Master's thesis

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

Caio Swan de Freitas

Control-Oriented Modelling for the Conversion of Surface Vessels to Unmanned

Master's thesis in Marine Technology

Supervisor: Roger Skjetne, Arild Hepsø and Stephanie Kemna

June 2020

NTNU
Norwegian University of Science and Technology

MARITIME ROBOTICS

Caio Swan de Freitas

Control-Oriented Modelling for the Conversion of Surface Vessels to Unmanned

Master's thesis in Marine Technology
Supervisor: Roger Skjetne, Arild Hepsø and Stephanie Kemna
June 2020

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

NTNU
Norwegian University of
Science and Technology

Abstract

Control-oriented modelling is a method capable of simplifying the identification and tuning of a mathematical model for a vessel. It can represent the dynamics of vessels that operate at a varied range of speeds and was adapted to unmannned surface vessels (USVs) by Breivik et al. (2008). Improvements to control-oriented modelling for USVs have been proposed by Eriksen and Breivik (2017) expanding the model structure to include yaw instead of only surge and by Kvalvaag (2018) partially automating the identification procedure. It has been successfully applied to high-speed USVs and was identified in the project prior to this thesis (de Freitas, 2019) as a good solution to be used during conversions from surface vessels (SVs) to USVs.

This thesis continues the improvements in control-oriented modelling in order to deliver conversions with minimized cost, time spent and required know-how. This is done by attempting a full automation of the identification procedure. After data collection in trials, the procedure is already almost completely automated. The exception is the stage known as data extraction, in which the data collected needs to be transformed into inertia and damping results to be fitted to the model. Therefore, two new methods are proposed for data extraction: one performing steady-state system identification to distinguish steady-state and transient regions based on Dalheim and Steen (2020), RW-SSID, and another simultaneously identifying inertia and damping data using optimization and skipping steady-state system identification, SIMID. Additionally, these methods are combined with methods for automatic removal of data considered unreliable. The criteria used to define unreliable data are low signal-to-noise ratio and not reaching steady-state.

The proposed improvements were tested using data acquired by Eriksen and Breivik (2017) in cooperation with Maritime Robotics. The same case-study USV was used by Breivik et al. (2008) and Kvalvaag (2018). Results show that the proposed data extraