldsymbol{\nu}}_f$ constant along with the last measured heading as $\psi_{ml} = \psi_m$, and uses it to estimate the pose $\hat{\boldsymbol{\eta}}_l$ until the lost measurement signal is restored.

$$\hat{\boldsymbol{\eta}}_{l,n} = \int_{t_{s,n-1}}^{t_{s,n}} \boldsymbol{R}(\psi_{ml})\hat{\boldsymbol{\nu}}_{fl} d\sigma + \hat{\boldsymbol{\eta}}_{l,n-1} \quad (5.8)$$

$$(5.9)$$

where $\hat{\boldsymbol{\eta}}_l$ is initialized using the value of $\hat{\boldsymbol{\eta}}_f$ from the previous sample where the measurement signal was not considered lost

$$\hat{\boldsymbol{\eta}}_{l,n-1} = \hat{\boldsymbol{\eta}}_{f,n-1}. \quad (5.10)$$

Then, when the lost measurement signal is restored, the estimator switches back to the state of normal operation, and is initialized by

$$\begin{aligned}
\boldsymbol{\eta}_{f,n-1} &= \hat{\boldsymbol{\eta}}_{l,n-1} && (5.11)\\
\dot{\boldsymbol{\eta}}_{f,n-1} &= \boldsymbol{R}(\psi_{ml})\hat{\boldsymbol{\nu}}_{fl}. && (5.12)
\end{aligned}$$

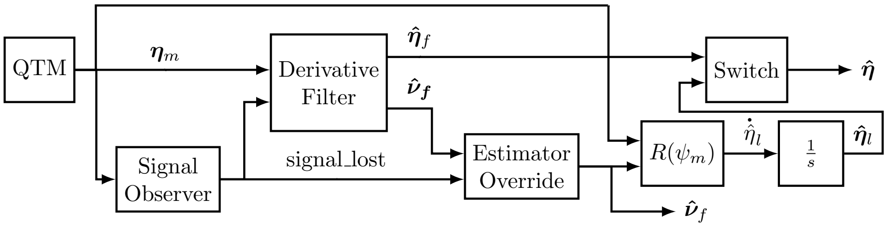

QTM
$\boldsymbol{\eta}_m$
Derivative Filter
$\hat{\boldsymbol{\eta}}_f$
$\hat{\boldsymbol{\nu}}_f$
Switch
$\hat{\boldsymbol{\eta}}$
Signal Observer
signal_lost
Estimator Override
$R(\psi_m)$
$\dot{\hat{\boldsymbol{\eta}}}_l$
$\frac{1}{s}$
$\hat{\boldsymbol{\eta}}_l$
$\hat{\boldsymbol{\nu}}_f$

Figure 5.4: Block diagram for the updated velocity estimator.

The block diagram for the updated estimator is shown in Figure 5.4. This design ensures a fault tolerant estimator, which removes most of the spikes caused by the lost Qualisys measurement signals. This revised estimator was used for laboratory week 2, and we can see a great improvement in the commanded actuator input plots compared to the same plots from laboratory week 1.

## 5.3 Experimental details

The controller gains and the actuator constraints for the different experiments will be presented along with the results. For convenience, the control laws for the different controllers are repeated here:

**NP-LV**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{5.13}$$

$$\dot{\boldsymbol{\alpha}} = -\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1 \tag{5.14}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{\Gamma}_1\boldsymbol{z}_2, \tag{5.15}$$

**NP-NV**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{5.16}$$

$$\dot{\boldsymbol{\alpha}} = -\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1 \tag{5.17}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2, \tag{5.18}$$

**DWC-FB**:

$$\boldsymbol{\alpha} = -\boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{5.19}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2, \tag{5.20}$$

where $\boldsymbol{\alpha}_{DWC}$ and $\dot{\boldsymbol{\alpha}}_{DWC}$ are chosen by the dynamic window algorithm, and DWC-FB refers to the dynamic window-based controller with a nonlinear feedback term included. Note that these are modified for setpoint tracking where $\ddot{\boldsymbol{\eta}}_t = \dot{\boldsymbol{\eta}}_t = 0$.

The experimental tests will be conducted through the 4-corner test, as presented in Section 4.1. Unless something else is stated, the experiments are conducted with setpoint tracking.

## 5.4 Experimental results - Laboratory week 1

The main focus of the first laboratory week was to test the performance of the NP-LV and NP-NV controllers with constraint handling methods, and to compare the performance of these methods. In addition to this, it was desired to the test the performance of the dynamic window-based controller and compare its performance to the other controllers.

During this week, it became clear that there were several modeling mistakes done in both the thrust allocation and ship model, as the simulated model performance did not correspond well with the laboratory behaviour of CSAD. Because of this, in addition to the tests presented below, several other tests were conducted to collect data which could be used to redesign the mathematical model of CSAD and the thrust allocation model.

In addition to this, the estimator, as explained in more detail above, did not handle signal faults, and thus caused several spikes in the commanded control signal $\boldsymbol{\tau}$. Several tests were conducted, and extensive work was put into finding a proper solution for this problem.

The experiments are conducted under the following conditions: In the experiments, the actual model-scale ship's $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ matrices will differ somewhat from those used in the controllers. Also, measurement noise is present in the Qualisys motion tracking system used in the laboratory.

### 5.4.1 NP-LV with constraint handling methods

Here, the experimental results for the NP-LV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this experiment are shown in Table 5.1. Originally, the magnitude constraints were given by $\boldsymbol{m} = [1.1089, 0.6387, 0.6176]$, based on simulations done with the thrust allocation model. Based on these constraints, the rate constraints $\boldsymbol{r}$ were designed such that the vessel would be able to go from zero to max thrust in $[0.6, 0.6, 0.8]$ [s] for surge, sway and yaw, respectively. Keeping the ratio between the magnitude constraints $\boldsymbol{m}$, the gain matrix $\boldsymbol{K}$ was then designed. Because it became clear that the magnitude constraints $\boldsymbol{m}$ were set too low, they were upscaled as shown in Table 5.1, while the other values were kept the same.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\Vert-$ | $-\Vert-$ |
| $\boldsymbol{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\Vert-$ | $-\Vert-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\Vert-$ | $-\Vert-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $-$ | $-$ |
| $\boldsymbol{m}$ | $-$ | $[2, 1.5, 1]$ | $-\Vert-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[1.85, 1.06, 0.77]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[4, 2.31, 2.23]$ |

Table 5.1: Controller gains and actuator constraint parameters for the experiments with the NP-LV controller.

In Figure 5.5, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained NP-LV-controlled ship, the dash-dotted black outline represents the magnitude constrained NP-LV-controlled ship, the dashed green outline represents the NP-LV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

Figure 5.6 shows the commanded control inputs for the three different controllers. It can be seen that the NP-LV with MRS effects handles the spikes generated by the faulty estimator better than the two other controllers, resulting in a more smooth performance, which is beneficial for actuator wear and tear and energy efficiency.

Figure 5.7 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers.

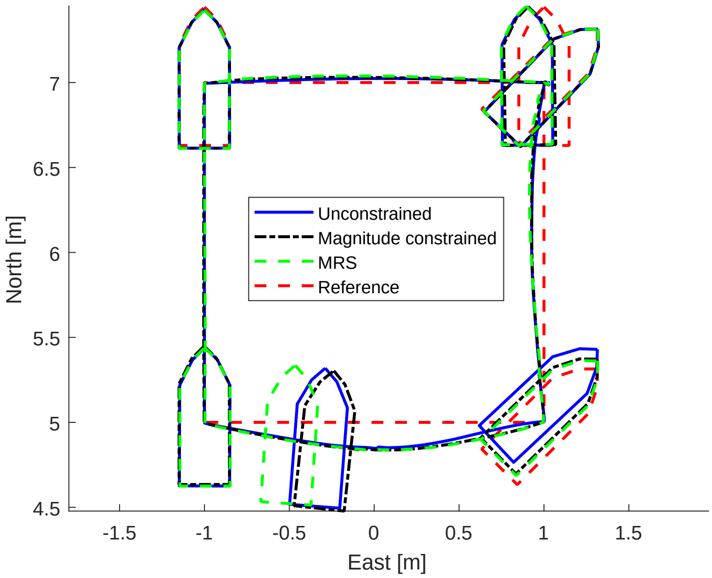


Figure 5.5: 4-corner path plot for the NP-LV controller with and without actuator constraint handling methods.

The controllers have similar performance, but with respect to pose and velocity tracking, the magnitude constrained NP-LV-controller has the best overall performance, which shows that limiting the actuator magnitude constraints can not only be positive for the case of energy efficiency, but also for performance accuracy. However, using the IADC metric, we can see that that the NP-LV controller with MRS effects achieves nearly the same performance as the magnitude constrained NP-LV controller while having less usage of the actuators, which is beneficial. That the NP-LV controller with MRS effects is slower and less accurate than the other controllers, is to be expected as we have introduced designed rate constraints for the system. Based on this, one can thus conclude that the NP-LV-controller with MRS effects has the overall best and desired performance. The final values for the performance metrics are summarized in Table 5.2, where the best performing controller is marked in bold.

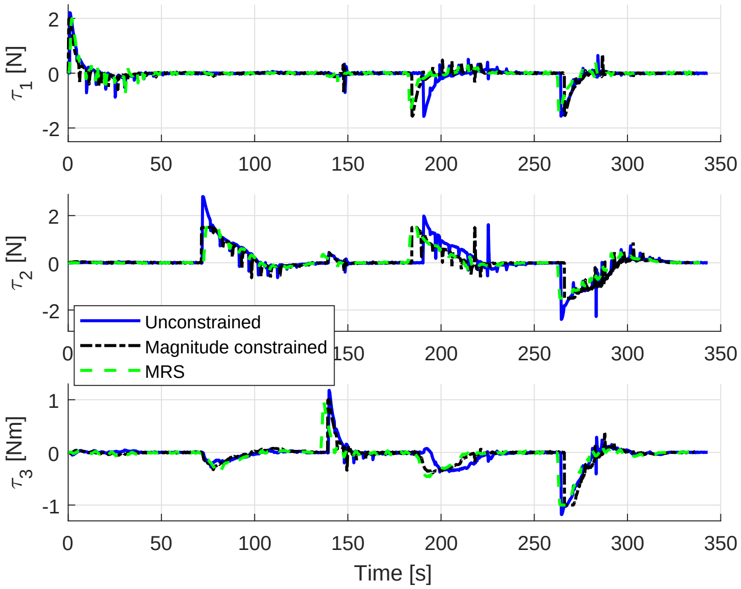


Figure 5.6: Commanded actuator inputs for the NP-LV controller with and without actuator constraint handling methods.

| | **Unconstrained** | **Mag. const.** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | 92.996 | **90.964** | 92.854 |
| **IAEW** $e_1$ | 410.115 | **366.085** | 382.230 |
| **IAE** $e_2$ | 26.040 | **25.190** | 26.141 |
| **IAEW** $e_2$ | 114.837 | **101.377** | 107.609 |
| **IADC** | 95.999 | 105.382 | **58.731** |
| **Time** [s] | 343 | **336** | 339 |

Table 5.2: Performance metrics for the experiments with the NP-LV controller.

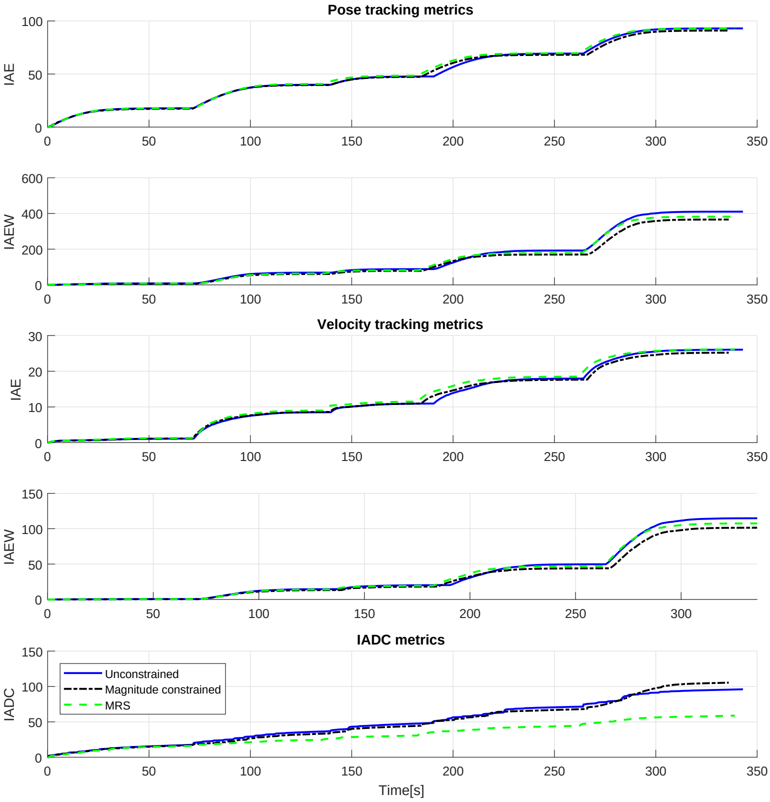


Figure 5.7: IAE, IAEW and IADC for the NP-LV controller with and without actuator constraint handling methods.

### 5.4.2 NP-NV with constraint handling methods

Here, the experimental results for the NP-NV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this experiment are shown in Table 5.3.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\boldsymbol{\Gamma_1}$ | diag([0.08, 0.08, 0.0698]) | $-\|\|-$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | diag([0.2, 0.2, 0.1745])$\boldsymbol{M}$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-\|\|-$ | $-\|\|-$ |
| $\boldsymbol{m}$ | $-$ | $[2, 1.5, 1]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[1.85, 1.06, 0.77]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[4, 2.31, 2.23]$ |

Table 5.3: Controller gains and actuator constraint parameters for the experiments with the NP-NV controller.

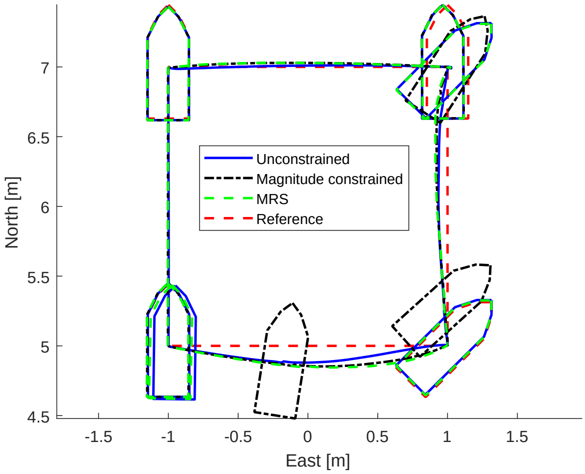


Figure 5.8: 4-corner path plot for the NP-NV controller with and without actuator constraint handling methods.

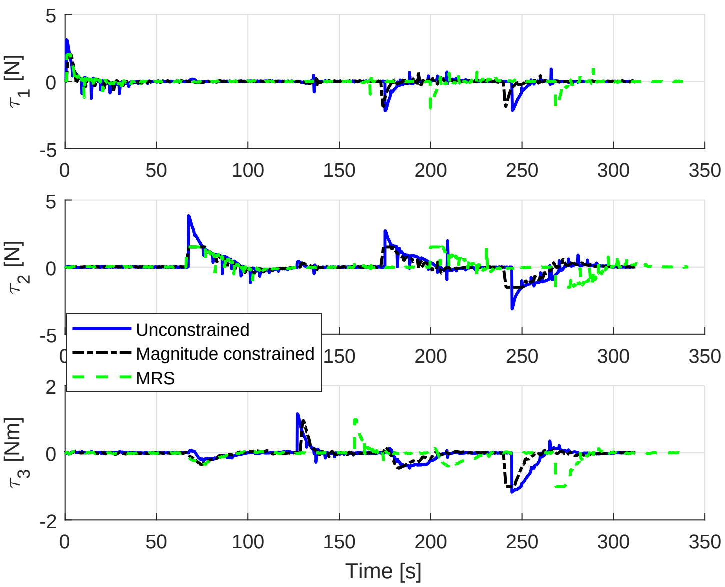


Figure 5.9: Commanded actuator inputs for the NP-NV controller with and without actuator constraint handling methods.

In Figure 5.8, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained NP-NV-controlled ship, the dash-dotted black outline represents the magnitude constrained NP-NV-controlled ship, the dashed green outline represents the NP-NV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

Figure 5.9 shows the commanded control inputs for the three different controllers. It can be seen that the NP-NV controller with MRS effects handles the spikes generated by the faulty estimator better than the two other controllers, resulting in a more smooth performance, which is beneficial for actuator wear and tear and energy efficiency, and that the magnitude constrained NP-NV-controlled ship is slower than the other controllers, because it needed more time for setpoint stabilization.

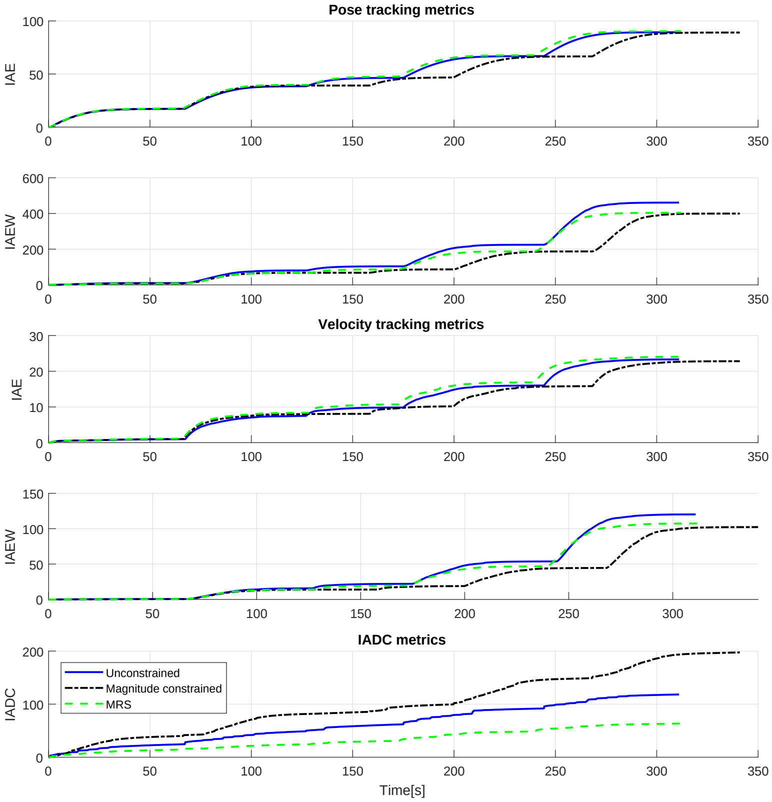


Figure 5.10: IAE, IAEW and IADC for the NP-NV controller with and without actuator constraint handling methods.

Figure 5.10 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The IAE and IAEW metrics show that the controllers have similar performance, but that the magnitude constrained NP-NV-controlled ship is slower without much increase in performance. Using the IADC metric, we can see that the NP-NV controller with MRS effects achieves nearly the same performance as the magnitude constrained NP-LV controller while using less time and having less usage of the actuators. Based on this, one can thus conclude that the NP-NV-controller with MRS effects has the overall best and desired performance. The final values for the performance metrics are summarized in Table 5.4, where the best performing controller is marked in bold.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | 89.396 | **89.086** | 90.382 |
| **IAEW** $e_1$ | 460.084 | **399.465** | 403.793 |
| **IAE** $e_2$ | 23.322 | **22.817** | 24.043 |
| **IAEW** $e_2$ | 120.239 | **102.314** | 107.415 |
| **IADC** | 118.260 | 197.500 | **63.327** |
| **Time [s]** | **311** | 341 | 312 |

Table 5.4: Performance metrics for the experiments with the NP-NV controller.

### 5.4.3 NP-LV with different MRS effects

The gains and actuator constraint parameters for this experiment are shown in Table 5.5. Here, the actuator constraint parameters for the case of a high valued gain matrix $\boldsymbol{K}$ are chosen as in Section 5.4.1, while for the case of a low valued gain matrix $\boldsymbol{K}$, $K_{1,1}$ is chosen as $K_{1,1} = 0.8$. The controller gains are chosen as in Section 3.1.4.

In Figure 5.11, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the NP-NV-controlled ship with MRS effects with a high valued gain matrix $\boldsymbol{K}$, the dash-dotted black outline represents the NP-NV-controlled ship with MRS effects with a low valued gain matrix $\boldsymbol{K}$, while the dashed red outline represents the 4-corner target reference.

| | **High $K_{1,1}$-gain** | **Low $K_{1,1}$-gain** |
|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $-$ |
| $\boldsymbol{m}$ | $[2, 1.5, 1]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $[1.85, 1.06, 0.77]$ | $-\|\|-$ |
| $\boldsymbol{K}$ | $[4, 2.31, 2.23]$ | $[0.8, 0.46, 0.45]$ |

Table 5.5: Controller gains and actuator constraint parameters for the experiments with the NP-LV controller with different MRS effects.

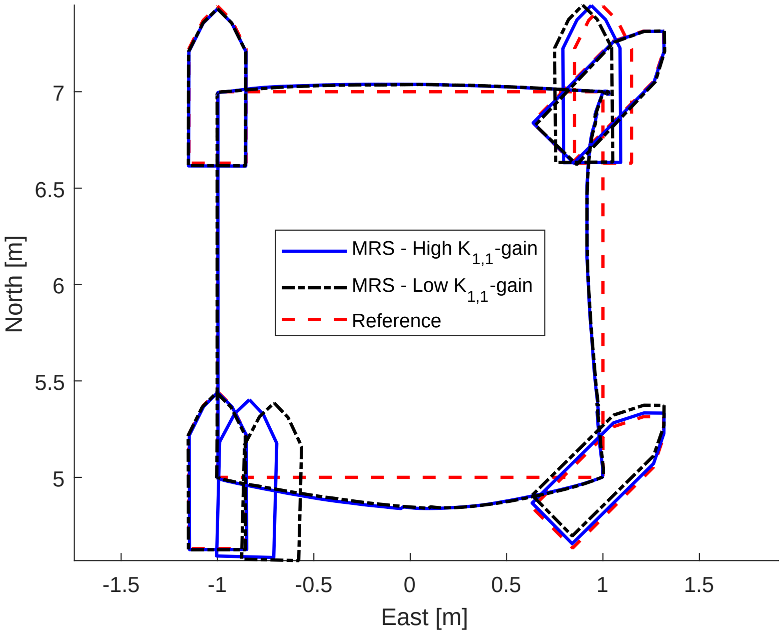


Figure 5.11: 4-corner path plot for the NP-LV controller with different MRS effects.

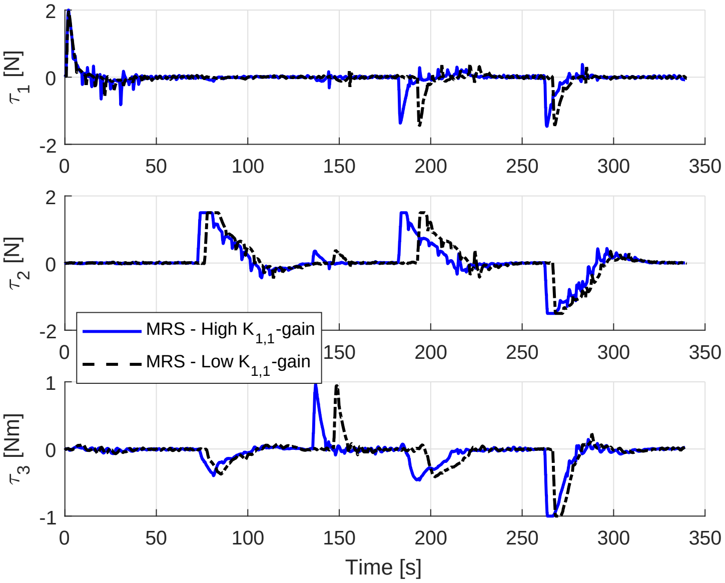


Figure 5.12: Commanded actuator inputs for the NP-LV controller with different MRS effects.

Figure 5.12 shows the commanded control inputs for the two different controllers. It can be seen that the NP-LV controller with a low valued gain matrix $\boldsymbol{K}$ handles the spikes generated by the faulty estimator slightly better than the one with a high valued gain matrix $\boldsymbol{K}$, but with an overall slower performance.

Figure 5.13 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The metrics show that the NP-LV controller with a low valued gain matrix $\boldsymbol{K}$ has better performance for both pose and velocity tracking, along with less use of the actuators, and thus is the best controller. The final values for the performance metrics are summarized in Table 5.6, where the best performing controller is marked in bold.

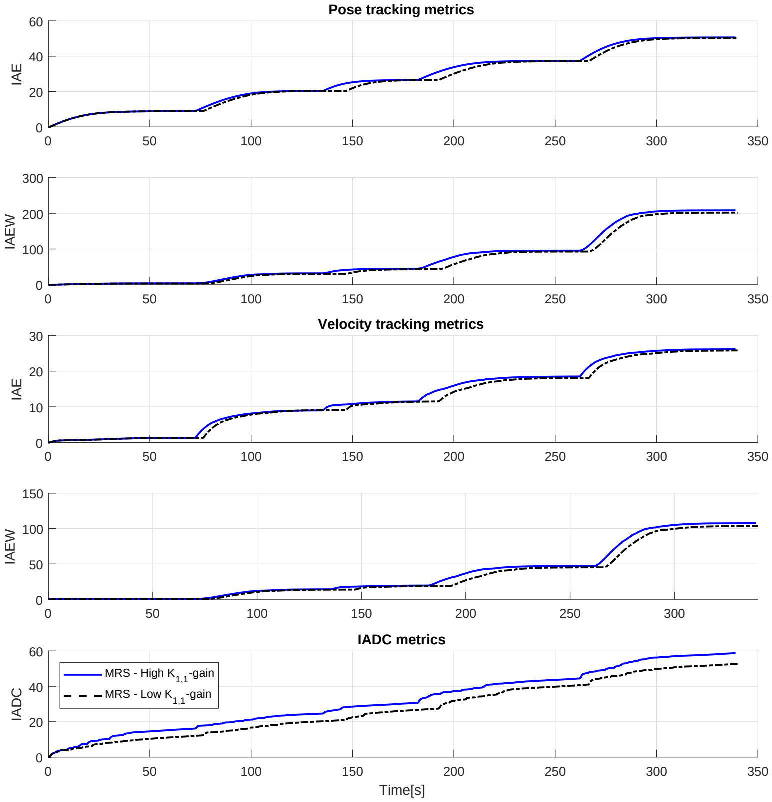


Figure 5.13: IAE and IAEW for pose tracking for the NP-LV controller with different MRS effects.

| | **High $K_{1,1}$-gain** | **Low $K_{1,1}$-gain** |
|---|---|---|
| **IAE $e_1$** | 92.854 | **92.620** |
| **IAEW $e_1$** | 382.230 | **371.852** |
| **IAE $e_2$** | 26.141 | **25.794** |
| **IAEW $e_2$** | 107.609 | **103.559** |
| **IADC** | 58.731 | **52.673** |
| **Time [s]** | **339** | 340 |

Table 5.6: Performance metrics for the experiments with the NP-LV controller with different MRS effects.

### 5.4.4 NP-NV with different MRS effects

The gains and actuator constraint parameters for this experiment are as in Section 5.4.3, and shown in Table 5.5.

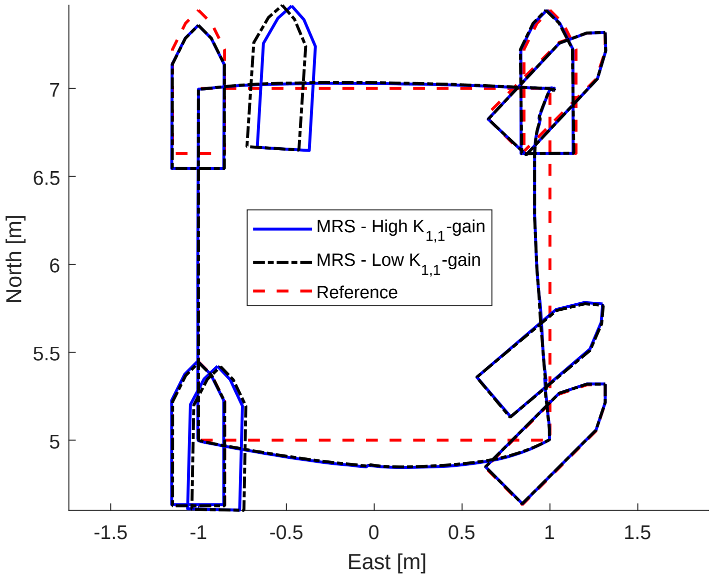


Figure 5.14: 4-corner path plot for the NP-NV controller with different MRS effects.

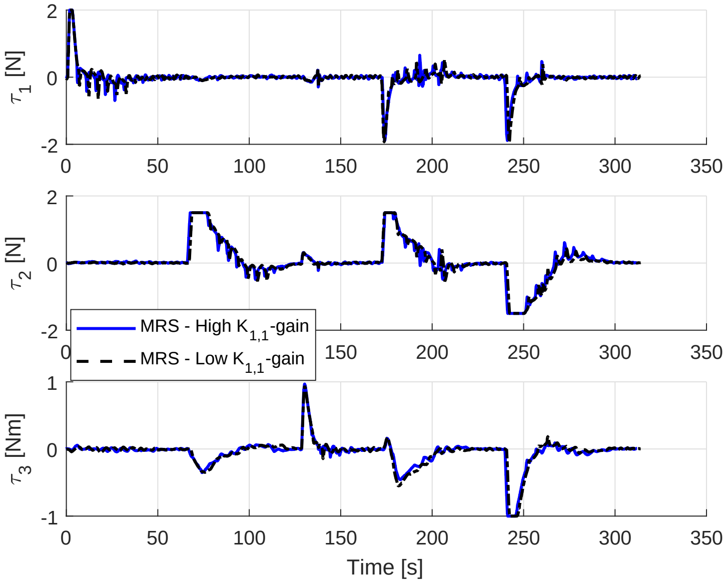


Figure 5.15: Commanded actuator inputs for the NP-NV controller with different MRS effects.

In Figure 5.14, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the NP-NV-controlled ship with MRS effects with a high valued gain matrix $\boldsymbol{K}$, the dash-dotted black outline represents the NP-NV-controlled ship with MRS effects with a low valued gain matrix $\boldsymbol{K}$, while the dashed red outline represents the 4-corner target reference.

Figure 5.15 shows the commanded control inputs for the two different controllers. It can be seen that the NP-LV controller with a low valued gain matrix $\boldsymbol{K}$ has a slightly smoother behaviour, but the difference is, however, not significantly large enough to have a considerable impact on overall performance.

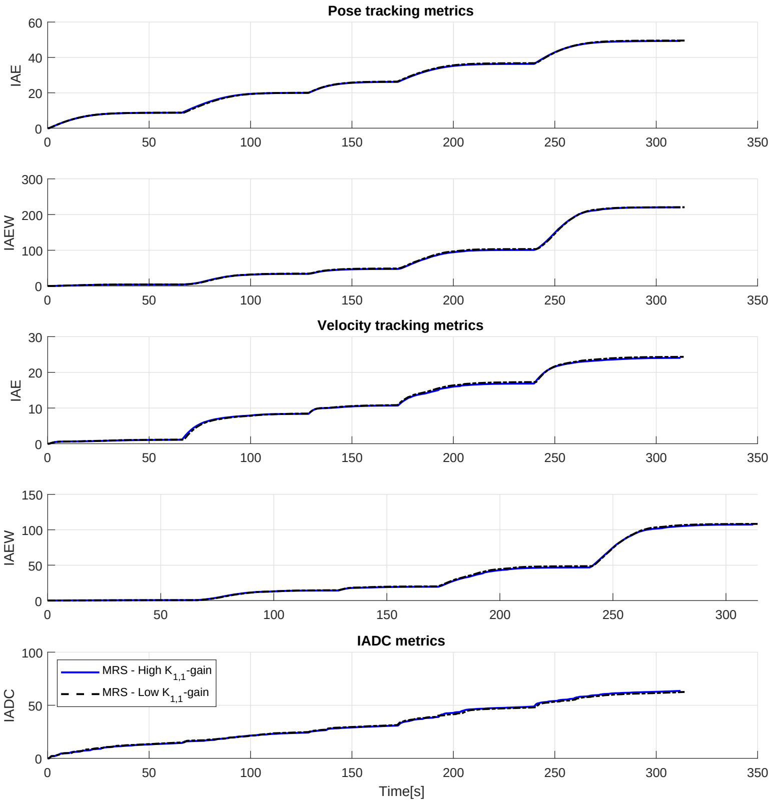


Figure 5.16: IAE and IAEW for pose tracking for the NP-NV controller with different MRS effects.

| | **High $K_{1,1}$-gain** | **Low $K_{1,1}$-gain** |
|---|---|---|
| **IAE $e_1$** | **90.382** | 90.716 |
| **IAEW $e_1$** | 403.793 | **403.763** |
| **IAE $e_2$** | **24.043** | 24.335 |
| **IAEW $e_2$** | **107.414** | 108.313 |
| **IADC** | 63.526 | **62.513** |
| **Time [s]** | **312** | 314 |

Table 5.7: Performance metrics for the experiments with the NP-NV controller with different MRS effects.

Figure 5.16 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The metrics show that the NP-NV controller with a high valued gain matrix $\boldsymbol{K}$ has slightly better pose tracking accuracy, but a marginally higher energy usage. For velocity tracking, the NP-NV controller with a high valued gain matrix $\boldsymbol{K}$ is slightly better in both tracking accuracy and energy efficiency. For IADC, the NP-NV controller with a low valued gain matrix $\boldsymbol{K}$ wins, which is to be expected. Based on this, the NP-NV controller with a high valued gain matrix $\boldsymbol{K}$ is the overall best performing controller in this experiment The final values for the performance metrics are summarized in Table 5.7, where the best performing controller is marked in bold.

### 5.4.5 Comparison

During this laboratory week, the NP-LV and the NP-NV controllers were tested with constraint handling methods. For the NP-LV controller, it was shown that including the MRS model could improve both accuracy and energy efficiency compared to the unconstrained controller. In addition, the actuator usage was almost halved, as shown in the IADC metric. However, only using magnitude saturation effects on the NP-LV-controlled ship proved to be able to further improve accuracy and energy efficiency, but at the cost of a higher actuator wear and tear compared to when the MRS model was used. Furthermore, it was shown that using a low valued gain matrix $\boldsymbol{K}$ could further improve performance for the NP-LV controller with the MRS model included.

For the NP-NV controller, it was shown that including the MRS model could improve energy efficiency, but at a small cost in accuracy. However, with the MRS model included, the actuator usage was nearly halved, as shown in the

IADC metric. Again, only using magnitude saturation effects could further improve both accuracy and energy efficiency, but at the cost of actuator wear and tear, which was more than three times as high for the magnitude constrained NP-NV controller compared to when the MRS model was included. It was also shown that lowering the value of the gain matrix $\boldsymbol{K}$ had more or less no impact on the performance of the NP-NV controller with the MRS model included.

It was also desired to test the DWC during this laboratory week, but due to the observed modeling errors, the DWC needed to be redesigned, and also, as discussed above, the estimator needed to be improved. The test were therefore postponed until the second laboratory week.

## 5.5 Experimental results - Laboratory week 2

The focus of the second laboratory week was to test the performance of the cascaded feedback controllers with constraint handling methods with the updated ship model and estimator design. However, it should be noted that the updated estimator did not successfully remove all of the estimation noise present in the first laboratory week, but that the overall amount of noise was severely reduced. It was also desired to thoroughly test the DWC, and compare its performance to that of the other controllers.

### 5.5.1 NP-LV vs NP-NV

| | **NP-LV** | **NP-NV** |
|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\text{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | $\text{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $[0.7, 1.0]$ |
| $\boldsymbol{m}$ | $-$ | $-$ |
| $\boldsymbol{r}$ | $-$ | $-$ |
| $\boldsymbol{K}$ | $-$ | $-$ |

Table 5.8: Controller gains and actuator constraint parameters for the experiments with the NP-NV and NP-LV controllers.

The gains and actuator constraint parameters for this experiment are shown in Table 5.8. In this experiment, the performance of the unconstrained NP-LV

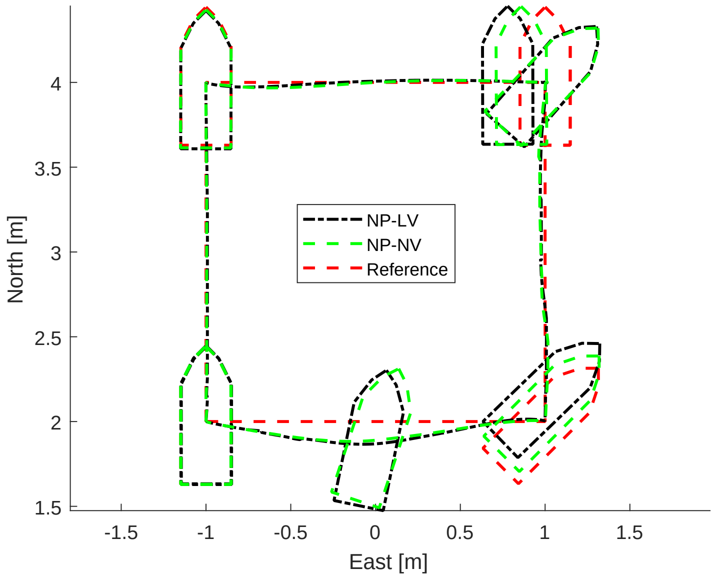


Figure 5.17: 4-corner path plot for the comparison between the NP-NV and NP-LV controllers.

and NP-NV controllers are tested against each other. It was desired to compare these controllers with the updated CSAD model to directly investigate the benefits of using linear and nonlinear velocity feedbacks. Here, the controller gains are chosen as in Section 3.1.4, and as the experiments are done on unconstrained controllers, no actuator limitations are included.

In Figure 5.17, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted black outline represents the unconstrained NP-LV-controlled ship, the dashed green outline represents the unconstrained NP-NV-controlled ship, while the dashed red outline represents the 4-corner target reference.

Figure 5.18 shows the commanded control inputs for the two different controllers. It can be seen that the unconstrained NP-LV controller has a slightly more smooth performance, which is beneficial for actuator wear and tear and energy efficiency.

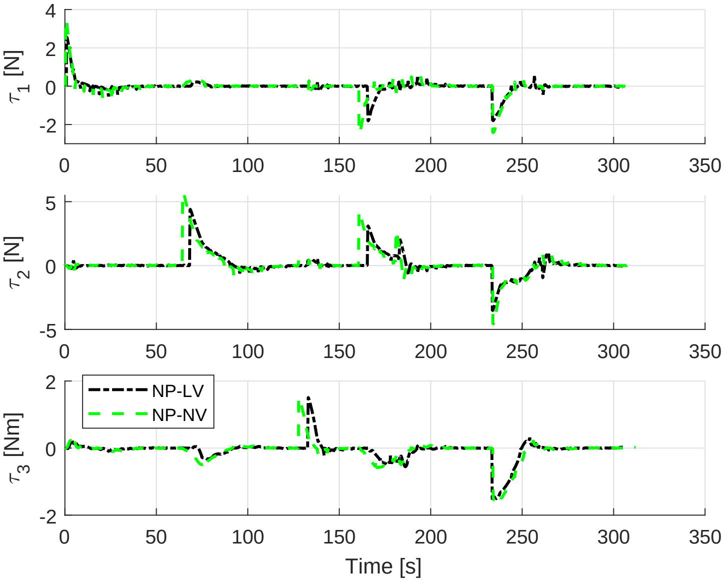


Figure 5.18: Commanded actuator inputs for the comparison between the NP-LV and NP-NV controllers.

Figure 5.19 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The IAE metrics show that the controllers have similar tracking performance for pose and velocity tracking, while the IAEW metrics show that the NP-LV controller uses less energy than its nonlinear counterpart. By the IADC metric, we also see that the NP-LV controller achieves a better overall performance by less usage of the actuators. Based on this, it can be concluded that the NP-LV controller has a better overall performance than the NP-NV controller. The final values for the performance metrics are summarized in Table 5.9, where the best performing controller is marked in bold.

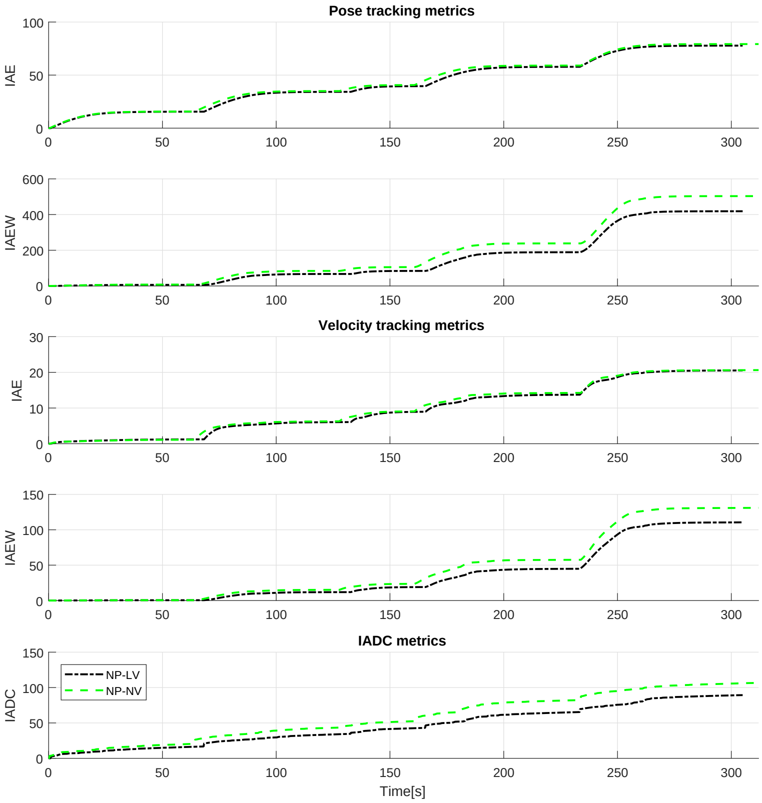


Figure 5.19: IAE, IAEW and IADC for the comparison between the NP-LV and NP-NV controllers.

| | **NP-LV** | **NP-NV** |
|---|---|---|
| **IAE** $e_1$ | **77.776** | 79.238 |
| **IAEW** $e_1$ | **418.619** | 503.311 |
| **IAE** $e_2$ | **20.540** | 20.625 |
| **IAEW** $e_2$ | **110.555** | 131.008 |
| **IADC** | **89.295** | 106.640 |
| **Time [s]** | **305** | 312 |

Table 5.9: Performance metrics for the experiments with the NP-LV and NP-NV controllers.

### 5.5.2 NP-LV with constraint handling methods

Here, experimental results for the NP-LV controller with actuator constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this experiment are shown in Table 5.10. It was desired to compare these controllers with the unconstrained NP-LV controller as well, but unfortunately the data for this laboratory experiment were found to be corrupted.

| | **Mag. constr.** | **MRS** |
|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\text{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | $\text{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $-$ | $-$ |
| $\boldsymbol{m}$ | $[2.88, 1.60, 1.36]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $[2.88, 1.60, 1.36]$ |
| $\boldsymbol{K}$ | $-$ | $[5, 2.78, 2.36]$ |

Table 5.10: Controller gains and actuator constraint parameters for the experiments with the NP-LV controller.

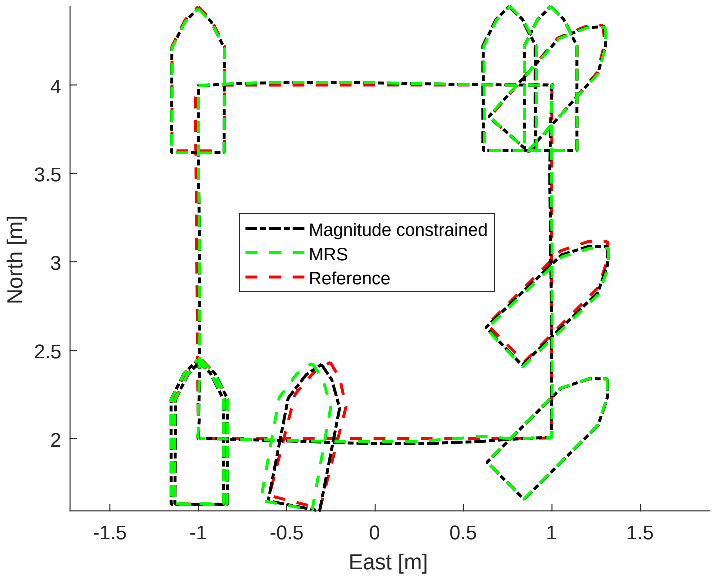


Figure 5.20: 4-corner path plot for the NP-LV controller with actuator constraint handling methods.

This experiment was conducted using trajectory tracking. For convenience, the control law for the NP-LV controller used together with trajectory tracking is repeated here:

$$\boldsymbol{\alpha} = \boldsymbol{R}^\top \dot{\boldsymbol{\eta}}_t - \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 \tag{5.21}$$

$$\dot{\boldsymbol{\alpha}} = \boldsymbol{R}^\top \ddot{\boldsymbol{\eta}}_t + \boldsymbol{S}^\top \boldsymbol{R}^\top \dot{\boldsymbol{\eta}}_t - \dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1 \tag{5.22}$$

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}\boldsymbol{\alpha} + \boldsymbol{D}\boldsymbol{\alpha} - \boldsymbol{\Gamma}_1 \boldsymbol{z}_2. \tag{5.23}$$

In Figure 5.20, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the dash-dotted black outline represents the magntiude constrained NP-LV-controlled ship, the dashed green outline represents the NP-NV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner trajectory target reference.

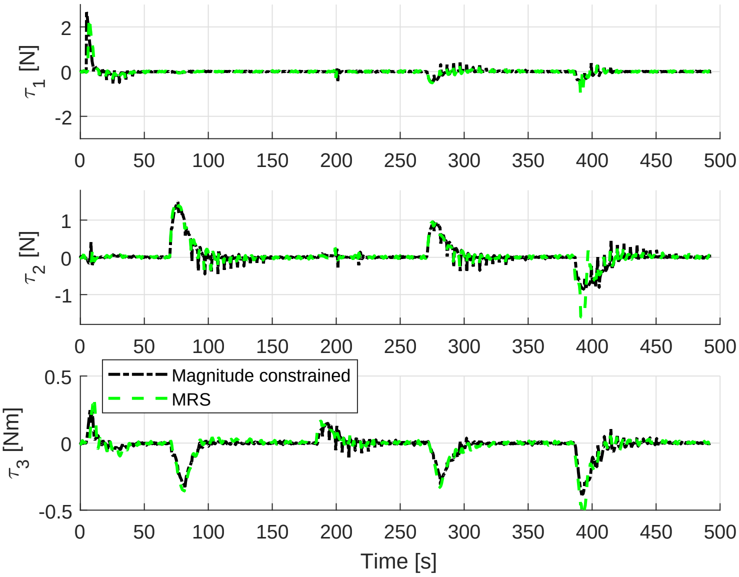


Figure 5.21: Commanded actuator inputs for the NP-LV controller with actuator constraint handling methods.

Figure 5.21 shows the commanded control inputs for the two different controllers. It can be seen that the NP-LV controller with MRS effects has a more smooth performance, which is beneficial for actuator wear and tear.

| | **Mag. constr.** | **MRS** |
|---|---|---|
| **IAE** $e_1$ | **21.3657** | 23.4945 |
| **IAEW** $e_1$ | **50.330** | 54.091 |
| **IAE** $e_2$ | **10.848** | 14.088 |
| **IAEW** $e_2$ | **25.5545** | 32.434 |
| **IADC** | 96.927 | **73.837** |
| **Time** [s] | 499 | **497** |

Table 5.11: Performance metrics for the experiments with the NP-LV controller with actuator constraint handling methods.

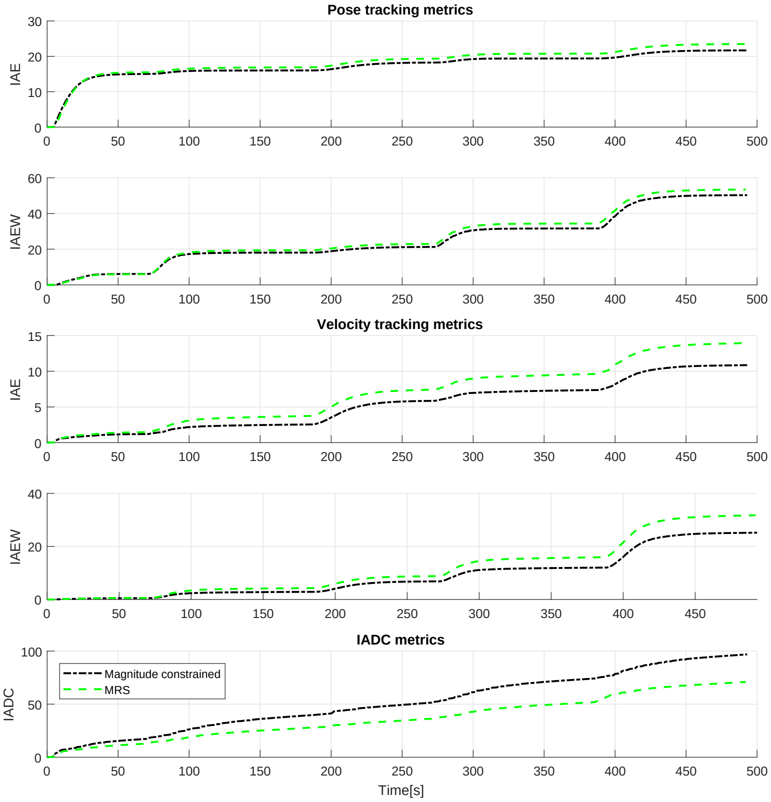


Figure 5.22: IAE, IAEW and IADC for the NP-LV controller with actuator constraint handling methods.

Figures 5.22 show the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. The IAE and IAEW metrics show that the magnitude constrained NP-LV controlled ship has a better pose and velocity tracking performance, together with a higher energy efficiency. The IADC metric show that the NP-LV controlled ship with MRS effects has less use of the actuators, but since the difference in performance from the magnitude constrained NP-LV is quite substantial, the magnitude constrained NP-LV has the best overall performance in this experiment. The final values for the performance metrics are summarized in Table 5.11, where the best performing controller is marked in bold.

### 5.5.3 NP-NV with constraint handling methods

Here, experimental results for the NP-NV controller with and without constraint handling methods will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this experiment are shown in Table 5.12.

| | **Unconstrained** | **Mag. constr.** | **MRS** |
|---|---|---|---|
| $\boldsymbol{\Gamma_1}$ | $\mathrm{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ | $-\|\|-$ |
| $\boldsymbol{\Gamma_2}$ | $\mathrm{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-\|\|-$ | $-\|\|-$ |
| $\boldsymbol{m}$ | $-$ | $[2.88, 1.60, 1.36]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $-$ | $[2.88, 1.60, 1.36]$ |
| $\boldsymbol{K}$ | $-$ | $-$ | $[5, 2.78, 2.36]$ |

Table 5.12: Controller gains and actuator constraint parameters for the experiments with the NP-NV controller with constraint handling methods.

In Figure 5.23, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the unconstrained NP-NV-controlled ship, the dash-dotted black outline represents the magnitude constrained NP-NV-controlled ship, the dashed green outline represents the NP-NV-controlled ship with MRS effects, while the dashed red outline represents the 4-corner target reference.

Figure 5.24 shows the commanded control inputs for the three different controllers. It can be seen that the NP-NV controller with MRS effects handles the

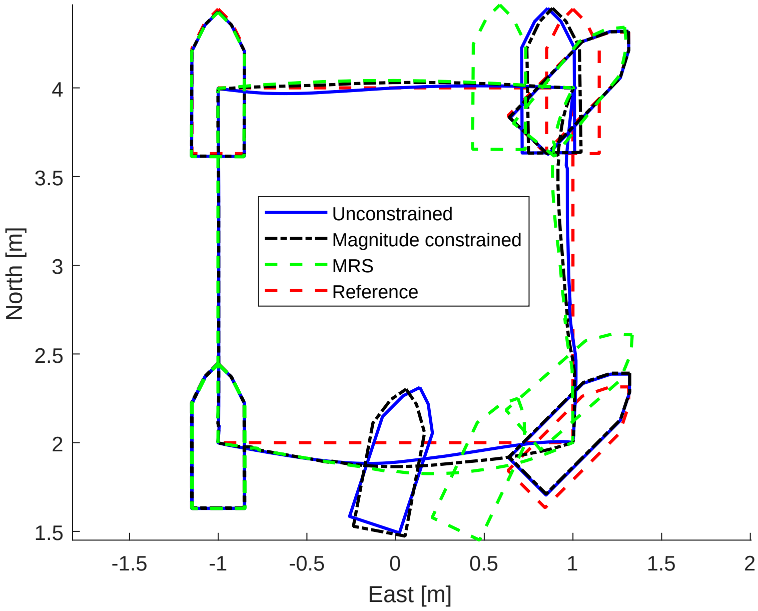


Figure 5.23: 4-corner path plot for the NP-NV controller with and without actuator constraint handling methods.

spikes generated by the estimator better than the two other controllers, resulting in a more smooth performance, which is beneficial for actuator wear and tear.

Figure 5.25 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the three controllers. The metrics show that the controllers have similar pose tracking performance, but that the NP-NV controller with the MRS model included has a less accurate and energy efficient performance in velocity tracking. However, the NP-NV controller with the MRS model included has a much lower value in the IADC metric. Based on this, one can thus conclude that the NP-NV-controller with MRS effects has the overall best and desired performance. The final values for the performance metrics are summarized in Table 5.13, where the best performing controller is noted in bold.

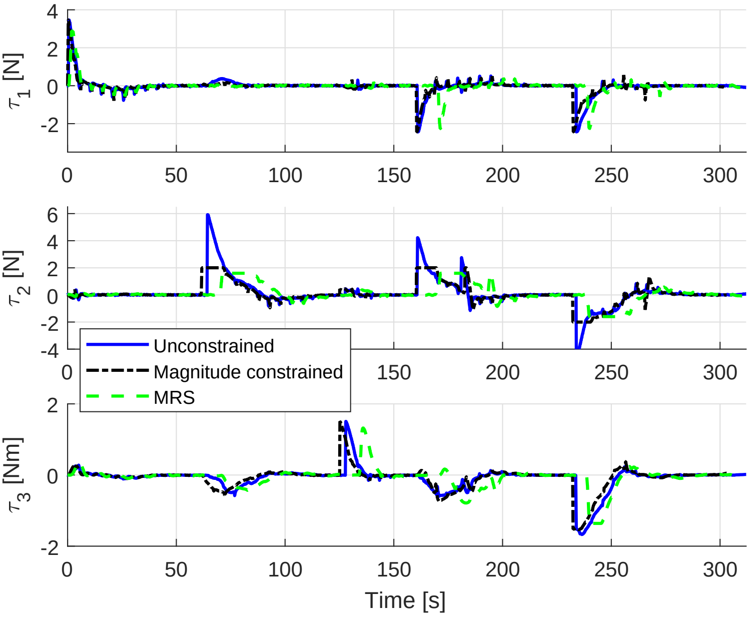


Figure 5.24: Commanded actuator inputs for the NP-NV controller with and without actuator constraint handling methods.

| | **Unconstrained** | **Magnitude constrained** | **MRS** |
|---|---|---|---|
| **IAE** $e_1$ | **79.238** | 80.932 | 85.176 |
| **IAEW** $e_1$ | **503.311** | 504.451 | 508.479 |
| **IAE** $e_2$ | **20.625** | 22.014 | 25.218 |
| **IAEW** $e_2$ | **131.008** | 137.217 | 150.543 |
| **IADC** | 106.640 | 97.807 | **66.105** |
| **Time [s]** | 312 | 303 | **311** |

Table 5.13: Performance metrics for the experiments with the NP-NV controller.

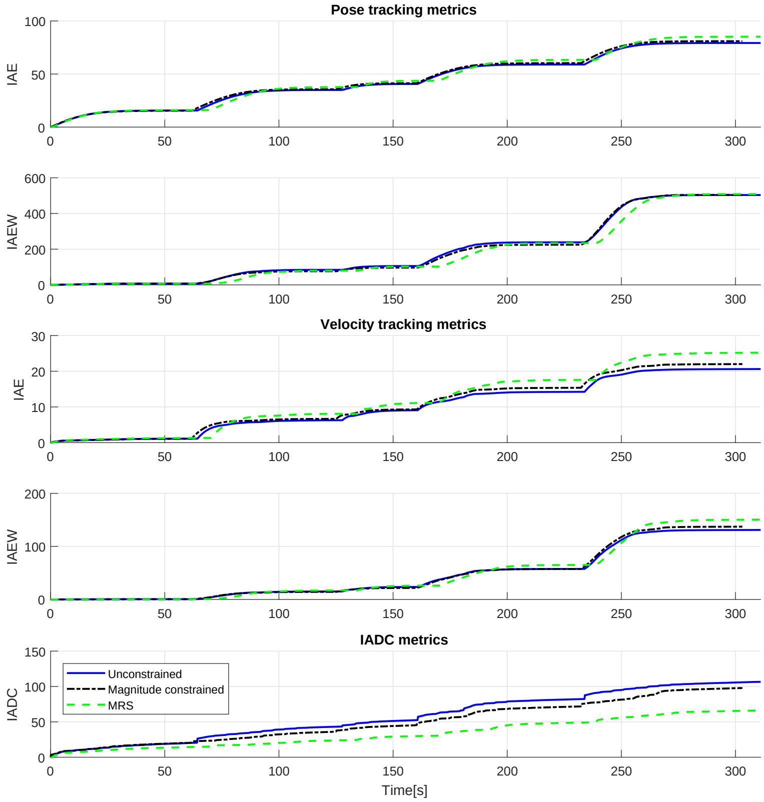


Figure 5.25: IAE, IAEW and IADC for the NP-NV controller with and without actuator constraint handling methods.

### 5.5.4 DWC

Here, experimental results for the experiment with the DWC controller will be presented. The gains, which are chosen by the tuning rules in Section 3.1.4, and actuator constraint parameters for this experiment are shown in Table 5.14. As this is a laboratory experiment where disturbances are present, the experiment with the DWC is conducted with a nonlinear feedback, such as in (3.126) and discussed in Section 3.2.4. The DWC was tested against the magnitude constrained NP-NV controller. Here, the laboratory system in Figure 5.2 was modified such that all the subsystems were running on the same frequency in order to reduced the amount of system time delays.

| | **NP-NV** | **DWC** |
|---|---|---|
| $\mathbf{\Gamma_1}$ | $\text{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\mathbf{\Gamma_2}$ | $\text{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1.0]$ | $-\|\|-$ |
| $\boldsymbol{m}$ | $[3.5996, 2.0003, 1.7027]$ | $-\|\|-$ |
| $\boldsymbol{r}$ | $-$ | $-$ |
| $\boldsymbol{K}$ | $-$ | $-$ |

Table 5.14: Controller gains and actuator constraint parameters for the experiment with the DWC.

In Figure 5.26, the outline of the ship pose is plotted to show the pose motion pattern of the different controllers. Here, the blue outline represents the DWC, the dashed green outline represents the magnitude constrained NP-NV-controlled ship, while the dashed red outline represents the 4-corner target reference.

Figure 5.27 shows the commanded control inputs for the two different controllers. As predicted, because of system disturbances and time delays, the DWC has a oscillating control signal, causing actuator twitching, as in the simulations where disturbances were present. Comparing Figure 5.27 to simulation results in Figure 4.39, it can be seen that the commanded actuator inputs for the DWC have a very similar performance, meaning that the poor laboratory performance most likely is caused by similar disturbances. It is shown in Figure 5.26 that the DWC, similarly to the simulation results in Figure 4.38, still navigates the 4-corner test, but that it has wider transient arches than in the simulations.

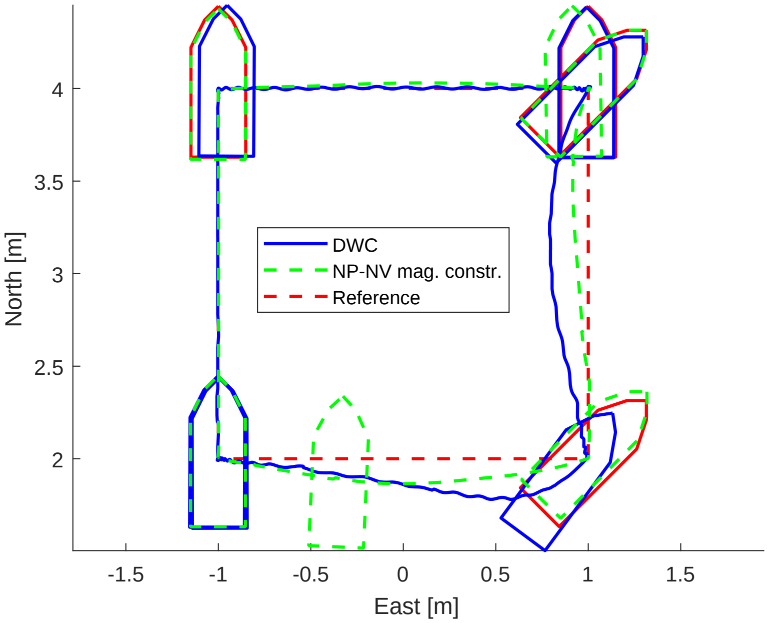


Figure 5.26: 4-corner path plot for the experiment with the DWC.

The same can be seen in Figure 4.40, 4.41 and 4.42, which illustrate how the surge speed, sway speed and yaw rate move in the velocity space in order to track the target through the 4-corner test, where again the performance is similar to that of the simulation. Here, the blue dash-dotted outline represents the DWC, while the green outline represents the target velocity. It can be seen that the disturbances present in the laboratory have a huge impact on the DWC's ability to track the desired velocity.

Figure 5.31 shows the performance metrics IAE and IAEW for pose tracking and velocity tracking, along with the IADC metric, for the two controllers. Here it is shown that, when laboratory disturbances are present and not account for, the NP-NV controller is both more accurate and energy efficient than the DWC, but that the DWC still has a faster transient response. The final values for the performance metrics are summarized in Table 5.15, where the best performing controller is marked in bold.

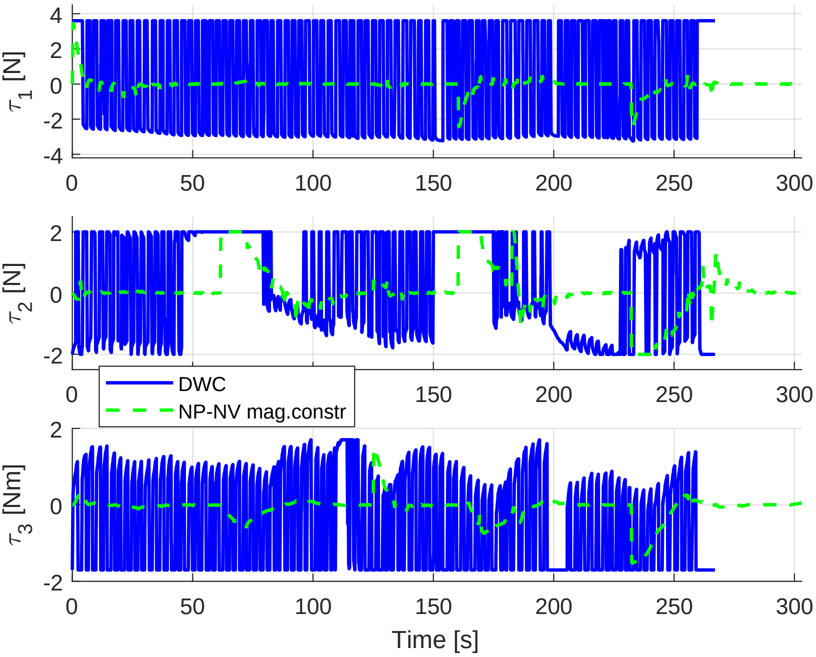


Figure 5.27: Commanded actuator inputs for the experiment with the DWC.

| | **NP-NV** | **DWC** |
|---|---|---|
| **IAE** $e_1$ | **80.932** | 95.700 |
| **IAEW** $e_1$ | **504.451** | 1926.300 |
| **IAE** $e_2$ | **22.014** | 36.800 |
| **IAEW** $e_2$ | **151.217** | 741.600 |
| **IADC** | **97.087** | 1999.800 |
| **Time** [s] | 303 | **268** |

Table 5.15: Performance metrics for the experiment with the DWC.

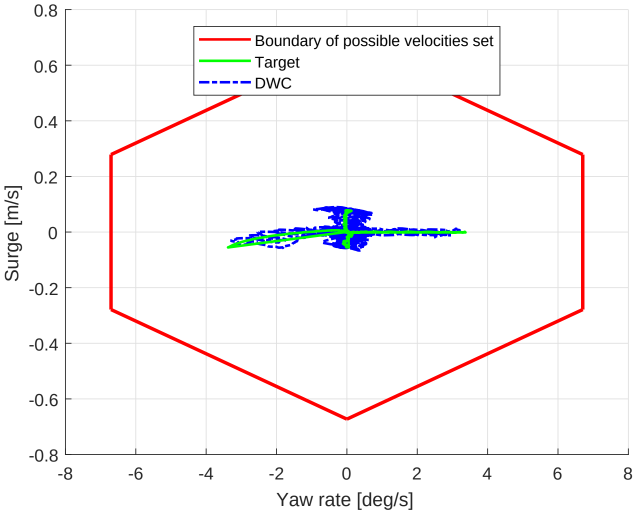


Figure 5.28: Yaw rate-surge speed window for the 3DOF DWC.

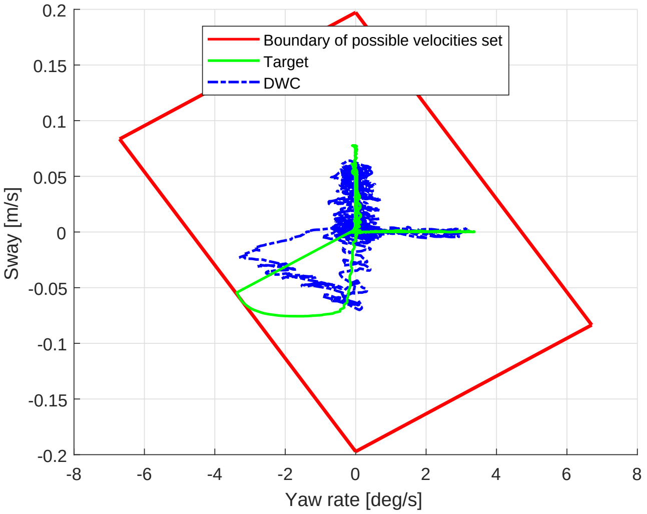


Figure 5.29: Yaw rate-sway speed window for the 3DOF DWC.

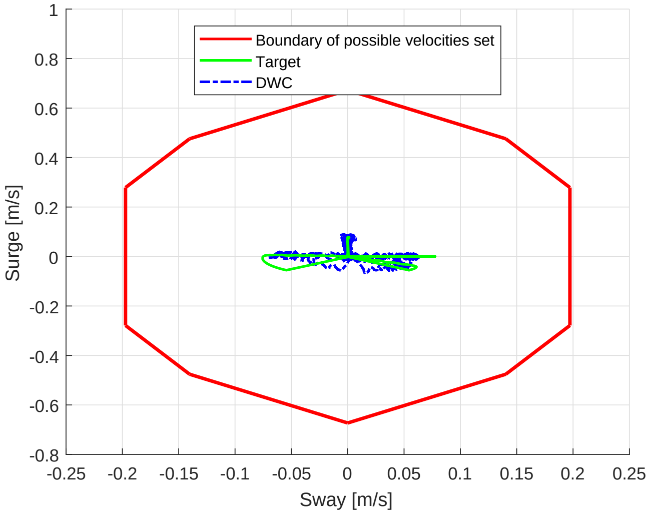


Figure 5.30: Sway-surge speed window for the 3DOF DWC.

### 5.5.5 Comparison

In this week, the NP-LV and the NP-NV controller were tested with the updated ship model and estimator, and with other actuator saturation effects. In addition, the DWC was finally experimentally tested.

First, a direct test with the NP-LV controller against the NP-NV controller was conducted. There it was shown that the NP-LV controller had both better accuracy and energy efficiency, along with less actuator wear and tear.

Then, it was shown that the magnitude constrained NP-LV controller slightly outperformed the NP-LV controller with the MRS model included when using trajectory tracking. However, using the MRS model resulted in almost a 25% reduction in actuator wear and tear.

For the NP-NV controller, similar results could be observed. However, both the magnitude constrained controller and the controller with MRS effects included got outperformed by the unconstrained controller. Possibly this is an indication that the actuator saturation effects were poorly tuned for the experiments this week, and that the saturation effects used in the first week were a better fit for improving the performance of CSAD.

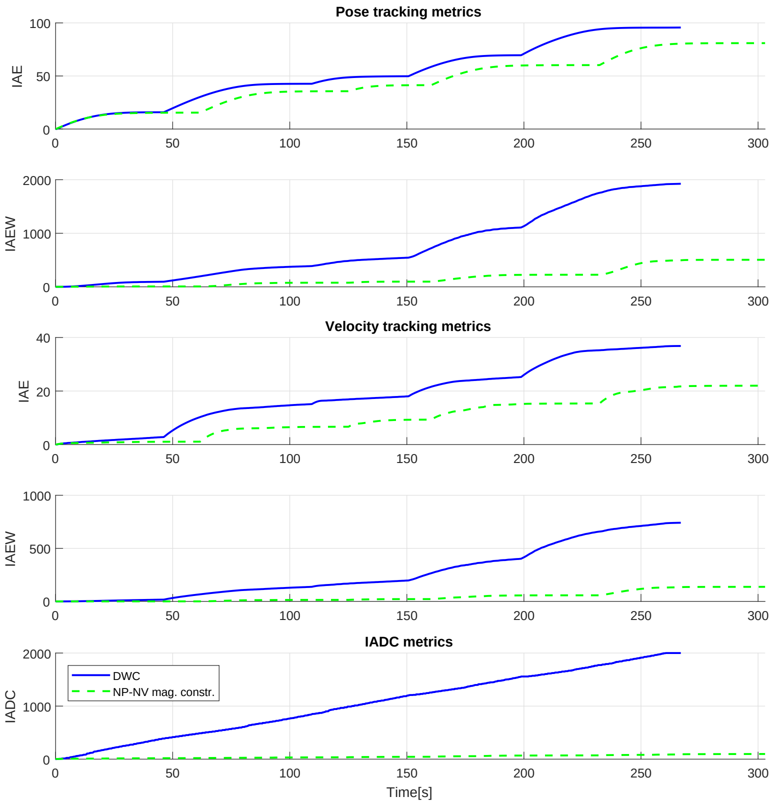


Figure 5.31: IAE, IAEW and IADC for the experiment with the DWC.

Lastly, the DWC was tested. As investigated in the simulations, the laboratory performance of the current implementation of the DWC was unstable due to the presence of disturbances and time delays. One important observation is that the experimental results coincide in great detail with the simulated results in Section 4.5.4.

## 5.6 Chapter summary

In this chapter, the Marine Cybernetics Laboratory was presented along with its hardware and software packages. In addition, the work done with the estimator was explained in detail. Then the experimental results for the different control algorithms from the two laboratory weeks were presented. A discussion about the combined result from the numerical simulations and the laboratory experiments follows in Chapter 6.

# Chapter 6

# Discussion

The different control algorithms were tested with and without actuator constraint handling methods in both numerical simulations and laboratory experiments and were evaluated through the use of performance metrics. These results can be combined in order to better understand in which scenarios the controllers have the best performance, and when it would be recommended to use the different actuator saturation effects and models.

Through simulations it became clear that the included nonlinearities in the NP-LV and the NP-NV controllers were beneficial for the total performance of the controllers, since the nonlinear feedback modifies the convergence rate to be faster than the linear feedback in the linear region. From the simulation results it could be seen that the NP-NV controller is the most accurate and energy efficient of the three controllers, while also having the fastest transient response. However, it should be noted that the magnitude of $\tau_1$ and $\tau_2$ are the lowest for the NP-LV controller, while in addition also having the least usage of the actuators. In laboratory week 1, it was shown that the NP-NV controller had the best accuracy, while the NP-LV controller had the best energy efficiency and the least actuator wear and tear. Furthermore, in laboratory week 2, with the updated ship model and estimator, which had greatly reduced noise in the velocity estimates, the NP-LV controller were shown to have the best control accuracy and energy efficiency, along with the least actuator wear and tear. Based on this, it can be concluded that the NP-LV controller comes out on top. The reason why the simulation results are different from the experimental results is because of the disturbances and uncertainties present in the laboratory. The NP-NV controller is inherently a more aggressive control scheme due to its nonlinear velocity feedback, which possibly made it better at handling the larger disturbances present in the first laboratory week, mostly caused by the velocity estimator. With the updated estimator, the amount of disturbances on the velocity estimate were greatly reduced, and the NP-LV controller achieved a better overall performance. It can be noted that the nonlinear velocity feedback in the NP-NV controller possibly increases the controller's robustness, but at the cost of energy usage. This robustness probably caused the NP-NV controller to outperform the NP-LV controller in the first laboratory week where the ship model had notable modeling errors. Another noteworthy observation is that the NP-NV controller still had the higher magnitude on the commanded actuator input signal in the laboratory experiments, which potentially can result in overall poorer performance, as will be discussed next.

Introducing actuator saturation effects into the numerical simulator yielded no practically useful results, as the reduction in the allowed magnitude and rate outputs of the actuators only caused the boat to navigate the 4-corner test slower. This is due to the fact that the simulator is a perfect scenario where no noise or disturbances are present, and where actuator saturation effects have no positive impact on performance. The main results for the usage of actuator constraint handling methods thus comes from the laboratory experiments.

In laboratory week 1, it was shown that including magnitude saturation or MRS effects had positive impacts on performance, where it was shown that the NP-LV and NP-NV controllers with the MRS model included had the best overall performance due to their reduction in actuator wear and tear. It was also shown that changing the gain matrix $\boldsymbol{K}$ can result in improved performance for the MRS model, proving that, as with other models and controllers, tuning is essential to achieve optimal control performance. In laboratory week 2, the MRS model was tuned differently, allowing for a higher rate of change in the commanded actuator input signal. With the now reduced estimation noise, the effects of the MRS model was reduced, but the notable positive effects on actuator usage were still visible. This accentuates the importance of the choice of constraints while using such a model, and that one must carefully choose the saturation limits $\boldsymbol{m}$ and $\boldsymbol{r}$ along with the MRS gain matrix $\boldsymbol{K}$, which has to be based on the ship you are to use it on and the environments it is to operate in, in order to achieve positive effects on performance. Based on this, it is shown that the MRS model can, if tuned properly, increase a controller's robustness, making it less vulnerable to noise and disturbances, while at the same time also having the potential to increase both control accuracy and energy efficiency and effectively decrease actuator twitching. These effects are especially important for vessels which must operate for longer times at sea, and can be particularly useful for ships in DP operations, effectively contributing to the longevity of the operation with a reduced need for maintenance and repairs. It should also be noted that the MRS model was tested on CSAD with another thrust allocation model, where the azimuth thrusters were not locked in specific angles. Here, it greatly reduced overall actuator twitching and energy usage.

Through the main simulation results, it was shown that, theoretically, the DWC was the best performing controller in this thesis. It should be noted that the DWC is based on the linear ship model, while the cascaded feedback controllers were tested based on the full nonlinear ship model, which has improved their overall performance, making the DWC's victory more profound. In the extended simulation results, it was shown that the current implementation of the DWC had some robustness issues. It was proven to be able to handle model uncertainties without creating instabilities in the commanded actuator signal, but started having issues when estimation noise and time delays were introduced into the simulator. These effects had little to no effect on the overall performance of the NP-NV controller, proving that the cascaded feedback controllers have a high level of robustness and thus can handle normal amounts of disturbances and time delays without having performance issues. Through the extended simulations results, it then became clear that it was expected that the DWC would have some performance issues when experimentally tested in the laboratory, where we knew that estimation noise were present due to the estimator and the camera position measurement system, and which we learned while testing the DWC, that CSAD most likely had a system time delay which we had no control over.

When experimentally testing the DWC, it became clear that there was several issues that needed to be addressed. In Figure 5.2, the original MC-Lab system block diagram is shown. To address the issue of system time delays, the system was changed in order to make sure that every subsystem were running on the same frequency. However, this did not have any impact on performance, and other adjustments to the algorithm were tried. Effort was put into including acceleration limits and rate constraints in the algorithm, minorly reducing or increasing the window size, redesigning the choice of the desired velocity, and including nonlinear feedback terms with different gains. Unfortunately, neither of these adjustments yielded any noteworthy improvements to performance. It thus became clear that there most likely was a system time delay present which caused instabilities for the DWC, and it was clear that for the DWC to work optimally, it is in need of an excellent ship model, not only of the ship's dynamics, but also of the signal processing in the system. To be able to account for a system time delay and reduce model uncertainties, a good model identification and analysis is then recommended.

The reason why the DWC is so sensitive to time delays is because as long as the desired velocity $\boldsymbol{\alpha}$ is outside of the window, the controller will keep the control inputs high while tracking the optimal velocity $\boldsymbol{\alpha}_f$ on the boundaries of the window. Theoretically, when $\boldsymbol{\alpha}$ is inside the window, the DWC will decrease the control inputs and track the desired velocity. However, when a time delay which is larger than one sample is present, the DWC will overshoot the desired velocity, practically making the window slide past the desired velocity. At the next time step, the desired velocity will then be outside the window again, making the DWC command the control input at maximum thrust in the opposite direction, effectively making the control signal oscillate while trying to track the desired velocity $\boldsymbol{\alpha}$. The DWC being a model predictive controller, these kind of oscillations can occur not only because of time delays, but whenever the current velocity at the next time step does not coincide with the predicted velocity, especially if the current velocity will end up outside of the predicted window. DWC being a feedforward approach makes it especially vulnerable to time delays and disturbances in measured signals, which is why it is usually not recommended to use such signals in feedforward-based controllers, and rather base the tracking signals upon reference signals.

In order to increase the DWC's performance, it is then necessary to increase the overall robustness of the controller. To avoid the problems which occur when time delays are present, greatly increasing the window size is an option. In practical situations, there is no need to run the DW algorithm at 100 Hz. By lowering the frequency of the algorithm, the algorithm will be less prone to time delays creating stability issues, since the desired velocity $\boldsymbol{\alpha}$ will most likely still be inside the window, which is now larger because of the reduction in algorithm frequency, unless the time delay is sufficiently big. If an analysis of the system time delay has been done and the time delay has been identified, one option is also to account for the time delay inside the algorithm, where it would be possible to increase the window size with an amount which corresponds to the maximum added time delay $t_d$ to the total time $T = t_s + t_d$ which the window is based upon, where $t_s$ is the sampling time of the control system and $T$ is the allowed time for acceleration for the DW algorithm during the next time step.

This was investigated through simulations, where the DWC was run at 100 Hz, and a time delay of $t_d = 0.02$ [s] was present. The results are shown in Figure 6.1, where in the top plot the commanded surge force is shown for the case where the time delay is not accounted for. The middle plot shows how the performance has improved when the time delay is exactly accounted for. The lower plot shows how the performance has further improved when

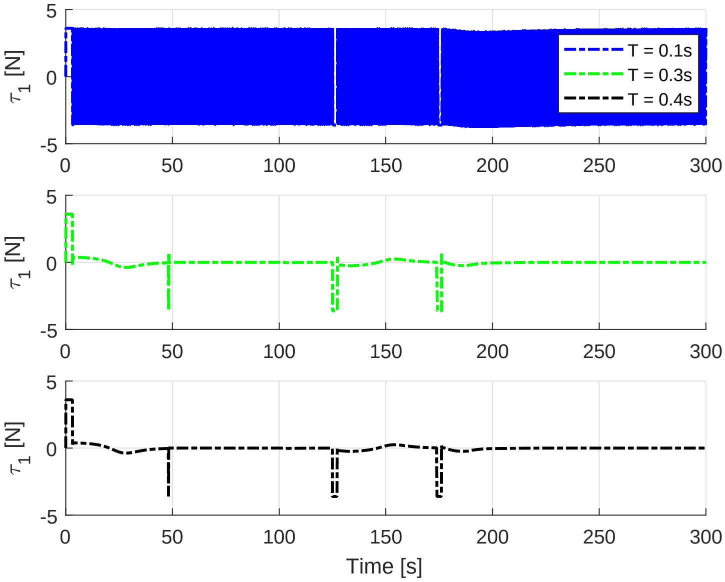


Figure 6.1: The commanded surge input for the DWC when accounting for a known time delay.

$T = t_s + t_d + t_e$ has been extended to include an additional time term $t_e = 0.01$ [s], which is meant to further increase the window size, resulting in a smoother control input signal because it makes sure that the desired velocity $\boldsymbol{\alpha}$ does not exactly end up on the boundary of the window one time step before it is reached, making the controller have a smother transient response on the actuators when approaching the desired velocity $\boldsymbol{\alpha}$. The improvement of performance can be observed in the figure, where the overshoots present in the middle plot have been avoided. From this, it can be concluded that by increasing the window size can effectively smoothen the performance of the DWC, which will result in an improvement to the overall performance when disturbances, uncertainties and time delays are present.

There is not much that can be done for the case of estimator estimation noise, expect for possibly filtering out the worst noise. For the case of ships that does not use estimated measurements, it would be beneficial to directly account for the disturbances, such as ocean currents, in the DW algorithm if they are measurable through sensor systems. This was tested through simulations, where disturbances were added to the simulator and then accounted for in the DW algorithm, which improved performance. To further increase the robustness of the DWC in real life scenarios, it would be beneficial to include a nonlinear feedback term to better account for external disturbances and uncertainties. In addition, extending this feedback into an adaptive feedback term can further increase performance. Adaptive control is a technique where the goal is to estimate unknown parameters of the system model. The estimates are then used to adjust the control input, thus compensating for model uncertainties and system disturbances and leading to improved performance. Different adaptive techniques have been investigated in the later years, including model reference adaptive control (MRAC) and $\mathcal{L}_1$ adaptive control. As seen in [30], adaptive control can improve ship performance where the model have inherent uncertainties, and therefore also further improve the performance of the DWC, which is vulnerable to both uncertainties and disturbances.

Furthermore, for the DWC to accurately predict the velocity at the next time step, actuator rate constraints will have to be added to the algorithm. This would limit the window size based on the actuator acceleration limits during the next time step, making sure that the chosen optimal velocity $\boldsymbol{\alpha}_f$ actually is feasible.

In addition, in the current design the window size is based on the linear ship model. As an extension to the work done in [5] and [6], a possible improvement to (3.97) and (3.98) would be to modify these to include the full model, giving

$$\dot{\boldsymbol{\nu}}_{min} = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{min}(\boldsymbol{u}) - \boldsymbol{C}\boldsymbol{\nu}^* - \boldsymbol{D}\boldsymbol{\nu}^*) \tag{6.1}$$

$$\dot{\boldsymbol{\nu}}_{max} = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{max}(\boldsymbol{u}) - \boldsymbol{C}\boldsymbol{\nu}^* - \boldsymbol{D}\boldsymbol{\nu}^*). \tag{6.2}$$

This would give more physical realistic acceleration limits when operating during higher velocities inside the linear model-based DWC, increasing the accuracy of the predicted velocity $\boldsymbol{\alpha}_f$ at the next time step, which potentially could improve overall performance by preventing oscillations caused by wrongly predicted velocities.

By working with the DWC, several things has thus been learned. When operating with a DW based on a linearized ship model designed to be used for low-speed manoeuvres, such as in dynamic positioning, a perfectly modeled mathematical model of the ship is not completely necessary, but one should strive to get it as accurate as possible to improve the overall performance. System time delays should be identified and accounted for inside the algorithm,

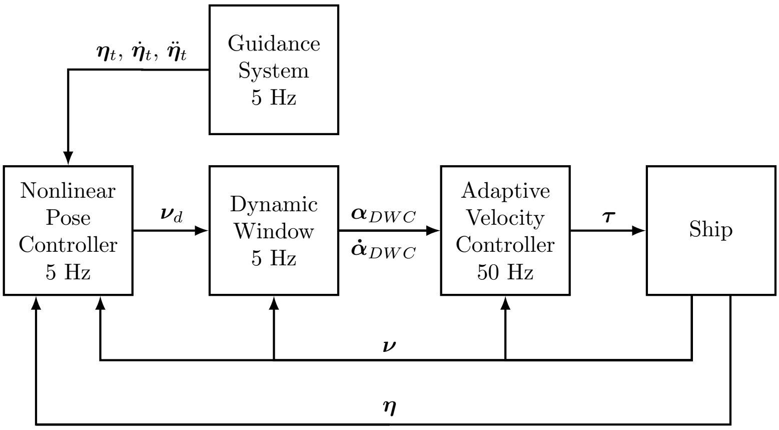

$\eta_t$, $\dot{\eta}_t$, $\ddot{\eta}_t$
Guidance System 5 Hz
Nonlinear Pose Controller 5 Hz
$\nu_d$
Dynamic Window 5 Hz
$\alpha_{DWC}$
$\dot{\alpha}_{DWC}$
Adaptive Velocity Controller 50 Hz
$\tau$
Ship
$\nu$
$\eta$

Figure 6.2: Block diagram for a suggestion for an improvement to the dynamic-window based controller.

and if disturbances are measurable, accounting for these in the DW algorithm is beneficial as well, and in addition, to achieve a better performance, a nonlinear feedback, such as the one from the NP-NV controller, should be included in the control system in order to better account for disturbances and uncertainties. Furthermore, this feedback could be extended to include adaptive control terms in order to further improve performance. In addition, to more accurately predict the velocity at the next time step, actuator rate constraints should be included into the algorithm as well. Moreover, it could be beneficial for the algorithm to work with larger window sizes, which could make the algorithm avoid oscillating around a desired velocity. To achieve this, the algorithm should run at lower frequencies, resulting in a higher system sampling time $t_s$, thus increasing the window size. This could possibly be combined with a control system which runs at a higher frequency, where the DW algorithm chooses the desired velocity $\boldsymbol{\alpha}_{DWC}$ and acceleration $\dot{\boldsymbol{\alpha}}_{DWC}$ for the next time $T$, while the feedback controller runs at a higher frequency, to more effectively account for the external disturbances. A suggestion for such a motion control system is presented in Figure 6.2, where disturbances should be measured by sensor systems and accounted for in both the DW algorithm and the adaptive velocity controller. Here the velocity controller runs at 50 Hz to effectively account for external disturbances, while the rest of the system runs at 5 Hz to make the set of feasible velocities larger to obtain an overall smoother control performance.

While the DWC continues to be the best performing controller in numerical simulations, it is clear that more work has to be done with the algorithm in order to make it work properly in real life scenarios. One could say that the DWC has great potential, and when the presented issues are solved, possibly through the suggestions discussed here, it will be interesting to see if the control algorithm will be able to live up to its full potential when applied to real life scenarios on-board actual vessels. Unfortunately, there was not enough laboratory time available during the time this project was done to test any of the suggested changes to the DWC system in laboratory environments.

# Chapter 7

# Conclusions and future work

## 7.1 Conclusions

Throughout this thesis, several controllers have been designed and tested. Cascaded feedback controllers were developed, and it was shown through both numerical simulations and laboratory experiments that including nonlinearities into the feedback algorithms improve both control accuracy and energy efficiency. The effects of handling actuator constraints were investigated, and proven to be of importance to increase overall ship performance. An MRS model was modified to fit ship motion control systems, and was implemented in order to handle actuator constraints. It was shown that such a model, if tuned correctly, can further increase the overall control accuracy and energy efficiency of a controller, while at the same time also reduce actuator wear and tear, for a ship in setpoint navigation.

Furthermore, dynamic window-based controllers were developed. Through numerical simulations it was shown that the 3 DOF DWC outperformed the other controllers in both control accuracy and energy efficiency. However, through laboratory experiments, the controller was shown to have stability issues due to system time delays. Because of this, further improvements are needed to be added into the control algorithm in order to account for time delays, disturbances and uncertainties before the controller is ready to work optimally in a real-life experimental environment.

Motion control systems for ships have traditionally not focused on handling actuator constraints. Through this thesis, it has been shown that taking actuator constraints into consideration when designing a controller will be beneficial in most scenarios. More research and effort needs to be put into this field, and will result in more accurate and efficient motion control systems for ships in the coming future.

## 7.2 Future work

The following topics are suggested as a continuation to this thesis:

- Conduct a proper model identification of the model-scale ship CSAD to obtain a more precise ship model, a better understanding of the limitations of the actuators, and to identify system time delays.
- Further develop the presented nonlinear cascaded feedback controller so that it handles actuator constraints.
- Increase the performance of the MRS model by optimizing the choice of actuator constraints and tuning rules through a set of laboratory experiments.
- Extend the 3 DOF DWC to include the full nonlinear ship model to achieve an overall better control performance, and possibly compute the set through analytical methods to obtain a more precise 3 DOF set of possible velocities.
- Introduce dynamics to the control input vector $\boldsymbol{\tau}$ for the dynamic window algorithm in order to handle rate constraints, to make sure that the chosen optimal velocity $\boldsymbol{\alpha}_f$ actually is feasible.
- Include adaptive control algorithms into the DWC in order to better account for disturbances and uncertainties present in the ship motion control system and the environment it operates in.
- Develop suggested window-sizing rules to ensure that an oscillating control input signal for the DWC is avoided.
- A thorough stability analysis of the DWC to obtain a better understanding about under which conditions the control algorithm is stable.

In addition, while working with the DW algorithm for a longer period of time, several ideas for alternative uses for the algorithm were contemplated.

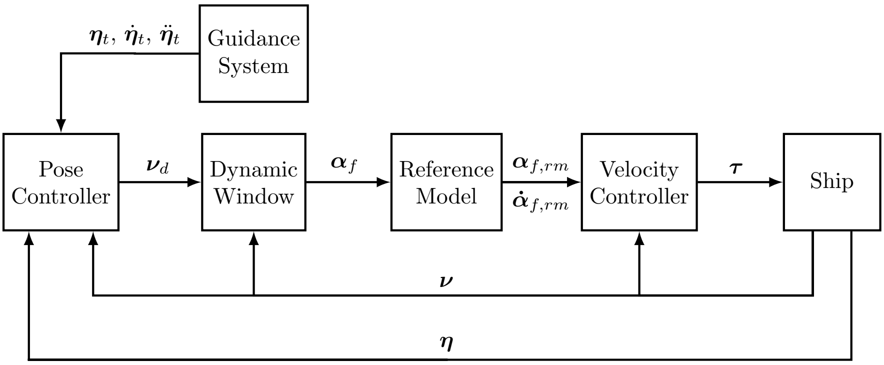


Figure 7.1: Block diagram for using the DW algorithm as a reference generator for a cascaded feedback controller.

### Using the DW algorithm as a reference generator

Instead of using the dynamic window algorithm to create a dynamic window-based controller, it could be used as a reference generator. The implementation would be similar to that of the DWC presented here, but instead of creating the desired acceleration and velocity, $\dot{\boldsymbol{\alpha}}_{DWC}$ and $\boldsymbol{\alpha}_{DWC}$, respectively, the chosen optimal velocity which is to be achieved within the next time step $T$ could be used as a reference signal. Thus, the DW algorithm would, if necessary, modify the desired velocity $\boldsymbol{\nu}_d$ from the pose controller to a feasible desired velocity $\boldsymbol{\alpha}_f$, and thereby making sure that the motion control system satisfies the actuator constraints. The feasible velocity could then be used together with a reference model, which is commonly implemented by low-pass filters, to obtain the desired acceleration $\dot{\boldsymbol{\alpha}}_f$, or it could be computed using numerical derivation. Using this DW implementation together with the NP-NV controller, the control law would be modified to

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_f + \boldsymbol{D}_L\boldsymbol{\alpha}_f - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2, \tag{7.1}$$

for the case of the use on the linearized ship model. This could potentially remove the problems with actuator twitching caused by the oscillating $\dot{\boldsymbol{\alpha}}_{DWC}$ in the current DWC implementation when disturbances and time delays are present. A block diagram for such a system is presented in Figure 7.1.

### Dynamic window-based estimator

It would be possible to implement an estimator based on the DW algorithm. Such an estimator would be performing well in dead reckoning scenarios, as for example in the MC-Lab when the measurement signal is lost, and would be similar to a Kalman filter or an observer.

This estimator, as with any use of the dynamic window algorithm, is in need of a good ship model to function properly. It would use the control input at the current time step $\boldsymbol{\tau}_n$ as input, and would, based on the current estimated velocity $\hat{\boldsymbol{\nu}}_n$, estimate, through the use of the dynamic window algorithm, the velocity at the next time step $\hat{\boldsymbol{\nu}}_{n+1}$ based on the ship dynamics and the control input. An estimate for the pose at the next time step $\hat{\boldsymbol{\eta}}_{n+1}$ could then be computed by using the estimated velocities $\hat{\boldsymbol{\nu}}_{n+1}$ and $\hat{\boldsymbol{\nu}}_n$ and the pose at the the current time step $\hat{\boldsymbol{\eta}}_n$.

Using such an estimator in the MC-Lab could potentially remove the need to use measurements from the camera positioning system, except for during initialization or resetting of the system. Because of this, it would be possible to get around the unwanted estimation noise that are added to the control system in the MC-Lab with the current estimator. In addition, such as system, because it is based on the dynamics of the ship, would make sure that only feasible estimated velocities would be sent to the control system, meaning that it would be a robust estimator in the sense that it would not feed the controller with dynamically infeasible velocity and pose estimates.

### Dynamic window-based guidance system

Another thought is that algorithms such as the dynamic window algorithm, could potentially be used in autonomy modules, such as guidance systems, to make sure that the target trajectory is feasible for the ship it is used on. This could be a part of a high-level anti-collision approach, creating an anti-collision dynamic window-based guidance system, which for example could be used in hard-to-navigate scenarios where it is important that the ship can exactly follow the desired path in order to avoid collisions with the terrain.

# Bibliography

[1] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, Wiley, 2011.

[2] V. Kapila and S. Vallura, *"Model predictive control of systems with actuator and rate saturation", in Proceedings of the 37th IEEE Conference on Decision and Control*, 1998.

[3] D. Fox, W. Burgard, and S. Thrun *"The dynamic window approach to collision avoidance", IEEE Robotics & Automation Magazine*, 4(1), pp 23-33, 1997.

[4] B.-O. H. Eriksen, M. Breivik, K. Y. Pettersen and M. S. Wiig, *"A modified dynamic window algorithm for horizontal collision avoidance for AUVs", in Proceedings of the IEEE Multi-Conference on Systems and Control, Buenos Aires, Argentina*, 2016.

[5] M. E. N. Sørensen, M. Breivik and B. O. H. Eriksen *"A Ship Heading and Speed Control Concept Inherently Satisfying Actuator Constraints", IEEE Conference on Control Technology and Applications, Mauna Lani, HI, USA*, 2017.

[6] M. E. N. Sørensen, O. N. Lyngstadaas, B.-O. H. Eriksen and M. Breivik, *"A Dynamic Window-Based Controller for Dynamic Positioning Satisfying Actuator Constraints", in Proceedings of the 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Opatija, Croatia*, 2018.

[7] M. E. N. Sørensen and M. Breivik, *"Comparing Combinations of Linear and Nonlinear Feedback Terms for Motion Control of Marine Surface Vessels", in*

*Proceedings of the 10th IFAC Conference on Control Applications in Marine Systems, Trondheim, Norway*, 2016.

[8] S. Galeani, S. Onori, A. R. Teel and L. Zaccarian, *"A Magnitude and Rate Saturation Model and its Use in the Solution of a Static Anti-Windup Problem", Systems & Control Letters*, 57(1), 2007.

[9] O. N. Lyngstadaas, T. E. Sæterdal, M. E. N. Sørensen and M. Breivik, *"Improvement of Ship Motion Control Using a Magnitude-Rate Saturation Model", in Proceedings of the 2nd IEEE Conference on Control Technology and Applications, Copenhagen, Denmark*, 2018.

[10] H. K. Khalil, *Nonlinear Systems*, Third Edition, Pearson, 2014.

[11] E. Panteley and A. Loria *"On Global Uniform Asymptotic Stability of Nonlinear Time-Varying Systems in Cascade", Systems & Control Letters*, 33(2), pp. 131-138, 1998.

[12] J. Bjørnø, R. Skjetne, and A. R. Frederich, *"Modeling, Parameter Identification and Thruster-Assisted Position Mooring of C/S Inocean Cat I Drillship", in Proceedings of the 36th ASME International Conference on Ocean, Offshore and Arctic Engineering, Trondheim, Norway*, 2017.

[13] O. N. Lyngstadaas, *"A Ship Motion Control Concept Inherently Satisfying Actuator Constraints"*, Project Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2017. Download.

[14] M. E. N. Sørensen, E. S. Bjørne and M. Breivik, *"Performance Comparison of Backstepping-based Adaptive Controllers for Marine Surface Vessels", in Proceedings of the 5th IEEE Conference on Control Applications, Buenos Aires, Argentina*, 2015.

[15] M. E. N. Sørensen, M. Breivik and R. Skjetne, *"Comparing Combinations of Linear and Nonlinear Feedback Terms for Ship Motion Control", submitted to IEEE Transactions on Control Systems Technology*, 2018.

[16] M. E. N. Sørensen and M. Breivik, *"Comparing nonlinear adaptive motion controllers for marine surface vessels", in Proceedings of the 10th IFAC Conference on Manoeuvring and Control of Marine Craft, Copenhagen, Denmark*, 2015.

[17] B.-O. H. Eriksen, *"Horizontal Collision Avoidance for Autonomous Underwater Vehicles"*, Master thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2015.

[18] T. I. Fossen, M. Breivik and R. Skjetne, *"Line-of-Sight Path Following of Underactuated Marine Craft", in Proceedings of the 6th IFAC Conference on Manoeuvring and Control of Marine Craft, Girona, Spain*, 2003.

[19] T. I. Fossen and J. P. Strand, *"Tutorial on Nonlinear Backstepping: Applications to Ship Control", Modeling, Identification and Control, 20(2), pp. 83-135*, 1999.

[20] J. E. Refsnes, *"Nonlinear Model-Based Control of Slender Body AUVs"*, PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2008.

[21] F. Lamnabhi-Lagarrigue, E. Panteley, E. Lefeber and A. Loria, *Advanced Topics in Control Systems Theory: Lecture Notes from FAP 2004*, Springer, 2005.

[22] M. Breivik, J. P. Strand and T. I. Fossen, *"Guided Dynamic Positioning for Fully Actuated Marine Surface Vessels", in Proceedings of the 7th IFAC Conference on Manoeuvring and Control of Marine Craft, Lisbon, Portugal*, 2006.

[23] M. Breivik and T. I. Fossen, *"Guidance laws for autonomous underwater vehicles" in Underwater Vehicles* (A. V. Inzartsev, ed.), IN-TECH Education and Publishing, 2009.

[24] T. I. Fossen and K. Y. Pettersen, *"On Uniform Semiglobal Exponential Stability (USGES) of Proportional Line-of-Sight Guidance Laws", Automatica*, 50(11), pp. 2912-2917, 2014.

[25] R. Skjetne, M. E. N. Sørensen, M. Breivik, S. A. T: Værnø, A. H. Brodtkorb, A. J. Sørensen, Ø. K. Kjerstad, V. Calabrò and B. O. Vinje, *"AMOS DP research cruise 2016: Academic full-scale testing of experimental dynamic positioning control algorithms onboard R/V Gunnerus", in Proceedings of the 36th International Conference on Ocean, Offshire and Arctic Engineering, Trondheim, Norway*, 2017.

[26] B. -O. H. Eriksen and M. Breivik *"Modeling, Identification and Control of High-Speed ASVs: Theory and Experiments"*, pp. 407-431, Sensing and

Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles, Springer International Publishing, 2017.

[27] T. I. Fossen, *"Nonlinear passive control and observer desgin for ships", Modeling, Identification and Control*, 21(3), pp. 129-184, 2000.

[28] P. Ögren and N. E. Leonard, *"A convergent dynamic window approach to obstacle avoidance", IEEE Transactions on Robotics"*, 21(2), pp. 188-195, 2005.

[29] A. Pavlov, H. Nordahl and M. Breivik, *"MPC-based optimal path following for underactuated vessels", in Proceedings of the 8th IFAC Conference on Manoeuvring and Control of Marine Craft, Guarujá, Brazil*, 2009.

[30] T. E. Sæterdal, *"Adaptive Motion Control for Ships Using Nonlinear Control Schemes"*, Master thesis, Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway, 2018.

[31] W. Caharija, *"Integral Line-of-Sight Guidance and Control of Underactuated Marine Vehicles"*, PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2014.

[32] *Marine Cybernetics Laboratory Handbook*, Norwegian University of Science and Technology, 2017. Access.

# Appendices

# Appendix A

# CSAD model, parameters and code

Here, the MATLAB code for the implementation of the full model of CSAD is presented. See Section 2.1 for a more thorough explanation of the modeling.

## A.1 CSAD MATLAB code

### A.1.1 Vessel function

```
function [Eta_dot,Nu_dot] = Vessel(eta,tau_s,Nu)
    Nu_dot = CSAD_full(Nu,tau_s);
    yaw = eta(3);
    R =[cos(yaw) -sin(yaw) 0;
        sin(yaw)  cos(yaw) 0;
        0         0        1];
    Eta_dot = R*Nu;
end
```

### A.1.2 CSAD function

```
function nu_dot = CSAD_full(Nu,tau)
    X_u        = -5.35;
    X_uu       =  0;
    X_uuu       = -19.6312;

    Y_v         = -10.16;
    Y_vv        = -0.8647;
    Y_vvv       = -681.1745;

    Y_r         = -7.25;
    Y_rr        = -3.45;
    Y_rrr   = 0;

    N_v         = 0;
    N_vv        = -0.2088;
    N_vvv = 0;

    N_r         = -14.55;
    N_rr        = -9.9597;
    N_rrr       = -0.3101;

    N_rv        = 0.08;
    N_vr        = 0.08;
    Y_rv        = -0.805;
    Y_vr        = -0.845;

    X_ud =      -10;
    Y_vd =      -105;
    Y_rd =      -0.525;
    N_vd =      -0.157;
    N_rd =      -3.4950;

    Nur  = Y_rd;
    Nuv  = -(Y_vd-X_ud);
    Yur  = X_ud;

    x_g     = 0.0375;
   m        = 127.92;
    I_z     = 61.967;

    u   = Nu(1);
    v   = Nu(2);
```

```
        r     = Nu(3);

        d_11  = - X_u - X_uu*abs(u) - X_uuu*u^2;
        d_22  = - Y_v - Y_vv*abs(v) - Y_rv*abs(r) - Y_vvv*v^2;
        d_23  = - Y_r - Y_rr*abs(r) - Y_vr*abs(v) - Y_rrr*r^2 - Yur*u;
        d_32  = - N_v - N_vv*abs(v) - N_rv*abs(r) - N_vvv*v^2 - Nuv*u;
        d_33  = - N_r - N_rr*abs(r) - N_vr*abs(v) - N_rrr*r^2 - Nur*u;

        D = [d_11   0          0;
              0     d_22     d_23;
              0     d_32    d_33];

        CA = [0                      0            Y_vd*v+Y_rd*r;
               0                     0                  -X_ud*u;
              -Y_rd*r-Y_vd*v      X_ud*u                      0];

        CRB = [0                   0          -m*(x_g*r+v);
                0                  0                   m*u;
                m*(x_g*r+v)       -m*u                   0];

        C = CRB + CA;

        M_RB_c = [m       0             0;
                  0       m          m*x_g;
                  0       m*x_g       I_z];

        M_A_c = [-X_ud      0               0;
                  0        -Y_vd       -Y_rd;
                  0        -N_vd      -N_rd];

        M = M_RB_c + M_A_c;

        nu_dot = M\(tau - C*Nu - D*Nu);
    end
```

# Appendix B

# DWC code and implementation improvements

Here, the code needed to implement the 3 DOF dynamic window-based controller in MATLAB/Simulink is presented. In addition, a fix which removes actuator twitching due to operation on or close to the boundary lines of $V_p$ is also presented.

## B.1 DWC MATLAB code

This is the code for the DWC based on the linearized ship model without a nonlinear feedback. It can further be extended to account for disturbances, time delays and uncertainties through the suggested improvements discussed in Chapter 6. When doing experiments with the DWC in the MC-Lab, it became clear that the processing time of this algorithm could be a potential issue if it is wished to run the control system at a high frequency, as the algorithm is quite computationally heavy. It is thus necessary to keep code optimization in mind. This code is not the most optimized one or the one used in the laboratory, but is included here because it is the most readable and easy to understand of the DWC implementations that were developed in this project. The reader should thus possibly put effort into optimizing this code if it is wanted to do experimental tests on a model-scale vessel with limited computational powers, or, simply, reduce the frequency of the control system.

### B.1.1 Pose controller

```
function Nu_d  = PoseControl(Eta_t, Eta_tdot, Eta)
Gamma1 =  diag([0.08, 0.08, 0.0698]);
delta_psi = 0.5;
delta_p = 0.5;

psi = Eta(3);

R = [cos(psi) -sin(psi) 0; sin(psi) cos(psi) 0; 0 0 1];
R_T = R';

z1 = R_T * (Eta - Eta_t);

z1_p = [z1(1), z1(2)]';
z1_psi = z1(3);

first = 1/(sqrt(z1_p'*z1_p + delta_p^2));
second = 1/(sqrt(z1_psi^2 + delta_psi^2));

K1 = Gamma1*[first*eye(2,2) zeros(2,1); zeros(1,2) second];

Nu_d = R_T * Eta_tdot - K1 * z1;
end
```

### B.1.2 3 DOF dynamic window algorithm

```
function [alpha_DWC_dot] = CSAD_DWC3_linear(Nu_d, Nu)
tau_min = [-3.5996 -2.0032 -1.7027]';
tau_max = [3.5996 2.0032 1.7027]';

[M,C,D] = CSAD_linear2(Nu);
Nu_dot_min = M\(tau_min - D*Nu);
Nu_dot_max = M\(tau_max - D*Nu);
```

```
ts = 0.01;
Nu_min = Nu+ts*Nu_dot_min;
Nu_max = Nu+ts*Nu_dot_max;

%% Yawrate/surge
u_points_ys = linspace(Nu_min(1),Nu_max(1),21);
r_points_ys = linspace(Nu_min(3),Nu_max(3),21);

if CSAD_g3_linear(Nu_d, 'yawrate_surge')>0 && (Nu_d(1)>= Nu_min(1) && Nu_d(1)<= Nu_max(1)) && (Nu_d(3)>= Nu_min(3) && Nu_d(3)<= Nu_max(3))
    u_diff_ys = u_points_ys-Nu_d(1);
    [~,u_move_ys] = min(abs(u_diff_ys));
    u_points_ys(u_move_ys) = u_points_ys(u_move_ys)-u_diff_ys(u_move_ys);

    r_diff_ys = r_points_ys-Nu_d(3);
    [~,r_move_ys] = min(abs(r_diff_ys));
    r_points_ys(r_move_ys) = r_points_ys(r_move_ys)-r_diff_ys(r_move_ys);
end

if CSAD_g3_linear(Nu_d,'yawrate_surge')>0 && (Nu_d(1)>= Nu_min(1) && Nu_d(1)<= Nu_max(1))
    u_diff_ys = u_points_ys-Nu_d(1);
    [~,u_move_ys] = min(abs(u_diff_ys));
    u_points_ys(u_move_ys) = u_points_ys(u_move_ys)-u_diff_ys(u_move_ys);
end

if CSAD_g3_linear(Nu_d,'yawrate_surge')>0 && (Nu_d(3)>= Nu_min(3) && Nu_d(3)<= Nu_max(3))
    r_diff_ys = r_points_ys-Nu_d(3);
    [~,r_move_ys] = min(abs(r_diff_ys));
    r_points_ys(r_move_ys) = r_points_ys(r_move_ys)-r_diff_ys(r_move_ys);
end
```

```
uu_points_ys  =[u_points_ys;u_points_ys;u_points_ys;
    u_points_ys;u_points_ys;u_points_ys;u_points_ys;
    u_points_ys;u_points_ys;u_points_ys;u_points_ys;
    u_points_ys;u_points_ys;u_points_ys;u_points_ys;
    u_points_ys;u_points_ys;u_points_ys;u_points_ys;
    u_points_ys;u_points_ys];
rr_points_ys  =[r_points_ys;r_points_ys;r_points_ys;
    r_points_ys;r_points_ys;r_points_ys;r_points_ys;
    r_points_ys;r_points_ys;r_points_ys;r_points_ys;
    r_points_ys;r_points_ys;r_points_ys;r_points_ys;
    r_points_ys;r_points_ys;r_points_ys;r_points_ys;
    r_points_ys;r_points_ys]';

uss_ys = reshape(uu_points_ys,[1,21*21]);
rss_ys = reshape(rr_points_ys,[1,21*21]);
Nu_r_ys = zeros(3,21*21);
Nu_r_ys(1,:) = uss_ys;
Nu_r_ys(3,:) = rss_ys;
h_ys = CSAD_g3_linear(Nu_r_ys,'yawrate_surge');

Max_point=21*21;
Max_point=Max_point-sum(h_ys<0);
Nu_f_ys = [uu_points_ys(h_ys>=0) zeros(Max_point,1)
    rr_points_ys(h_ys>=0)]';

Ob_ys = CSAD_objective_function_dw3(Nu_d,Nu_f_ys,'
    yawrate_surge');

[~,vel_pair_ys] = max(Ob_ys(:,3));

alpha_ys = Nu_f_ys(:,vel_pair_ys);

%% Yawrate/sway
v_points_ysw = linspace(Nu_min(2),Nu_max(2),21);
r_points_ysw = linspace(Nu_min(3),Nu_max(3),21);

if CSAD_g3_linear(Nu_d, 'yawrate_sway')>0 && (Nu_d(2)
   >= Nu_min(2) && Nu_d(2)<= Nu_max(2)) && (Nu_d(3)>=
    Nu_min(3) && Nu_d(3)<= Nu_max(3))
    v_diff_ysw = v_points_ysw-Nu_d(2);
    [~,v_move_ysw] = min(abs(v_diff_ysw));
    v_points_ysw(v_move_ysw) = v_points_ysw(v_move_ysw)-v_diff_ysw(v_move_ysw);

    r_diff_ysw = r_points_ysw-Nu_d(3);
    [~,r_move_ysw] = min(abs(r_diff_ysw));
    r_points_ysw(r_move_ysw) = r_points_ysw(r_move_ysw)-r_diff_ysw(r_move_ysw);
end

if CSAD_g3_linear(Nu_d,'yawrate_sway')>0 && (Nu_d(2)>= Nu_min(2) && Nu_d(2)<= Nu_max(2))
    v_diff_ysw = v_points_ysw-Nu_d(2);
    [~,v_move_ysw] = min(abs(v_diff_ysw));
    v_points_ysw(v_move_ysw) = v_points_ysw(v_move_ysw)-v_diff_ysw(v_move_ysw);
end

if CSAD_g3_linear(Nu_d,'yawrate_sway')>0 && (Nu_d(3)>= Nu_min(3) && Nu_d(3)<= Nu_max(3))
    r_diff_ysw = r_points_ysw-Nu_d(3);
    [~,r_move_ysw] = min(abs(r_diff_ysw));
    r_points_ysw(r_move_ysw) = r_points_ysw(r_move_ysw)-r_diff_ysw(r_move_ysw);
end

vv_points_ysw =[v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw;v_points_ysw];
rr_points_ysw =[r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;r_points_ysw;
```

```
    r_points_ysw;r_points_ysw;r_points_ysw;
    r_points_ysw;r_points_ysw;r_points_ysw]';

vss_ysw = reshape(vv_points_ysw,[1,21*21]);
rss_ysw = reshape(rr_points_ysw,[1,21*21]);
Nu_r_ysw = zeros(3,21*21);
Nu_r_ysw(2,:) = vss_ysw;
Nu_r_ysw(3,:) = rss_ysw;
h_ysw = CSAD_g3_linear(Nu_r_ysw,'yawrate_sway');

Max_point=21*21;
Max_point=Max_point-sum(h_ysw<0);
Nu_f_ysw = [zeros(Max_point,1) vv_points_ysw(h_ysw>=0) rr_points_ysw(h_ysw>=0)]';

Ob_ysw = CSAD_objective_function_dw3(Nu_d,Nu_f_ysw,'yawrate_sway');

[~,vel_pair_ysw] = max(Ob_ysw(:,3));

alpha_ysw = Nu_f_ysw(:,vel_pair_ysw);

%% Sway/Surge
v_points_sws = linspace(Nu_min(2),Nu_max(2),21);
u_points_sws = linspace(Nu_min(1),Nu_max(1),21);

if CSAD_g3_linear(Nu_d, 'sway_surge')>0 && (Nu_d(2)>= Nu_min(2) && Nu_d(2)<= Nu_max(2)) && (Nu_d(1)>= Nu_min(1) && Nu_d(1)<= Nu_max(1))
    v_diff_sws = v_points_sws-Nu_d(2);
    [~,v_move_sws] = min(abs(v_diff_sws));
    v_points_sws(v_move_sws) = v_points_sws(v_move_sws)-v_diff_sws(v_move_sws);

    u_diff_sws = u_points_sws-Nu_d(1);
    [~,u_move_sws] = min(abs(u_diff_sws));
    u_points_sws(u_move_sws) = u_points_sws(u_move_sws)-u_diff_sws(u_move_sws);
end
```

```

if CSAD_g3_linear(Nu_d,'sway_surge')>0 && (Nu_d(2)>= Nu_min(2) && Nu_d(2)<= Nu_max(2))
    v_diff_sws = v_points_sws-Nu_d(2);
    [~,v_move_sws] = min(abs(v_diff_sws));
    v_points_sws(v_move_sws) = v_points_sws(v_move_sws)-v_diff_sws(v_move_sws);
end

if CSAD_g3_linear(Nu_d,'sway_surge')>0 && (Nu_d(1)>= Nu_min(1) && Nu_d(1)<= Nu_max(1))
    u_diff_sws = u_points_sws-Nu_d(1);
    [~,u_move_sws] = min(abs(u_diff_sws));
    u_points_sws(u_move_sws) = u_points_sws(u_move_sws)-u_diff_sws(u_move_sws);
end

vv_points_sws =[v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws;v_points_sws]';
uu_points_sws =[u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws;u_points_sws];

vss_sws = reshape(vv_points_sws,[1,21*21]);
uss_sws = reshape(uu_points_sws,[1,21*21]);
Nu_r_sws = zeros(3,21*21);
Nu_r_sws(2,:) = vss_sws;
Nu_r_sws(1,:) = uss_sws;
h_sws = CSAD_g3_linear(Nu_r_sws,'sway_surge');

```

```
Max_point=21*21;
Max_point=Max_point-sum(h_sws<0);
Nu_f_sws = [uu_points_sws(h_sws>=0) vv_points_sws(h_sws>=0) zeros(Max_point,1)]';

Ob_sws = CSAD_objective_function_dw3(Nu_d,Nu_f_sws,'sway_surge');

[~,vel_pair_sws] = max(Ob_sws(:,3));

alpha_sws = Nu_f_sws(:,vel_pair_sws);

%%
alpha_f_DWC = (alpha_ys + alpha_ysw + alpha_sws)/2;

alpha_DWC_dot = (alpha_f_DWC - Nu)/ts;
end
```

### B.1.3 Function to find feasible velocities, $g(u, v, r)$

```
function h = CSAD_g3_linear(Nu,switchcase)
switch switchcase
    case 'yawrate_surge'
        rr_ys = Nu(3,:)';
        uu_ys = Nu(1,:)';
        h1_ys = (rr_ys+0.117);
        h2_ys = (3.3684*rr_ys + 0.6728) - uu_ys;
        h3_ys = (-3.3684*rr_ys + 0.6728) - uu_ys;
        h4_ys = -(rr_ys-0.117);
        h5_ys = -(3.3684*rr_ys - 0.6728) + uu_ys;
        h6_ys = -(-3.3684*rr_ys - 0.6728) + uu_ys;
        h = min([h1_ys';h2_ys';h3_ys';h4_ys';h5_ys';h6_ys'])';

    case 'yawrate_sway'
        rr_ysw = Nu(3,:)';
        vv_ysw = Nu(2,:)';
        h1_ysw = (0.97171*rr_ysw + 0.1972) - vv_ysw;
        h2_ysw = (-2.3992*rr_ysw + 0.1972) - vv_ysw;
        h3_ysw = -(0.97171*rr_ysw - 0.1972) + vv_ysw;
        h4_ysw = -(-2.3992*rr_ysw - 0.1972) + vv_ysw;
        h = min([h1_ysw'; h2_ysw'; h3_ysw'; h4_ysw'])';

    case 'sway_surge'
        uu_sws = Nu(1,:)';
        vv_sws = Nu(2,:)';
        h1_sws = (vv_sws+0.1972);
        h2_sws = (3.464*vv_sws + 0.9618) - uu_sws;
        h3_sws = (1.4041*vv_sws + 0.6728) - uu_sws;
        h4_sws = (-1.4041*vv_sws + 0.6728) - uu_sws;
        h5_sws = (-3.464*vv_sws + 0.9618) - uu_sws;
        h6_sws = -(vv_sws-0.1972);
        h7_sws = -(3.464*vv_sws - 0.9618) + uu_sws;
        h8_sws = -(1.4041*vv_sws - 0.6728) + uu_sws;
        h9_sws = -(-1.4041*vv_sws - 0.6728) + uu_sws;
        h10_sws = -(-3.464*vv_sws - 0.9618) + uu_sws;
        h = min([h1_sws'; h2_sws'; h3_sws'; h4_sws';
            h5_sws'; h6_sws'; h7_sws'; h8_sws'; h9_sws';
            h10_sws'])';

    otherwise
        h = 0;
end
end
```

### B.1.4 Objective function, $G(\nu_d, \nu_f)$

```
function Ob = CSAD_objective_function_dw3(Nu_d, Nu_f,
    switchcase)
u_f = Nu_f(1,:);
u_d = Nu_d(1,:);
v_f = Nu_f(2,:);
v_d = Nu_d(2,:);
r_f = Nu_f(3,:);
r_d = Nu_d(3,:);

surge = 1 - abs(u_d - u_f)/(max(abs(u_d - u_f)));
sway = 1 - abs(v_d - v_f)/(max(abs(v_d - v_f)));
```

```
yawrate = 1 - abs(r_d - r_f)/(max(abs(r_d- r_f)));

switch switchcase
    case 'yawrate_surge'
        Ob = [yawrate' surge' yawrate'+surge'];
    case 'yawrate_sway'
        Ob = [yawrate' sway' yawrate'+sway'];
    case 'sway_surge'
        Ob = [sway' surge' sway'+surge'];
    otherwise
        Ob = [0 0 0];
end
end
```

### B.1.5 Velocity controller

```
function Tau_dw = VelocityControl(Alpha_DWC, Alpha_DWC_dot, Nu)
    [M, C, D] = CSAD_linear2(Nu);
    Tau_dw = M * Alpha_DWC_dot + D * Alpha_DWC;
end
```

## B.2 Improvements to the DW algorithm

When operating close to the boundary lines of $V_p$, in most cases the algorithm does not include feasible velocities at the boundary. This means that actuator twitching can occur as the algorithm tracks the desired velocity along the boundary lines of $V_p$. To get around this, effort has been put into adding a fix that always makes sure that there are feasible velocities along the boundary lines of $V_p$, thus removing the unwanted actuator twitching that can occur when operating at or close to maximum velocities. Here, a fix for the left and right boundaries of $V_p$ for the 2 DOF yaw rate-sway speed case is presented. Similar fixes will have to be implemented for all the other boundary lines of all three 2 DOF cases in order to properly improve performance in all possible operational scenarios.

Figure B.1 illustrates $V_p$, $V_w$, $V_f$ and $\boldsymbol{\nu}_d = [0.4\,\text{m/s}, 0.13\,\text{m/s}, 1.5\,\text{deg/s}]^\top$ given a current velocity of $\boldsymbol{\nu} = [0.3\,\text{m/s}, 0.1\,\text{m/s}, 2\,\text{deg/s}]^\top$. Note that this figure is not to scale, and is used for illustrative purposes only. The lower plot shows the whole yaw rate-sway speed window, while the upper right plot shows a zoomed-in version of the yaw rate-sway speed window were no boundary line fix is included. The upper left plot shows how the fix has added discrete solutions along the boundary line of $V_p$, effectively making it possible to move along the boundary line without causing actuator twitching.

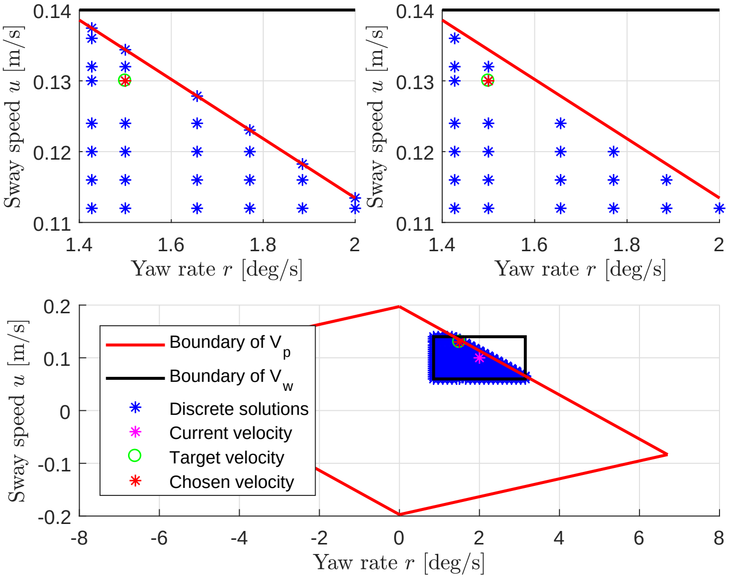


Figure B.1: Improving performance by adding feasible velocities to the boundary line of $V_p$.

### B.2.1 Improvement to movement on the boundary of $V_p$

This code should be added to the 3 DOF dynamic window algorithm, and should, for this specific example, be added at line 98 in the given code. The code is given by:

```
if sum(h<0)>0
    if Nu(3) > -9.4405e-13 && Nu(2) >= 0
        v1 = -2.3992*r_points_ysw + 0.1972;
        Nu_b = [zeros(1,21); v1; r_points_ysw];
        h2 = gDW(Nu_b,Nu_min,Nu_max);
        Nu_fb = Nu_b(:,h2>=0);
        Nu_f_ysw = [Nu_f_ysw Nu_fb];
    elseif Nu(3) > 0.0822 && Nu(2) < 0
        v1 = -2.3992*r_points_ysw + 0.1972;
        Nu_b = [zeros(1,21); v1; r_points_ysw];
        h2 = gDW(Nu_b,Nu_min,Nu_max);
        Nu_fb = Nu_b(:,h2>=0);
        Nu_f_ysw = [Nu_f_ysw Nu_fb];
    elseif Nu(3) < -0.0822 && Nu(2) > 0
        v1 = -2.3992*r_points_ysw - 0.1972;
        Nu_b = [zeros(1,21); v1; r_points_ysw];
        h2 = gDW(Nu_b,Nu_min,Nu_max);
        Nu_fb = Nu_b(:,h2>=0);
        Nu_f_ysw = [Nu_f_ysw Nu_fb];
    elseif Nu(3) < 9.4405e-13 && Nu(2) <= 0
        v1 = -2.3992*r_points_ysw + 0.1972;
        Nu_b = [zeros(1,21); v1; r_points_ysw];
        h2 = gDW(Nu_b,Nu_min,Nu_max);
        Nu_fb = Nu_b(:,h2>=0);
        Nu_f_ysw = [Nu_f_ysw Nu_fb];
    end
end
```

where the function $gDW(\boldsymbol{\nu}, \boldsymbol{\nu}_{min}, \boldsymbol{\nu}_{max})$ is given by

```
function h = gDW(Nu,Nu_min,Nu_max)
rr = Nu(3,:)';
vv = Nu(2,:)';

h1 = -(vv-Nu_max(2));
h2 = (vv-Nu_min(2));
h3 = -(rr-Nu_max(3));
h4 = (rr-Nu_min(3));

h = min([h1';h2';h3'; h4'])';

end
```

# Appendix C

# Publications

Here, the publications that are based upon the work done in this thesis are presented. First, the article ”Improvement of Ship Motion Control Using a Magnitude-Rate Saturation Model, in Proceedings of the 2nd IEEE Conference on Control Technology and Applications” [9], is presented. Then, the article ”A Dynamic Window-Based Controller for Dynamic Positioning Satisfying Actuator Magnitude Constraints, in Proceedings of the 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehciles” [6], is presented. The final version of the second article is to be submitted by July 5, 2018. The current version, as of June 11, 2018, is included here, but changes may have been done to the final version after this thesis has been handed in. For more accurate reading, it is recommended to find the final version elsewhere, but the current version is included here because, as of June 11, 2018, the article is yet to be available online and might be of interest to the early readers of this thesis.

# Improvement of Ship Motion Control Using a Magnitude-Rate Saturation Model

Ole Nikolai Lyngstadaas, Tore Egil Sæterdal, Mikkel Eske Nørgaard Sørensen, Morten Breivik

***Abstract*— Motion control concepts for ships have traditionally not focused on handling actuator constraints. This paper investigates the effects on performance of a pair of nonlinear control schemes by developing and implementing a magnitude-rate saturation (MRS) model. The effects of using the MRS model is tested in experiments with a model ship in an ocean basin. Performance metrics are used to evaluate performance in terms of control error, energy efficiency, and actuator wear and tear.**

***Index Terms*— Ship motion control, Magnitude-rate saturation model, Constraint handling, Nonlinear control, Model-scale experiments, Wear and tear**

## I. INTRODUCTION

In traditional control theory, an ideal controller might achieve perfect reference tracking in simulations, having no or non-sufficient limitations on the control input. However, in real-life applications it would not be feasible due to limitations in physical output and wear and tear of the actuators.

Several ways of handling actuator constraints have been investigated throughout the years. In [1], model predictive control for systems with actuator magnitude and rate constraints is presented. A solution using a modified dynamic window approach to handle actuator constraints is investigated in [2], and further expanded in [3].

To easily include magnitude and rate saturation (MRS) effects into a control system, a possible low-level approach is to limit the output of the control signal within the limits of the actuators. However, this may lead to an under-damped closed-loop system. To avoid this, effort has been put into implementing a model for combining MRS to smoothen the control output within allowed actuator limits. In [4], an MRS model is derived to address the issue of anti-windup, and the MRS model used in this paper is based on this approach.

In particular, the magnitude and rate saturations in this paper are set at lower limits than the actual actuator constraints. The main purpose is to investigate how limiting the actuator's magnitude and rate outputs will impact the overall performance of the motion control system. The MRS model, depending on how it is tuned, can be implemented in a simulation scenario, where the purpose is to mimic the actual constraints of the system, or be used to limit actuator outputs in laboratory experiments and on-board actual vessels.

The main contribution of this paper are the experimental results from scale testing on a 1:90 ship model. The MRS model from [4] is adapted to a three degrees of freedom (DOF) ship model and experimentally tested at the Marine Cybernetics Laboratory (MC-Lab) at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway. Furthermore, the positive effects of employing MRS to a pair of nonlinear feedback control schemes from [5] have been investigated.

The rest of this paper is organized as follows: Section II presents a mathematical ship model; Section III defines the control objective and the 4-corner test, derivation of the MRS model, and also presents a pair of nonlinear controllers from [5]; Section IV presents the experimental results from model-scale testing in the MC-Lab, while Section V concludes the paper.

## II. SHIP MODEL

The motion of a ship can be represented by the pose vector $\boldsymbol{\eta} = [x, y, \psi]^\top \in \mathbb{R}^2 \times \mathbb{S}$ and the velocity vector $\boldsymbol{\nu} = [u, v, r]^\top \in \mathbb{R}^3$. Here, $(x, y)$ represents the Cartesian position in the local earth-fixed reference frame, $\psi$ is the yaw angle, $(u, v)$ represents the body-fixed linear velocities and $r$ is the yaw rate. The 3-DOF dynamics of a ship can then be stated as in [6]:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{1}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}, \tag{2}$$

where $\boldsymbol{M} \in \mathbb{R}^{3\times3}$, $\boldsymbol{C}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$, $\boldsymbol{D}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3]^\top$ represent the inertia matrix, Coriolis and centripetal matrix, damping matrix and control input vector, respectively. The rotation matrix $\boldsymbol{R}(\psi) \in SO(3)$ is given by

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The system matrices are assumed to satisfy the properties $\boldsymbol{M} = \boldsymbol{M}^\top > 0$, $\boldsymbol{C}(\boldsymbol{\nu}) = -\boldsymbol{C}(\boldsymbol{\nu})^\top$ and $\boldsymbol{D}(\boldsymbol{\nu}) > 0$.

### A. Nominal model

The model and parameters of the model-scale ship C/S Inocean Cat I Drillship (CSAD) [7], as shown in Fig. 1, will be used in this paper. CSAD is a 1:90 scale replica of a supply ship, with a length of $L = 2.578$ m. The inertia matrix is given as

$$\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_A, \tag{4}$$

O. N. Lyngstadaas and T. E. Sæterdal are M.Sc. students at the Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway. M. E. N. Sørensen and M. Breivik are with the Centre for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway. Email: {mikkel.e.n.sorensen, morten.breivik}@ieee.org

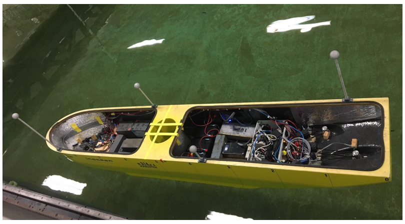


Fig. 1: C/S Inocean Cat I Drillship in the MC-lab.

where

$$\boldsymbol{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix} \tag{5}$$

$$\boldsymbol{M}_{A} = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix}. \tag{6}$$

The mass of CSAD is $m = 127.92$ kg, while $x_g = 0.00375$ m is the distance along the $x$-axis in the body frame from the centre of gravity, and $I_z = 61.987$ kg m$^2$ is the moment of inertia about the $z$-axis in the body frame. Other parameter values are listed in Table I, which are updated values from [7], where a few changes to the numerical values and signs have been done to better fit the actual laboratory performance of CSAD.

CSAD has six azimuth thrusters, which in the experiments presented here are fixed to the angles
$\boldsymbol{\delta} = [\pi, \pi/4, -\pi/4, 0, 5\pi/4, 3\pi/4]^\top$ rad, in the body-fixed coordinate system, giving a fully actuated vessel [3].

The Coriolis and centripetal matrix is

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_{A}(\boldsymbol{\nu}), \tag{7}$$

with

$$\boldsymbol{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \tag{8}$$

$$\boldsymbol{C}_{A}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -c_{A,13}(\boldsymbol{\nu}) \\ 0 & 0 & c_{A,23}(\boldsymbol{\nu}) \\ c_{A,13}(\boldsymbol{\nu}) & -c_{A,23}(\boldsymbol{\nu}) & 0 \end{bmatrix}, \tag{9}$$

where

$$c_{A,13}(\boldsymbol{\nu}) = -Y_{\dot{r}} r - Y_{\dot{v}} v \tag{10}$$

$$c_{A,23}(\boldsymbol{\nu}) = -X_{\dot{u}} u. \tag{11}$$

Finally, the damping matrix $\boldsymbol{D}(\boldsymbol{\nu})$ is given as

$$\boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{D}_L + \boldsymbol{D}_{NL}(\boldsymbol{\nu}), \tag{12}$$

where

$$\boldsymbol{D}_L = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \tag{13}$$

TABLE I: Parameters for CSAD, updated from [7].

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| $X_{\dot{u}}$ | $-3.262$ | $Y_{\|r\|r}$ | $-3.450$ |
| $Y_{\dot{v}}$ | $-28.890$ | $Y_{rrr}$ | $0$ |
| $Y_{\dot{r}}$ | $-0.525$ | $N_r$ | $-6.916$ |
| $N_{\dot{v}}$ | $-0.157$ | $N_{\|r\|r}$ | $-4.734$ |
| $N_{\dot{r}}$ | $-13.980$ | $N_{rrr}$ | $-0.147$ |
| $X_u$ | $-2.332$ | $N_v$ | $0$ |
| $X_{\|u\|u}$ | $0$ | $N_{\|v\|v}$ | $-0.209$ |
| $X_{uuu}$ | $-8.557$ | $N_{vvv}$ | $0$ |
| $Y_v$ | $-4.673$ | $N_{\|r\|v}$ | $0.080$ |
| $Y_{\|v\|v}$ | $-0.398$ | $N_{\|v\|r}$ | $0.080$ |
| $Y_{vvv}$ | $-313.300$ | $Y_{\|r\|v}$ | $-0.805$ |
| $Y_r$ | $-7.250$ | $Y_{\|v\|r}$ | $-0.845$ |

$$\boldsymbol{D}_{NL}(\boldsymbol{\nu}) = \begin{bmatrix} d_{NL,11}(\boldsymbol{\nu}) & 0 & 0 \\ 0 & d_{NL,22}(\boldsymbol{\nu}) & d_{NL,23}(\boldsymbol{\nu}) \\ 0 & d_{NL,32}(\boldsymbol{\nu}) & d_{NL,33}(\boldsymbol{\nu}) \end{bmatrix}, \tag{14}$$

with

$$d_{NL,11}(\boldsymbol{\nu}) = -X_{|u|u}|u| - X_{uuu}u^2 \tag{15}$$

$$d_{NL,22}(\boldsymbol{\nu}) = -Y_{|v|v}|v| - Y_{|r|v}|v| - Y_{vvv}v^2 \tag{16}$$

$$d_{NL,23}(\boldsymbol{\nu}) = -Y_{|r|r}|r| - Y_{|v|r}|v| - Y_{rrr}v^2 - Y_{ur}u \tag{17}$$

$$d_{NL,32}(\boldsymbol{\nu}) = -N_{|v|v}|v| - N_{|r|v}|r| - N_{vvv}v^2 - N_{uv}u \tag{18}$$

$$d_{NL,33}(\boldsymbol{\nu}) = -N_{|r|r}|v| - N_{|v|r}|v| - N_{rrr}r^2 - N_{ur}u, \tag{19}$$

where

$$Y_{ur} = X_{\dot{u}} \tag{20}$$

$$N_{uv} = -(Y_{\dot{v}} - X_{\dot{u}}) \tag{21}$$

$$N_{ur} = Y_{\dot{r}}, \tag{22}$$

which are damping terms which are linearly increasing with the forward speed. These are added to compensate for the Munk moment, and to get a more physically realistic model behavior [2], [8].

## III. CONTROL DESIGN

### A. Control objective and 4-corner test

The main control objective is to make $\tilde{\boldsymbol{\eta}}(t) \triangleq \boldsymbol{\eta}(t) - \boldsymbol{\eta}_t(t) \rightarrow \mathbf{0}\ t \rightarrow \infty$, where $\boldsymbol{\eta}_t(t) = [x_t(t), y_t(t), \psi_t(t)]^\top \in \mathbb{R}^2 \times \mathbb{S}$ represents the pose associated with a target point. The motion of the target is typically defined by a human or generated by a guidance system. For notational simplicity, time $t$ will mostly be omitted for the rest of the paper.

It is desirable to investigate the effect of the magnitude-rate saturation model during different ship maneuvers. For this reason, a 4-corner maneuvering test is used, as shown in Fig. 2. For comparison, the experiments will be conducted with and without using the MRS model to identify notable effects on performance.

The 4-corner maneuvering test is proposed in [9] as a way to compare ship performance of dynamic positioning control algorithms. The ship is first initialized in dynamic positioning to point straight North at heading $0\,(deg)$. Then the following setpoint changes are commanded:

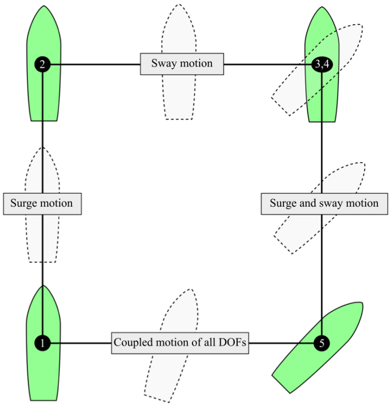


Fig. 2: The 4-corner dynamic positioning test. Modified from [9].

- Position change $2\,(m)$ straight North: tests a pure surge movement ahead.
- Position change $2\,(m)$ straight East: tests a pure sway movement in the starboard direction.
- Heading change $45\,(deg)$ clockwise: tests a pure yaw motion while keeping position steady.
- Position change $2\,(m)$ straight South: tests a combined surge-sway movement while keeping heading steady.
- Position change $2\,(m)$ straight West and heading change $45\,(deg)$ counterclockwise: tests a combined surge-sway-yaw movement.

### *B. Magnitude-rate saturation model design*

Modelling the vessel's actuator constraints is important to ensure that the controller output remains inside a feasible range of values. Both magnitude and rate constraints will impact a vessel's ability to maneuver, and should be handled in the control system.

*1) Saturation modeling:* A generalized saturation block for an actuator can be modeled as

$$\tau_{s,i}(\tau_i) = \begin{cases} \tau_{i,min} & \text{if } \tau_i \leq \tau_{i,min} \\ \tau_i & \text{if } \tau_{i,min} < \tau_i < \tau_{i,max}, \quad \forall i \in \{1,2,3\}, \\ \tau_{i,max} & \text{if } \tau_i \geq \tau_{i,max} \end{cases} \tag{23}$$

where $\tau_i$ is the commanded control input without saturation with $i \in \{1,2,3\}$ to control surge, sway and yaw forces and moment, respectively. The saturation limits are represented by $\boldsymbol{\tau}_{min} = [\tau_{1,min}, \tau_{2,min}, \tau_{3,min}]^\top$ and $\boldsymbol{\tau}_{max} = [\tau_{1,max}, \tau_{2,max}, \tau_{3,max}]^\top$ with negative and positive bounded elements, respectively.

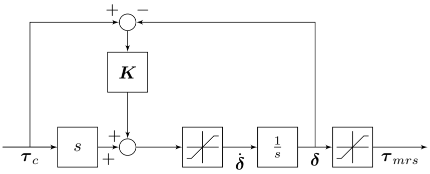


Fig. 3: Block diagram for the MRS model (24).

*2) Magnitude-rate saturation model:* An approach to model the MRS effects is given by

$$\dot{\boldsymbol{\delta}} = sat_{\boldsymbol{r}}(\dot{\boldsymbol{\tau}}_c + \boldsymbol{K}(\boldsymbol{\tau_c} - \boldsymbol{\delta})) \tag{24}$$

$$\boldsymbol{\tau}_{mrs} = sat_{\boldsymbol{m}}(\boldsymbol{\delta}), \tag{25}$$

where $\boldsymbol{\tau}_c$, $\boldsymbol{\delta}$ and $\boldsymbol{\tau}_{mrs}$ are the input, state and output of the MRS model, respectively, and where $\boldsymbol{K} > 0$ is a diagonal tuning matrix. The matrix is introduced in order to avoid an unstable cancellation between the derivative operator $s$ and the integrator in Fig. 3, where the block diagram for the MRS model is shown. Because of this, an important observation is that neither of the elements of the matrix $\boldsymbol{K}$ can be equal to 1, and thus also $\boldsymbol{K} \neq \boldsymbol{I}$. The gain matrix $\boldsymbol{K}$ affects the speed of the inner-loop in the MRS model, and should be chosen based on the desired tracking performance. The derivative of the input, $\dot{\boldsymbol{\tau}}_c$, is supposed to exist and can be calculated using numerical derivation. The saturation limits $sat_{\boldsymbol{r}}$ and $sat_{\boldsymbol{m}}$ are modeled as the saturation block above, and contain the vessel's rate and magnitude constraints, respectively. See [4] for further details.

In this setup, the rate is limited first and the magnitude next, meaning that the MRS model state $\boldsymbol{\delta}$ can exceed the magnitude-bounds vector $\boldsymbol{m}$, although the output $\boldsymbol{\tau}_{mrs}$ never does. It should also be noted that this model can be further extended to effectively solve anti-windup problems, should such effects be needed to be accounted for.

### *C. Nonlinear control design*

The MRS model will be tested with two types of feedback controllers in order to investigate the impact on performance for both linear and nonlinear feedback terms.

*1) Nonlinear pose and linear velocity feedbacks:* Using a control scheme based on a combination of nonlinear feedback of pose and linear feedback of velocity from [5], the control input can be chosen as

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\alpha} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\alpha} - \boldsymbol{\Gamma}_2\boldsymbol{z}_2, \tag{26}$$

where

$$\dot{\boldsymbol{\alpha}} = \boldsymbol{R}^\top(\psi)\ddot{\boldsymbol{\eta}}_t + \boldsymbol{S}(r)^\top\boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_t - \dot{\boldsymbol{K}}_1(\cdot)\boldsymbol{z}_1 - \boldsymbol{K}_1(\cdot)\dot{\boldsymbol{z}}_1, \tag{27}$$

with $\mathbf{\Gamma}_2 > 0$ and where

$$\boldsymbol{S}(r) = \begin{bmatrix} 0 & -r & 0 \\ r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{28}$$

Here, the error variables $\boldsymbol{z}_1 = [z_{1,x}, z_{1,y}, z_{1,\psi}]^\top$ and $\boldsymbol{z}_2 = [z_{2,u}, z_{2,v}, z_{2,r}]^\top$ are defined as

$$\boldsymbol{z}_1 \overset{\triangle}{=} \boldsymbol{R}(\psi)(\boldsymbol{\eta} - \boldsymbol{\eta}_t) \tag{29}$$

$$\boldsymbol{z}_2 \overset{\triangle}{=} \boldsymbol{\nu} - \boldsymbol{\alpha}, \tag{30}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^3$ is a vector of stabilizing functions, which can be interpreted as a desired velocity. As in [5], $\boldsymbol{\alpha}$ can be chosen as

$$\boldsymbol{\alpha} = \boldsymbol{R}^\top(\psi)\dot{\boldsymbol{\eta}}_{\boldsymbol{t}} - \boldsymbol{K}_1(\cdot)\boldsymbol{z}_1, \tag{31}$$

with the nonlinear feedback term $\boldsymbol{K}_1(\cdot)$ chosen as

$$\boldsymbol{K_1}(\cdot) = \mathbf{\Gamma}_1 \begin{bmatrix} \dfrac{1}{\sqrt{\boldsymbol{z}_{1,\tilde{p}}^\top \boldsymbol{z}_{1,\tilde{p}} + \Delta_{\tilde{p}}^2}} \boldsymbol{I}_{2\times2} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \dfrac{1}{\sqrt{z_{1,\tilde{\psi}}^2 + \Delta_{\tilde{\psi}}^2}} \end{bmatrix}, \tag{32}$$

where $\boldsymbol{z}_{1,\tilde{p}} = [z_{1,x}, z_{1,y}]^\top$, $\mathbf{\Gamma}_1 > 0$ and $\Delta_i > 0$ are tuning parameters. Furthermore, $\dot{\boldsymbol{K}}_1(\cdot)$ is given by

$$\dot{\boldsymbol{K}}_1(\cdot) = -\mathbf{\Gamma}_1 \begin{bmatrix} \dfrac{\boldsymbol{z}_{1,\tilde{p}}^\top \dot{\boldsymbol{z}}_{1,\tilde{p}}}{(\boldsymbol{z}_{1,\tilde{p}}^\top \boldsymbol{z}_{1,\tilde{p}} + \Delta_{\tilde{p}}^2)^{\frac{3}{2}}} \boldsymbol{I}_{2\times2} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \dfrac{z_{1,\tilde{\psi}} \dot{z}_{1,\tilde{\psi}}}{(z_{1,\tilde{\psi}}^2 + \Delta_{\tilde{\psi}}^2)^{\frac{3}{2}}} \end{bmatrix}. \tag{33}$$

*2) Nonlinear pose and velocity feedbacks:* The other control scheme from [5] augments (26) with a nonlinear velocity feedback term, giving the control input

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\alpha} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\alpha} - \boldsymbol{K}_2(\cdot)\boldsymbol{z}_2, \tag{34}$$

where $\dot{\boldsymbol{\alpha}}$ and $\boldsymbol{\alpha}$ are given by (27) and (31), respectively, and with the nonlinear feedback term $\boldsymbol{K}_2(\cdot)$ chosen as

$$\boldsymbol{K_2}(\cdot) = \mathbf{\Gamma}_2 \begin{bmatrix} \dfrac{1}{\sqrt{\boldsymbol{z}_{2,\tilde{\nu}^\top} \boldsymbol{z}_{2,\tilde{v}} + \Delta_{\tilde{v}}^2}} \boldsymbol{I}_{2\times2} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \dfrac{1}{\sqrt{z_{2,\tilde{r}^2} + \Delta_{\tilde{r}}^2}} \end{bmatrix}, \tag{35}$$

where $\boldsymbol{z}_{2,\tilde{v}} = [z_{2,u}, z_{2,v}]^\top$ and $\Delta_i > 0$ are tuning parameters. The feedback gain $\mathbf{\Gamma}_2$ is the same matrix as in (26).

The nonlinear pose and linear velocity feedback controller and the nonlinear pose and velocity feedback controller will be abbreviated NP-LV and NP-NV, respectively, throughout the rest of this paper.

TABLE II: Control gains.

| | **NP-LV** | **NP-NV** |
|---|---|---|
| $\mathbf{\Gamma}_1$ | $\text{diag}([0.08, 0.08, 0.0698])$ | $-\|\|-$ |
| $\mathbf{\Gamma}_2$ | $\text{diag}([0.2, 0.2, 0.1745])\boldsymbol{M}$ | $-\|\|-$ |
| $\Delta_{\tilde{p}}$ | $0.5$ | $-\|\|-$ |
| $\Delta_{\tilde{\psi}}$ | $0.5$ | $-\|\|-$ |
| $\Delta_{\tilde{v}}$ | $-$ | $0.7$ |
| $\Delta_{\tilde{r}}$ | $-$ | $1$ |
| $\boldsymbol{K}$ | $\text{diag}([4, 3, 2])$ | $-\|\|-$ |

*3) Stability:* Based on the theorems and stability proofs in [10], we can conclude that the two controllers have the following stability properties: The origin $(\boldsymbol{z}_1, \boldsymbol{z}_2) = (\mathbf{0}, \mathbf{0})$ is uniformly globally asymptotically stable (UGAS) and on each compact set $B \subset \mathbb{R}^6$ containing the origin, it is uniformly exponentially stable (UES) [10]. The MRS model is a nonlinear filter, and it is proven in [4] that the output will be an $\mathcal{L}_2$ signal if the input is an $\mathcal{L}_2$ signal, so it can be concluded that the MRS model does not alter the stability properties of the system.

*4) Parameter tuning:* The experiments are conducted with the gain parameters shown in Table II. The choice of the gain parameters for the two controllers are based on the tuning rules described in [10]. Here, the goal is to make the kinetic subsystem faster than the kinematic subsystem, which means that the kinetic subsystem needs to have smaller time constants than the kinematic subsystem in the linear region. The $\Delta$-values scale the linear feedback gains and therefore the resulting time constants of the linear region, and must therefore be chosen such that they do not make the kinematic subsystem faster than the kinetic subsystem.

The actuator saturation limits are chosen by the following set of suggested tuning rules as well [11]. Here, the magnitude saturation limits are set lower than the actual limitations in order to save energy, and chosen as $\boldsymbol{m} = [2, 1.5, 1]$. The rate saturation limits are chosen by $\boldsymbol{r} = [m_1/t_{mrs,1}, m_2/t_{mrs,2}, m_3/t_{mrs,3}]^\top$, where $m_1$, $m_2$ and $m_3$ are the magnitude saturation limits given by $\boldsymbol{m}$, and where $t_{mrs,1}$, $t_{mrs,2}$ and $t_{mrs,3}$ are the desired transition times for the actuators to go from zero to max thrust in surge, sway and yaw, respectively. Here, suitable values for the rate saturation limits were found to be $\boldsymbol{r} = [1.9, 1.1, 0.8]$. Then, the gain matrix $\boldsymbol{K}$ can be chosen by $\boldsymbol{K} = \text{diag}([K_{1,1}, \frac{m_2}{m_1}K_{1,1}, \frac{m_3}{m_1}K_{1,1}])$, where under normal operations it is desired to have all the diagonal elements $\boldsymbol{K}_{i,i} > 1$, $\forall i \in \{1, 2, 3\}$. Here, $K_{1,1} = 4$ to ensure a fast tracking of the target signal in all three degrees of freedom. The block diagram for the full control system is shown in Fig. 4.

## IV. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

### A. Marine Cybernetics Laboratory

As already mentioned, the MC-Lab is a small ocean basin at NTNU. Due to its relatively small size and advanced instrumentation package, the facility is especially suited for

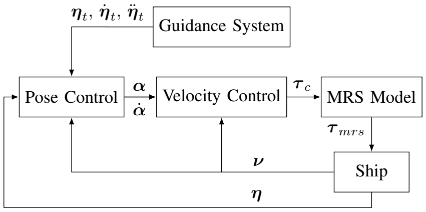


Fig. 4: Block diagram for the ship control system.

tests of motion control systems for marine vessel models, but is also suitable for more specialized hydrodynamic tests due to the advanced towing carriage, which has capability for precise movement of models up to six degrees of freedom [12].

The experiments will be conducted under the following conditions: In the experiments, the actual model ship's $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ matrices will differ somewhat from those used in the controllers. Also, measurement noise is present in the Qualisys motion tracking system used in the laboratory.

### B. Performance metrics

Performance metrics are used to objectively compare the performance of different control schemes. In this paper, the error variable is defined as the scaled norm of the pose control error $\boldsymbol{z}_1$, such that

$$e = \sqrt{\bar{\boldsymbol{z}}_1^\top \bar{\boldsymbol{z}}_1}, \tag{36}$$

where

$$\bar{\boldsymbol{z}}_1 = [\frac{z_{1,x}}{4}, \frac{z_{1,y}}{4}, \frac{z_{1,\psi}}{\pi/2}]^\top. \tag{37}$$

Since the position and yaw angle in pose have different units, we have defined the normalized pose error signals $\bar{z}_{1,x}$, $\bar{z}_{1,y}$ and $\bar{z}_{1,\psi}$ on the intervals $[-0.5, 0.5]$ in the expected operational space of the ship [13]. To get this interval, the position errors are divided by $4$ and the yaw error is divided by $\frac{\pi}{2}$, since the position errors are in the intervals $[-2, 2]$ and the yaw error is in the interval $[-\frac{\pi}{4}, \frac{\pi}{4}]$, resulting in the normalized control error $e$.

Three different performance metrics are used in this paper, namely IAE, IAEW and IADC. The IAE (integral of the absolute error) metric is defined as an unweighted integral over time:

$$IAE(t) = \int_0^t |e(\gamma)| \, d\gamma. \tag{38}$$

The IAEW (integral of the absolute error multiplied by energy consumption) metric scales IAE by the energy consumption

$$IAEW(t) = \int_0^t |e(\gamma)| \, d\gamma \int_0^t P(\gamma) d\gamma, \tag{39}$$

where $P = |\boldsymbol{\nu}^\top \boldsymbol{\tau}|$, thus yielding a measure of energy efficiency.

Since the aim of the MRS model is also to reduce actuator wear and tear, it is interesting to investigate the dynamic behavior of the control signal. The IADC (integral of absolute differentiated control) metric is defined as in [13]:

$$IADC(t) = \int_{t_0}^t |\dot{\bar{\tau}}(\gamma)| \, d\gamma, \tag{40}$$

with $\bar{\tau}(t) = \sqrt{\boldsymbol{\tau}^\top \boldsymbol{\tau}}$, and where $\dot{\bar{\tau}}$ is computed using numerical derivation.

### C. Experimental results

In the experiments, the target pose changes between setpoints for the 4-corner test. The system is implemented such that the target will automatically change to the next setpoint when the ship is within $0.003$ m from the target in both $x$ and $y$ direction and $0.2$ deg from the target heading. When the 4-corner test is completed, the ship will have returned accurately to its initial position and heading, ready for a new test at the same pose and along the same track.

While CSAD has a length of $L = 2.578$ m, its outline has been scaled by 1:6 in the 4-corner plots in Fig. 5 and 8, to better display the ship behaviour. By the plotted values of the performance metrics in Fig. 6 and 9, the effects of the MRS model on control performance can be examined. Fig. 5 shows the 4-corner track and the actual trajectory for the CSAD with and without the MRS model applied to the NP-LV controller. The results show no remarkable difference in the trajectory.

The performance metrics are plotted in Fig. 6. The metrics show that the while MRS does not reduce the overall tracking error by the IAE metric, both energy consumption (IAEW) and actuator wear and tear (IADC) are reduced by 6.8% and 38.8%, respectively.

In Fig. 7, the commanded thrust signals are shown for the 4-corner test. It can be seen that the MRS contributes to a smoother and amplitude-wise smaller control signal, while achieving approximately the same tracking performance. The spikes that can be seen in the control signal, especially during transients, are caused by noise related to the velocity estimation.

Fig. 8 displays the 4-corner trajectory for the NP-NV controller. Even though the NP-NV-controlled vessel with MRS effects takes a wider arch in the coupled motion (5 → 1) in Fig. 2, the overall tracking error is not increased, as seen in Table III.

Furthermore, Fig. 9 shows improvement in energy efficiency, shown by the IAEW metric, and lower actuator wear and tear through the IADC metric. The reduction is greater for the NP-NV controller than the NP-LV controller, which is due to the fact the NP-NV is inherently a more aggressive controller, and thus benefits more from using an MRS model. For the NP-NV controller, the reduction is 12.2% and 46.4% for IAEW and IADC, respectively.

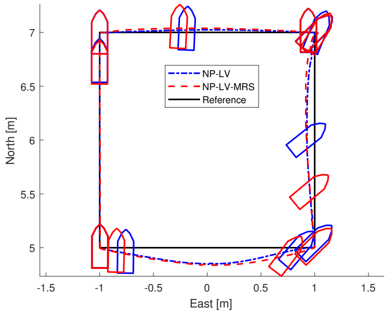


Fig. 5: Vessel performing the 4-corner manoeuver using the NP-LV controller.

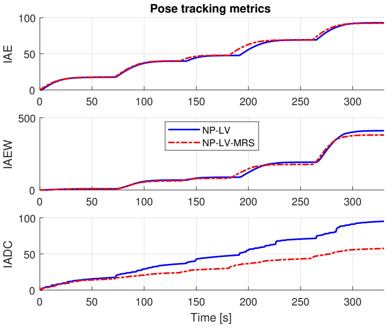


Fig. 6: Performance metrics for NP-LV.

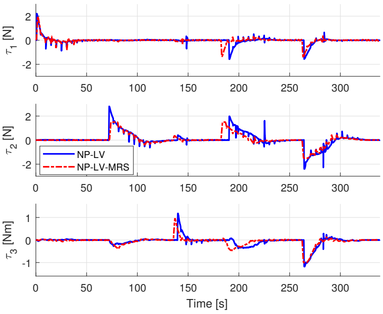


Fig. 7: Commanded control input for NP-LV.

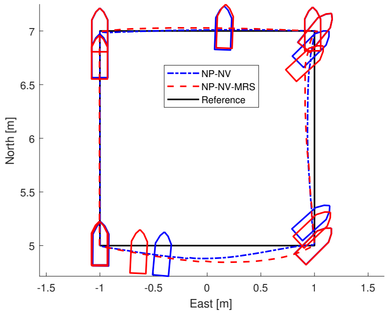


Fig. 8: Vessel performing the 4-corner maneuver using the NP-NV controller.

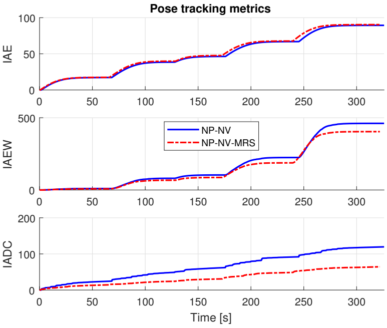


Fig. 9: Performance metrics for NP-NV.

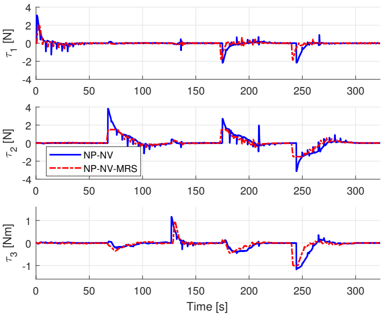


Fig. 10: Commanded control input for NP-NV.

TABLE III: Performance metrics final values.

| | NP-LV | NP-LV MRS | NP-NV | NP-NV MRS |
|---|---|---|---|---|
| **IAE final** | 92.99 | **92.85** | **89.49** | 90.38 |
| **IAEW final** | 410.12 | **382.23** | 460.08 | **403.79** |
| **IADC final** | 95.99 | **58.73** | 118.26 | **63.34** |

Fig. 10 shows the commanded control inputs for the NP-NV controller. Similar to NP-LV, a smoothing effect can be observed, although less significant. This is likely due to the nature of the pure nonlinear feedback, giving overall better tracking performance, which has previously been discussed in [5].

A significant effect of the MRS model, which can be seen in the performance metrics in Fig. 6 and 9, is that it results in a significantly reduced rate of change in the commanded control input.

The final values for the performance metrics are displayed in Table III, where the best performing controller for the different metrics is noted in bold.

## V. CONCLUSION

Depending on the type of controller that is being used, including an MRS model to limit the actuator magnitude and rate outputs can contribute positively in several ways. As seen in both cases presented, an MRS model can effectively reduce actuator twitching, and thus wear and tear, without the degradation of performance in ship control. In addition, it has the potential to improve overall energy efficiency and pose tracking abilities, as can be seen from the performance metrics and trajectory plots, and can thus have positive effects on ship performance in setpoint navigation. These effects are especially important for vessels which must operate for long times at sea, and can be particularly useful for ships in DP operations, effectively contributing to the longevity of the operation with a reduced need for maintenance and repairs.

Future work includes optimizing the MRS model to further improve performance. This includes, through experimental tests in a laboratory, further tuning of the gain matrix $\boldsymbol{K}$ and the desired magnitude and rate saturation effects to obtain optimal ship control for the wanted ship operational environment.

## ACKNOWLEDGEMENTS

This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, project number 223254. The authors gratefully acknowledge senior engineer Torgeir Wahl and Ph.D. candidate Andreas R. Dahl at NTNU's Department of Marine Technology for valuable support during the experiments.

## REFERENCES

[1] V. Kapila and S. Valluri, "Model predictive control of systems with actuator amplitude and rate saturation," *in Proceedings of the 37th IEEE Conference on Decision and Control, pp. 1396-1401*, 1998.

[2] M. E. N. Sørensen, M. Breivik, and B.-O. H. Eriksen, "A ship heading and speed control concept inherently satisfying actuator constraints," *in Proceedings of the 1st IEEE Conference on Control Technology and Applications, HI, USA*, 2017.

[3] M. E. N. Sørensen, O. N. Lyngstadaas, B.-O. H. Eriksen, and M. Breivik, "A dynamic window-based controller for dynamic positioning satisfying actuator magnitude constraints," *in Proceedings of the 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Opatija, Croatia*, 2018.

[4] S. Galeani, S. Onori, A. R. Teel, and L. Zaccarian, "A magnitude and rate saturation model and its use in the soluton of a static anti-windup problem," *Systems & Control Letters, Volume 57, Issue 1, pp. 1-9*, 2008.

[5] M. E. N. Sørensen and M. Breivik, "Comparing combinations of linear and nonlinear feedback terms for motion control of marine surface vessels," *in Proceedings of the 10th IFAC Conference on Control Applications in Marine Systems, Trondheim, Norway*, 2016.

[6] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.

[7] J. Bjørnø, *Thruster-Assisted Position Mooring of C/S Inocean Cat I Drillship*. Master thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2016.

[8] J. E. Refsnes, *Nonlinear Model-Based Control of Slender Body AUVs*. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2008.

[9] R. Skjetne, M. E. N. Sørensen, M. Breivik, S. A. T. Værnø, A. H. Brodtkorb, A. J. Sørensen, Ø. K. Kjerstad, V. Calabrò, and B. O. Vinje, "AMOS DP research cruise 2016: Academic full-scale testing of experimental dynamic positioning control algorithms onboard R/V Gunnerus," *in Proceedings of the 36th International Conference on Ocean, Offshore and Arctic Engineering*, 2017.

[10] M. E. N. Sørensen, M. Breivik, and R. Skjetne, "Comparing combinations of linear and nonlinear feedback terms for ship motion control," *submitted to IEEE Transactions on Control Systems Technology*, 2018.

[11] O. N. Lyngstadaas, *Ship Motion Control Concepts Considering Actuator Constraints*. Master thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2018.

[12] "Marine cybernetics laboratory." https://www.ntnu.edu/imt/lab/cybernetics. Accessed: 2018-01-30.

[13] B.-O. H. Eriksen and M. Breivik, *Modeling, Identification and Control of High-Speed ASVs: Theory and Experiments*, pp. 407–431. Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles, Springer International Publishing, 2017.

# A Dynamic Window-Based Controller for Dynamic Positioning Satisfying Actuator Magnitude Constraints

**Mikkel Eske Nørgaard Sørensen, Ole Nikolai Lyngstadaas,
Bjørn-Olav H. Eriksen and Morten Breivik**

*Centre for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway*
*E-mail: {mikkel.e.n.sorensen, bjorn-olav.h.eriksen, morten.breivik}@ieee.org, olenl@stud.ntnu.no*

**Abstract:** This paper considers the use of a simplified dynamic window (DW) algorithm to handle actuator magnitude constraints for a 3 degrees-of-freedom dynamic positioning controller for ships. To accomplish this, we use the simplified DW algorithm to design a dynamic window-based controller (DWC) which guarantees that the velocities remain within a feasible set, while simultaneously respecting the actuator constraints. The DWC is compared with a benchmark motion controller which uses nonlinear position and velocity feedback terms. The comparison is made using performance metrics which consider both control accuracy and energy efficiency.

## 1. INTRODUCTION

Numerous motion controllers and autopilots have been proposed over the years. However, many control algorithms found in the literature do not explicitly consider saturation constraints for the actuators. Examples of traditional motion control designs for ships are given in (Fossen, 2011). Not considering actuator constraints may lead to unsatisfying performance or stability issues.

In (Fox et al., 1997), the dynamic window (DW) algorithm is suggested as a method to perform collision avoidance and deal with actuator constraints imposed by limited velocities and accelerations for mobile robots. The DW algorithm is modified for AUVs in (Eriksen et al., 2016), and shows promising results for handling actuator magnitude and rate constraints. In (Sørensen et al., 2017), a simplification of this algorithm is developed for a 2 degrees-of-freedom (DOF) heading and speed controller, by removing the collision avoidance part of the algorithm. In (Sørensen et al., 2017), this DW-based controller (DWC) is combined with a motion controller based on the design in (Sørensen and Breivik, 2016).

The contribution of this paper is the extension of the 2 DOF DWC presented in (Sørensen et al., 2017) to a 3 DOF DWC suitable for dynamic positioning (DP). The 3 DOF DWC is compared with a benchmark controller (BC) from (Sørensen and Breivik, 2016), where the comparison is made using performance metrics which consider both control accuracy and energy efficiency.

The structure of the paper is as follows: A mathematical ship model is presented in Section 2; Section 3 describes the assumptions and control objective; Section 4 presents the design of a benchmark controller inspired by backstepping and constant-bearing guidance; Section 5 presents the proposed DWC concept; Section 6 presents simulation results, while Section 7 concludes the paper.

## 2. SHIP MODEL

The motion of a ship can be represented by the pose vector $\boldsymbol{\eta} = [x, y, \psi]^\top \in \mathbb{R}^2 \times \mathbb{S}$ and the velocity vector $\boldsymbol{\nu} = [u, v, r]^\top \in \mathbb{R}^3$. Here, $(x, y)$ represents the Cartesian position in a local earth-fixed reference frame, $\psi$ is the yaw angle, $(u, v)$ represents the body-fixed linear velocities and $r$ is the yaw rate. The 3 DOF dynamics of a ship can then be stated as (Fossen, 2011):

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{1}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}, \tag{2}$$

where $\boldsymbol{M} \in \mathbb{R}^{3\times3}$, $\boldsymbol{C}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$, $\boldsymbol{D}(\boldsymbol{\nu}) \in \mathbb{R}^{3\times3}$ and $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3]^\top$ represent the inertia matrix, Coriolis and centripetal matrix, damping matrix and control input vector, respectively. The rotation matrix $\boldsymbol{R}(\psi) \in SO(3)$ is given as

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The system matrices are assumed to satisfy the properties $\boldsymbol{M} = \boldsymbol{M}^\top > 0$, $\boldsymbol{C}(\boldsymbol{\nu}) = -\boldsymbol{C}(\boldsymbol{\nu})^\top$ and $\boldsymbol{D}(\boldsymbol{\nu}) > 0$. In this paper, we use the model and parameters of the model-scale ship CyberShip Inocean Cat I Arctic Drillship (CSAD) (Bjørnø et al., 2017) for control design and evaluation through numerical simulations. CSAD is a 1:90 scale replica of the full-scale Statoil Cat I Arctic Drillship, with a length of $L = 2.578$ m, shown in Fig 1. The inertia matrix is given as

$$\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_A, \tag{4}$$

where

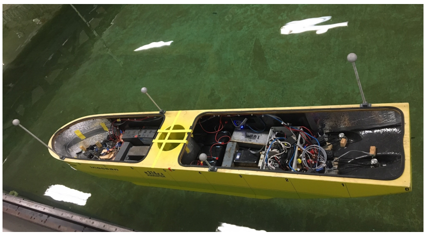


Fig. 1. CyberShip Inocean Cat I Arctic Drillship in the Marine Cybernetics Laboratory at NTNU.

$$\boldsymbol{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix} \tag{5}$$

$$\boldsymbol{M}_{A} = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix}. \tag{6}$$

The mass of CSAD is $m = 127.92$ kg, while $x_g = 0.00375$ m is the distance along the $x$-axis in the body frame from the center of gravity. The moment of inertia about the $z$-axis in the body frame is $I_z = 61.987\ \mathrm{kg\,m}^2$. Other parameter values are listed in Table 1. Note that $N_r$, which is marked in bold, has been changed to correspond better with to the actual physical behavior of CSAD. The Coriolis and centripetal matrix is

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{C}_{RB}(\boldsymbol{\nu}) + \boldsymbol{C}_{A}(\boldsymbol{\nu}), \tag{7}$$

with

$$\boldsymbol{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \tag{8}$$

$$\boldsymbol{C}_{A}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -c_{A,13}(\boldsymbol{\nu}) \\ 0 & 0 & c_{A,23}(\boldsymbol{\nu}) \\ c_{A,13}(\boldsymbol{\nu}) & -c_{A,23}(\boldsymbol{\nu}) & 0 \end{bmatrix}, \tag{9}$$

where

$$c_{A,13}(\boldsymbol{\nu}) = -Y_{\dot{v}} v - \frac{1}{2}(N_{\dot{v}} + Y_{\dot{r}})r \tag{10}$$

$$c_{A,23}(\boldsymbol{\nu}) = -X_{\dot{u}} u. \tag{11}$$

Finally, the damping matrix $\boldsymbol{D}(\boldsymbol{\nu})$ is given as

$$\boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{D}_L + \boldsymbol{D}_{NL}(\boldsymbol{\nu}), \tag{12}$$

where

$$\boldsymbol{D}_{L} = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & 0 \\ 0 & 0 & -N_r \end{bmatrix} \tag{13}$$

$$\boldsymbol{D}_{NL}(\boldsymbol{\nu}) = \begin{bmatrix} d_{NL,11}(\boldsymbol{\nu}) & 0 & 0 \\ 0 & d_{NL,22}(\boldsymbol{\nu}) & 0 \\ 0 & 0 & d_{NL,33}(\boldsymbol{\nu}) \end{bmatrix}, \tag{14}$$

and

$$d_{NL,11}(\boldsymbol{\nu}) = -X_{|u|u}|u| - X_{uuu}u^2 \tag{15}$$

$$d_{NL,22}(\boldsymbol{\nu}) = -Y_{|v|v}|v| - Y_{|r|v}|r| \tag{16}$$

$$d_{NL,33}(\boldsymbol{\nu}) = -N_{|v|r}|v| - N_{|r|r}|r|. \tag{17}$$

Table 1. Parameters for CSAD (Bjørnø et al., 2017)

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| $X_{\dot{u}}$ | $-3.262$ | $X_{uuu}$ | $-8.557$ |
| $Y_{\dot{v}}$ | $-28.89$ | $Y_v$ | $-4.673$ |
| $Y_{\dot{r}}$ | $-0.525$ | $Y_{\lvert v\rvert v}$ | 0.398 |
| $N_{\dot{v}}$ | $-0.157$ | $Y_{\lvert r\rvert v}$ | $-0.805$ |
| $N_{\dot{r}}$ | $-13.98$ | $N_{\lvert v\rvert r}$ | 0.080 |
| $X_u$ | $-2.332$ | $N_r$ | **-6.900** |
| $X_{\lvert u\rvert u}$ | 0 | $N_{\lvert r\rvert r}$ | $-0.0115$ |

The considered model describes a fully actuated ship, where the actuator forces and moments are modelled using the six mounted thrusters $\boldsymbol{u} = [u_1, u_2, u_3, u_4, u_5, u_6]^\top \in \mathbb{R}^6$ (Bjørnø et al., 2017). These are related to the input vector $\boldsymbol{\tau}$ through the actuator model

$$\boldsymbol{\tau}(\boldsymbol{u}) = \boldsymbol{T}\boldsymbol{K}_T\boldsymbol{u}, \tag{18}$$

where $\boldsymbol{T} \in \mathbb{R}^{3\times 6}$ is an actuator configuration matrix, while $\boldsymbol{K}_T \in \mathbb{R}^{6\times 6}$ is an actuator force matrix. The actuator configuration matrix is

$$\boldsymbol{T} = \begin{bmatrix} c(\delta_1) & c(\delta_2) & c(\delta_3) & c(\delta_4) & c(\delta_5) & c(\delta_6) \\ s(\delta_1) & s(\delta_2) & s(\delta_3) & s(\delta_4) & s(\delta_5) & s(\delta_6) \\ \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 \end{bmatrix}, \tag{19}$$

where $c(\delta_i) = \cos(\delta_i)$, $s(\delta_i) = \sin(\delta_i)$. The constant $\phi_i = L_i \cos(\beta_i)\sin(\delta_i)$ with $L_i = \sqrt{L_{x,i}^2 + L_{y,i}^2}$, where $L_{x,i}$ and $L_{y,i}$ represent the physical placements of the $i$th actuator and $\beta_i = \tan(L_{y,i}/L_{x,i})$ for $i \in [1, 6]$. The actuator force matrix is given as

$$\boldsymbol{K}_T = \mathrm{diag}([K_{T,1}, K_{T,2}, K_{T,3}, K_{T,4}, K_{T,5}, K_{T,6}]), \tag{20}$$

where $K_{T,i} > 0$ is the thrust force from the $i$th propeller. In (Bjørnø et al., 2017), the actuator magnitude constraints are stated as $u_i \in [-0.5, 0.5]$. In this work, we fix the actuators to the following angles $\boldsymbol{\delta} = [\pi, \pi/4, -\pi/4, 0, 5\pi/4, 3\pi/4]$.

The considered ship has to move at low speeds in order to be fully actuated for DP operations. Assuming low-speed maneuvers, the kinetic model in (2) can be simplified to

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{D}_L\boldsymbol{\nu} = \boldsymbol{\tau}, \tag{21}$$

since in low-speed maneuvers the linear damping will dominate over both the nonlinear damping and the Coriolis and centripetal forces(Fossen, 2011). The model (21) will be used in the control designs in the following sections.

## 3. ASSUMPTIONS AND CONTROL OBJECTIVE

It is assumed that both the pose vector $\boldsymbol{\eta}(t)$ and velocity vector $\boldsymbol{\nu}(t)$ can be measured, and that no disturbances and uncertainties affect the system.

The control objective is to make $\tilde{\boldsymbol{\eta}}(t) \triangleq \boldsymbol{\eta}(t) - \boldsymbol{\eta}_t(t) \rightarrow \boldsymbol{0}$ as $t \rightarrow \infty$, where $\boldsymbol{\eta}_t(t) \in \mathbb{R}^2 \times \mathbb{S}$ represents the pose associated with a virtual target ship. The motion of the target ship is typically defined by a human or generated by a guidance system.

For notational simplicity, the time $t$ is omitted in the rest of this paper.

## 4. BENCHMARK CONTROLLER

In (Sørensen and Breivik, 2016), a cascaded motion controller with nonlinear pose and velocity feedback is suggested, which is shown to be able to limit the control signal with respect to the control error. It should be noted that this controller is a step towards a final solution which can handle actuator magnitude constrains. The controller from (Sørensen and Breivik, 2016) is modified to a low-speed DP-version where the control input can be chosen as

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}} + \boldsymbol{D}_L\boldsymbol{\alpha} - \boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2. \tag{22}$$

The error variables $\boldsymbol{z}_1 = [z_{1,x}, z_{1,y}, z_{1,\psi}]^\top$ and $\boldsymbol{z}_2 = [z_{2,u}, z_{2,v}, z_{2,r}]^\top$ are defined as

$$\boldsymbol{z}_1 \triangleq \boldsymbol{R}^\top(\psi)(\boldsymbol{\eta} - \boldsymbol{\eta}_t) \tag{23}$$

$$\boldsymbol{z}_2 \triangleq \boldsymbol{\nu} - \boldsymbol{\alpha}, \tag{24}$$

and $\boldsymbol{\alpha} = [\alpha_u, \alpha_v, \alpha_r] \in \mathbb{R}^3$ is a vector of stabilising functions. This can be interpreted as a desired velocity

$$\boldsymbol{\alpha} = \boldsymbol{R}^\top\dot{\boldsymbol{\eta}}_t - \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1, \tag{25}$$

where

$$\boldsymbol{K}_1(\boldsymbol{z}_1) \triangleq \boldsymbol{\Gamma}_1\boldsymbol{\Omega}(\boldsymbol{z}_1), \tag{26}$$

and

$$\boldsymbol{\Omega}(\boldsymbol{z}_1) \triangleq \begin{bmatrix} \frac{1}{\sqrt{\boldsymbol{z}_{1,\tilde{p}}^\top\boldsymbol{z}_{1,\tilde{p}}+\Delta_{\tilde{p}}^2}}\mathbf{I}_{2\times2} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \frac{1}{\sqrt{z_{1,\tilde{\psi}}^2+\Delta_{\tilde{\psi}}^2}} \end{bmatrix} \tag{27}$$

represents a nonlinear control gain with $\boldsymbol{\Gamma}_1 > 0$, $\mathbf{z}_{1,\tilde{p}} \triangleq [z_{1,x}, z_{1,y}]^\top$, $\Delta_{\tilde{p}} > 0$ and $\Delta_{\tilde{\psi}} > 0$ and $\Delta_{\tilde{\psi}} > 0$. The nonlinear feedback term in (22) is given as

$$\boldsymbol{K}_2(\boldsymbol{z}_2) = \boldsymbol{\Gamma}_2 \begin{bmatrix} \frac{1}{\sqrt{\boldsymbol{z}_{2,\tilde{v}}^\top\boldsymbol{z}_{2,\tilde{v}}+\Delta_{\tilde{v}}^2}}\boldsymbol{I}_{2\times2} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \frac{1}{\sqrt{z_{2,\tilde{r}}^2+\Delta_{\tilde{r}}^2}} \end{bmatrix}, \tag{28}$$

with the control gain $\boldsymbol{\Gamma}_2 > 0$, where $\boldsymbol{z}_{2,\tilde{v}}$ is defined as $\boldsymbol{z}_{2,\tilde{v}} \triangleq [z_{2,u}, z_{2,v}]^\top$, $\Delta_{\tilde{v}} > 0$ and $\Delta_{\tilde{r}} > 0$. The time derivative of $\boldsymbol{\alpha}$ is:

$$\dot{\boldsymbol{\alpha}} = \boldsymbol{R}^\top\ddot{\boldsymbol{\eta}}_t + \boldsymbol{S}^\top\boldsymbol{R}^\top\dot{\boldsymbol{\eta}}_t - \dot{\boldsymbol{K}}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\dot{\boldsymbol{z}}_1, \tag{29}$$

where

$$\dot{\boldsymbol{K}}_1(\boldsymbol{z}_1) = -\boldsymbol{\Gamma}_1 \begin{bmatrix} \frac{\boldsymbol{z}_{1,\tilde{p}}^\top\dot{\boldsymbol{z}}_{1,\tilde{p}}\boldsymbol{I}_{2\times2}}{(\boldsymbol{z}_{1,\tilde{p}}^\top\boldsymbol{z}_{1,\tilde{p}}+\Delta_{\tilde{p}}^2)^{\frac{3}{2}}} & \mathbf{0}_{2\times1} \\ \mathbf{0}_{1\times2} & \frac{z_{1,\tilde{\psi}}\dot{z}_{1,\tilde{\psi}}}{(z_{1,\tilde{\psi}}^2+\Delta_{\tilde{\psi}}^2)^{\frac{3}{2}}} \end{bmatrix}, \tag{30}$$

with

$$\dot{\boldsymbol{z}}_1 = \boldsymbol{S}^\top\boldsymbol{z}_1 - \boldsymbol{K}_1(\boldsymbol{z}_1)\boldsymbol{z}_1 + \boldsymbol{z}_2, \tag{31}$$

where

$$\mathbf{S}(r) = \begin{bmatrix} 0 & -r & 0 \\ r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{32}$$

is a skew-symmetric matrix satisfying $\mathbf{z}_1^\top\mathbf{S}(r)^\top\mathbf{z}_1 = 0$.

## 5. DYNAMIC WINDOW-BASED CONTROL DESIGN

### 5.1 *Simplified Dynamic Window Algorithm*

Here, we present a 3 DOF extension to the 2 DOF DWC controller presented in (Sørensen et al., 2017).

Based on the ship model and its actuator magnitude constraints, a set of possible velocities can be found. This set contains all velocities the ship can achieve with respect to the actuator magnitude constraints. The possible velocities can be found by computing the steady-state solutions of the kinetics of (21) for all possible control inputs:

$$\boldsymbol{\tau}(\boldsymbol{u}) = \boldsymbol{D}_L\boldsymbol{\nu}_{ss}, \tag{33}$$

within the actuator magnitude constraints

$$u_i \in [-0.5, 0.5]. \tag{34}$$

The steady-state solutions of (33) for a uniformly distributed set of the control inputs is shown in Fig. 2. The

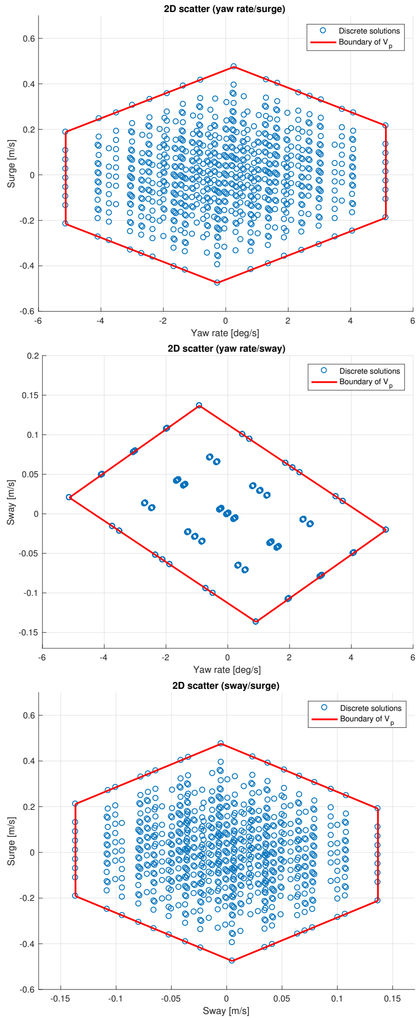


Fig. 2. Possible combinations of surge speed, sway speed and yaw rate, with respect to the actuator magnitude saturation limits.

set of possible velocities can be defined as:

$$V_p = \{(u, v, r) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R} \mid g(u, v, r) \geq 0\}, \tag{35}$$

where $g(u, v, r)$ is a positive semidefinite function for feasible velocities with respect to the actuator constraints. An approximation of the 3 DOF set is done by projecting the set into three 2 DOF sets to simplify calculations. We justify this approximation by noting that each of the steady-state solution boundary faces are almost parallel with one axis, see Fig. 2. Following this, faces that are parallel with one axis can be parameterized by the remaining two variables. Notice, however, that we lose information where all three variables are correlated, and can therefore not model faces which are not parallel with one of the axes. The result of the approximation is the following three sets of possible velocities:

$$V_{p,(u,r)} = \{(u, r) \in \mathbb{R} \times \mathbb{R} \mid g_{(u,r)}(u, r) \geq 0\} \tag{36}$$

$$V_{p,(v,r)} = \{(v, r) \in \mathbb{R} \times \mathbb{R} \mid g_{(v,r)}(v, r) \geq 0\} \tag{37}$$

$$V_{p,(u,v)} = \{(u, v) \in \mathbb{R} \times \mathbb{R} \mid g_{(u,v)}(u, v) \geq 0\}, \tag{38}$$

where $g_{(u,r)}(u, r)$, $g_{(v,r)}(v, r)$ and $g_{(u,v)}(u, v)$ are greater than or equal to zero for velocities inside the corresponding boundaries. Given $m$, $n$ and $k$ approximated boundaries, defined by the functions $h_{a,(u,r)}(u, r) = h_{b,(v,r)}(v, r) = h_{c,(u,v)}(u, v) = 0$, $a \in \{1, 2, ..., m\}$, $b \in \{1, 2, ..., n\}$ and $c \in \{1, 2, ..., k\}$, the approximated functions are given as:

$$g_{(u,r)}(u, r) = min(h_{1,(u,r)}(u, r), h_{2,(u,r)}(u, r), \\ ..., h_{m,(u,r)}(u, r)) \tag{39}$$

$$g_{(v,r)}(v, r) = min(h_{1,(v,r)}(v, r), h_{2,(v,r)}(v, r), \\ ..., h_{n,(v,r)}(v, r)) \tag{40}$$

$$g_{(u,v)}(u, v) = min(h_{1,(u,v)}(u, v), h_{2,(u,v)}(u, v), \\ , ..., h_{k,(u,v)}(u, v)). \tag{41}$$

Here, the functions $h_{a,(u,r)}(u, r) = h_{b,(v,r)}(v, r) = h_{c,(u,v)}(u, v) = 0$ are defined by using regression on the boundary of the sets $V_{p_{(u,r)}}$, $V_{p_{(v,r)}}$ and $V_{p_{(u,v)}}$, where $\nabla h_{a,(u,r)}(u, r)$, $\nabla h_{b,(v,r)}(v, r)$ and $\nabla h_{c,(u,v)}(u, v)$ are required to be pointing inwards to the valid solutions. Next, the space of reachable points within one time step $T$ needs to be defined. Using

$$\dot{\boldsymbol{\nu}}_{min} = [\dot{u}_{min}, \dot{v}_{min}, \dot{r}_{min}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{min}(\boldsymbol{u}) - \boldsymbol{D}_L \boldsymbol{\nu}^*) \tag{42}$$

$$\dot{\boldsymbol{\nu}}_{max} = [\dot{u}_{max}, \dot{v}_{max}, \dot{r}_{max}] = \boldsymbol{M}^{-1}(\boldsymbol{\tau}_{max}(\boldsymbol{u}) - \boldsymbol{D}_L \boldsymbol{\nu}^*), \tag{43}$$

we find the acceleration limits and the reachable velocities for the current time step, resulting in the dynamic velocity window

$$\begin{aligned} V_w = \{(u, v, r) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R} \mid \\ u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ \wedge v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T] \\ \wedge r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\}, \end{aligned} \tag{44}$$

which we project into the three cases

$$\begin{aligned} V_{w,(u,r)} = &\{(u, r) \in \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ &\wedge r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\} \end{aligned} \tag{45}$$

$$\begin{aligned} V_{w,(v,r)} = &\{(v, r) \in \mathbb{R} \times \mathbb{R} \mid v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T] \\ &\wedge r \in [r^* + \dot{r}_{min}T, r^* + \dot{r}_{max}T]\} \end{aligned} \tag{46}$$

$$\begin{aligned} V_{w,(u,v)} = &\{(u, v) \in \mathbb{R} \times \mathbb{R} \mid u \in [u^* + \dot{u}_{min}T, u^* + \dot{u}_{max}T] \\ &\wedge v \in [v^* + \dot{v}_{min}T, v^* + \dot{v}_{max}T]\}. \end{aligned} \tag{47}$$

This defines the sets of dynamically feasible velocities as

$$V_{f,(u,r)} \triangleq V_{p,(u,r)} \cap V_{w,(u,r)} \tag{48}$$

$$V_{f,(v,r)} \triangleq V_{p,(v,r)} \cap V_{w,(v,r)} \tag{49}$$

$$V_{f,(u,v)} \triangleq V_{p,(u,v)} \cap V_{w,(u,v)}. \tag{50}$$

Next, the sets of dynamically feasible velocities are discretised uniformly to obtain discrete sets of dynamically feasible velocities. For the 3 DOF case, the desired velocity is defined as

$$\boldsymbol{\nu}_d \triangleq [u_d, v_d, r_d]^\top. \tag{51}$$

Given $\boldsymbol{\nu}_d$, the optimal dynamically feasible velocity $\boldsymbol{\nu}_f = [u_f, v_f, r_f]^\top$ can be selected as

$$\boldsymbol{\nu}_f = \underset{(u,v,r) \in V_f}{\text{argmax}}\ G(\boldsymbol{\nu}, \boldsymbol{\nu}_d), \tag{52}$$

where $V_f$ is the general 3 DOF solution and $G(\boldsymbol{\nu}, \boldsymbol{\nu}_d)$ is an objective function, which is defined as

$$\begin{aligned} G(\boldsymbol{\nu}, \boldsymbol{\nu}_d) \overset{\triangle}{=} &\operatorname{surge}(u, u_d) + \operatorname{sway}(v, v_d) \\ &+ \operatorname{yawrate}(r, r_d), \end{aligned} \tag{53}$$

with

$$\operatorname{surge}(u, u_d) = 1 - \frac{|u_d - u|}{\max\limits_{u' \in V_f}(|u_d - u'|)} \in [0, 1] \tag{54}$$

$$\operatorname{sway}(v, v_d) = 1 - \frac{|v_d - u|}{\max\limits_{v' \in V_f}(|v_d - v'|)} \in [0, 1] \tag{55}$$

$$\operatorname{yawrate}(r, r_d) = 1 - \frac{|r_d - r|}{\max\limits_{r' \in V_f}(|r_d - r'|)} \in [0, 1]. \tag{56}$$

Notice that by using this objective function, we minimise the scaled 1-norm of the entire discrete set of dynamically feasible velocity pairs. For the three 2 DOF case, this algorithm is modified to fit 2 DOF and run once for each velocity pair scenario; surge speed and yaw rate, sway speed and yaw rate, and surge and sway speed. Resulting in the sets of dynamically feasible velocities

$$\boldsymbol{\nu}_{f,(u,r)} = [\nu_{f,u}, 0, \nu_{f,r}]^\top \tag{57}$$

$$\boldsymbol{\nu}_{f,(v,r)} = [0, \nu_{f,v}, \nu_{f,r}]^\top \tag{58}$$

$$\boldsymbol{\nu}_{f,(u,v)} = [\nu_{f,u}, \nu_{f,v}, 0]^\top, \tag{59}$$

which combines into

$$\boldsymbol{\nu}_f = \frac{\boldsymbol{\nu}_{f,(u,r)} + \boldsymbol{\nu}_{f,(v,r)} + \boldsymbol{\nu}_{f,(u,v)}}{2} \tag{60}$$

for the full 3 DOF case. Fig. 3 illustrates $V_p$, $V_w$, $V_f$ and $\boldsymbol{\nu}_d = [0.15 \text{ m/s}, -0.07 \text{ m/s}, -1.4324 \text{ deg/s}]$ given a current velocity $\boldsymbol{\nu} = [0.2 \text{ m/s}, -0.05 \text{ m/s} -1.1459 \text{ deg/s}]$.

### *5.2 Dynamic Window-based Controller*

We now combine the traditional control design with the simplified DW algorithm in order to develop a dynamic window-based controller (DWC). In this setup, the simplified DW algorithm will use $\boldsymbol{\alpha} = [\alpha_u, \alpha_v, \alpha_r]^\top$ as an input such that $\boldsymbol{\nu}_d = \boldsymbol{\alpha}$. In the case where $\boldsymbol{\alpha}$ is an infeasible velocity, the simplified DW algorithm will modify $\boldsymbol{\alpha}$ to a feasible velocity $\boldsymbol{\alpha}_f = [\alpha_{f,u}, \alpha_{f,v}, \alpha_{f,r}]^\top$, otherwise $\boldsymbol{\alpha}_f = \boldsymbol{\alpha}$. We want the ship to reach $\boldsymbol{\alpha}_f$ after the time $T$, hence the desired acceleration is chosen to be

$$\dot{\boldsymbol{\alpha}}_{DWC} = \frac{\boldsymbol{\alpha}_f - \boldsymbol{\nu}}{T}, \tag{61}$$

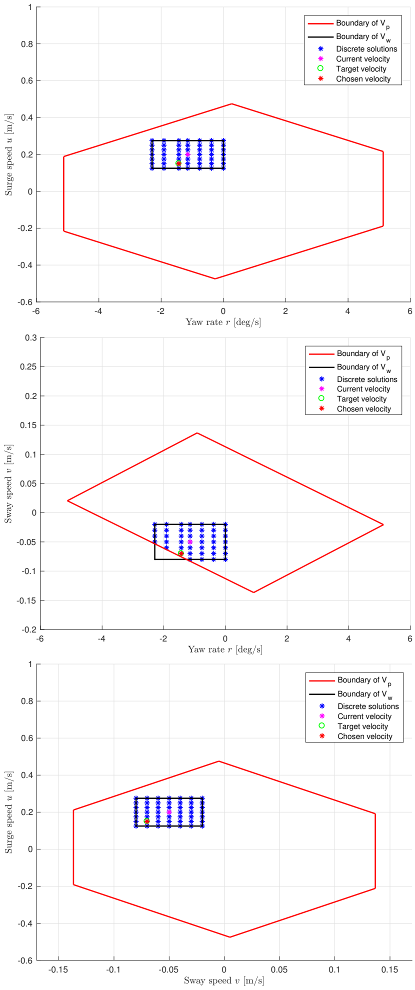


Fig. 3. The dynamically feasible velocity set, surrounded by the boundaries of the dynamic velocity window and set of possible velocities.

and

$$\boldsymbol{\alpha}_{DWC} = \int_0^t \dot{\boldsymbol{\alpha}}_{DWC} d\sigma + \boldsymbol{\alpha}_{DWC}(0). \tag{62}$$

Both $\boldsymbol{\alpha}_{DWC}$ and $\dot{\boldsymbol{\alpha}}_{DWC}$ are used in the kinetic controller (22) which is modified to

$$\boldsymbol{\tau} = \boldsymbol{M}\dot{\boldsymbol{\alpha}}_{DWC} + \boldsymbol{D}_L\boldsymbol{\alpha}_{DWC}. \tag{63}$$

When comparing the control law (63) against (22), it can be seen that the feedback term $-\boldsymbol{K}_2(\boldsymbol{z}_2)\boldsymbol{z}_2$ in (22) is not included in (63) since the DWC makes the optimal velocity pair track the target velocity by using (61)-(62). The DWC is a feedforward-based control algorithm which has some weaknesses against uncertainties and disturbances since the DWC is heavily model-dependent. Introducing a feedback term and adaptation can help to overcome these weaknesses.

## 6. SIMULATION RESULTS

In this section, we present numerical simulation results of the two motion controllers using the nonlinear ship model and actuator constraints of CSAD presented in Section 2. In addition, performance metrics are used to evaluate the controller behavior.

The control target is defined through a 4-corner test (Skjetne et al., 2017). The 4-corner test first tests the surge, sway and yaw motion individually and then increase the complexity of the task until the ship needs to do a combined surge, sway and yaw motion. In this test we use set-point tracking. This will affect (25) and (29) since $\dot{\boldsymbol{\eta}}_t = \boldsymbol{0}$ and $\ddot{\boldsymbol{\eta}}_t = \boldsymbol{0}$. The initial ship states are chosen to be $\boldsymbol{\eta}(0) = [5, 1, 0]$ and $\boldsymbol{\nu}(0) = \boldsymbol{0}$. The control gains are listed in Table 2, which are chosen such that the benchmark controller (BC) does not exceed the magnitude saturation constraints and follow the tuning rules suggested in (Sørensen et al., 2018).

Table 2. Control gains

| | **BC** | **DWC** |
|---|---|---|
| $\boldsymbol{\Gamma}_1$ | diag$([0.03, 0.03, 0.0349])$ | $-\|\|-$ |
| $\boldsymbol{\Gamma}_2$ | diag$([0.2, 0.12, 0.1745])\boldsymbol{M}$ | $N/A$ |
| $\Delta_{\tilde{p},\tilde{\psi}}$ | $[0.5, 0.5]$ | $-\|\|-$ |
| $\Delta_{\tilde{v},\tilde{r}}$ | $[0.7, 1]$ | $N/A$ |

### 6.1 Performance Metrics

To evaluate and compare the performance of the two controllers, performance metrics are used. We define

$$e(t) \triangleq \sqrt{\bar{\boldsymbol{\eta}}(t)^\top \bar{\boldsymbol{\eta}}(t)}, \tag{64}$$

as the error inputs for the performance metrics, with $\bar{\boldsymbol{\eta}}$ being the normalized signal of $\tilde{\boldsymbol{\eta}} = [\tilde{x}, \tilde{y}, \tilde{\psi}]^\top \triangleq \boldsymbol{\eta} - \boldsymbol{\eta}_t$, where $\tilde{x}$, $\tilde{y}$ and $\tilde{\psi}$ are in the intervals $[-0.5, 0.5]$ in the expected operational space of the ship (Eriksen and Breivik, 2017). These signals represent the instantaneous control errors, while we would like to consider the accumulated errors over time. Therefore, we use the performance metric integral of the absolute error (IAE)

$$IAE(e, t) \triangleq \int_0^t |e(\sigma)| d\sigma, \tag{65}$$

which integrates the temporal evolution of the absolute error. We also consider the integral of the absolute error multiplied by the energy consumption (IAEW) (Sørensen and Breivik, 2015) as

$$IAEW(e, t) \triangleq \int_0^t |e(\sigma)| d\sigma \int_0^t P(\sigma) d\sigma, \tag{66}$$

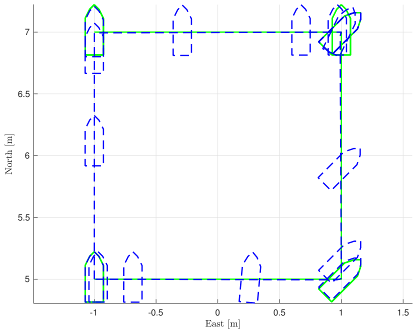
where

$$P(t) = |\boldsymbol{\nu}(t)^\top \boldsymbol{\tau}(t)| \tag{67}$$

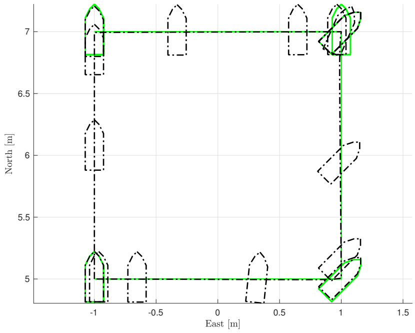
represents the mechanical power. IAEW thus indicates which controller has the best combined control accuracy and energy efficiency in one single metric.

### *6.2 Simulation Results*

In Fig. 4, the outline of the ship pose is plotted to show the transient convergence behavior.



Using DWC.



Using BC.

Fig. 4. The 4-corner test, where the dashed blue outline represents the DWC-controlled ship, the dash-dotted black outline represents the BC-controlled ship, while the green outline represents the 4-corner box.

Fig. 5 shows the pose of the ship together with the target pose. It can be seen that both control laws are able to track the target pose even though the DWC does not have a traditional velocity feedback term. Additionally, it can be seen that there is a small difference in how fast the controllers are able to track the target pose, where DWC is the fastest.

Fig. 6 shows that the DWC commands the control inputs to stay just below the maximum magnitude constraints of the actuators, while BC is tuned such that it does not exceed the magnitude constraints. The DWC keeps the control inputs high longer than the BC, since the DWC tracks the optimal velocity $\boldsymbol{\alpha}_f$ which is on the boundaries of the window unless the target velocity $\boldsymbol{\alpha}$ is inside the velocity window, while the control inputs from BC have a more conservative behavior.

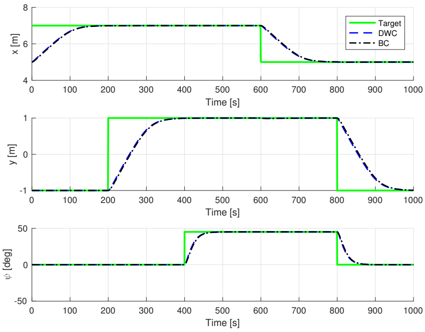


Fig. 5. Tracking the target pose.

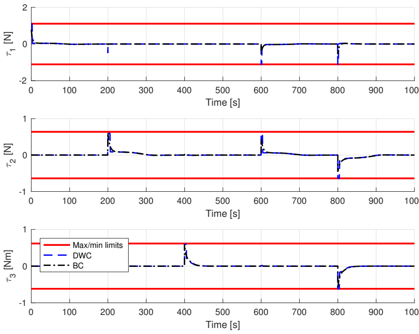


Fig. 6. The commanded control inputs with magnitude limits.

Fig. 7 illustrates how the surge speed, sway speed and yaw rate moves in the velocity space in order to track the target through the 4-corner test. The velocities of the ship are small in magnitude while performing the 4-corner test. This satisfies the requirements for assuming a linear ship model.

In Fig. 8, the performance metrics IAE and IAEW are shown. In particular, the IAE trajectory in the left of Fig. 8 confirms that the DWC has a faster transient response since it converges faster to a stationary value. The IAEW trajectory in the right of Fig. 8 shows that the DWC uses slightly more energy to fulfil the control objective. DWC have a better overall performance, since it has a faster transient response, but it comes at a cost in energy consumption as shown in IAEW.

## 7. CONCLUSION

This paper has proposed an extension of a simplified dynamic window algorithm from 2 DOF to 3 DOF, as a way

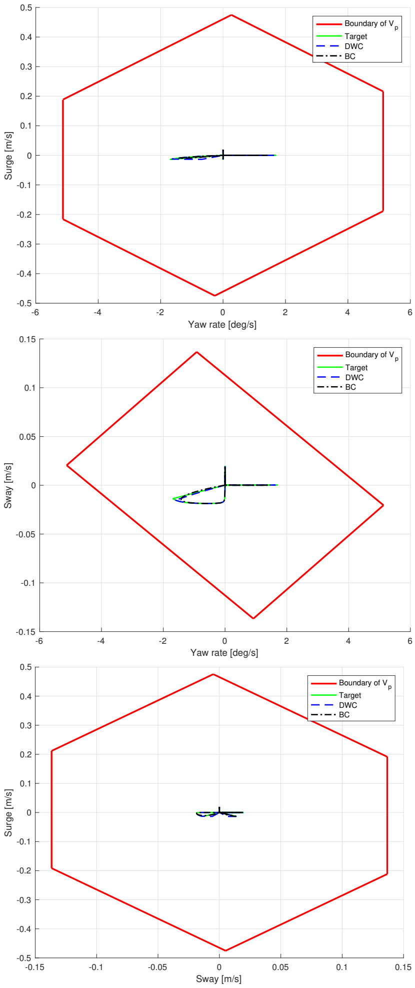


Fig. 7. Velocity trajectories in the set of possible velocities $V_p$.

to ensure that the actuator constraints of a fully actuated ship are satisfied. This algorithm has been used as a dynamic window-based controller (DWC) to guarantee that ship velocities remain within a feasible set. The DWC was evaluated against a motion controller using a traditional design approach. The methods were compared through numerical simulations, using two performance metrics to compare the behavior of the controllers. The simulation results showed that the proposed DWC controller has good tracking performance and that it is able to handle actuator magnitude constraints.

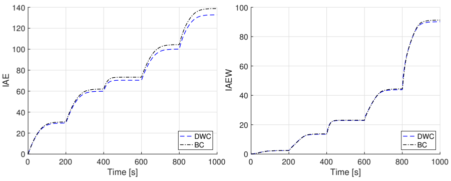


Fig. 8. IAE and IAEW performance metrics.

Future work will include introducing model uncertainties and unknown disturbances and making a stability analysis of the proposed control algorithm. It is also relevant to consider actuator rate constraints in addition to magnitude constraints. Finally, it is desirable to experimentally verify the results by implementing the methods on a model-scale test vessel in a controlled environment.

## ACKNOWLEDGEMENTS

This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, project number 223254.

## REFERENCES

Bjørnø, J., Heyn, H.M., Skjetne, R., Dahl, A.R., and Frederich, P. (2017). Modeling, parameter identification and thruster-assisted position mooring of C/S Inocean CAT I Drillship. *in Proceedings of the 36th International Conference on Ocean, Offshore and Arctic Engineering, Trondheim, Norway.*

Eriksen, B.-O. H., Breivik, M., Pettersen, K.Y., and Wiig, M.S. (2016). A modified dynamic window algorithm for horizontal collision avoidance for AUVs. *in Proceedings of the IEEE Multi-Conference on Systems and Control, Buenos Aires, Argentina.*

Eriksen, B.-O. H. and Breivik, M. (2017). *Modeling, Identification and Control of High-Speed ASVs: Theory and Experiments*, 407–431. Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles, Springer International Publishing.

Fossen, T.I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control.* Wiley.

Fox, D., Burgard, W., and Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1), 23–33.

Skjetne, R., Sørensen, M.E.N., Breivik, M., Værnø, S.A.T., Brodtkorb, A.H., Sørensen, A.J., Kjerstad, Ø.K., Calabrò, V., and Vinje, B.O. (2017). AMOS DP research cruise 2016: Academic full-scale testing of experimental dynamic positioning control algorithms onboard R/V Gunnerus. *in Proceedings of the 36th International Conference on Ocean, Offshore and Arctic Engineering, Trondheim, Norway.*

Sørensen, M.E.N. and Breivik, M. (2015). Comparing nonlinear adaptive motion controllers for marine surface vessels. *in Proceedings of the 10th IFAC Conference on Manoeuvring and Control of Marine Craft, Copenhagen, Denmark.*

Sørensen, M.E.N. and Breivik, M. (2016). Comparing combinations of linear and nonlinear feedback terms for motion control of marine surface vessels. *in Proceedings of the 10th IFAC Conference on Control Applications in Marine Systems, Trondheim, Norway.*

Sørensen, M.E.N., Eriksen, B.-O. H., and Breivik, M. (2017). A ship heading and speed control concept inherently satisfying actuator constraints. *in Proceedings of the 1st IEEE Conference on Control Technology and Applications, Hawai'i, USA.*

Sørensen, M.E.N., Skjetne, R., and Breivik, M. (2018). Comparing combinations of linear and nonlinear feedback terms for motion control of marine surface vessels. *submitted to IEEE Transactions on Control Systems Technology.*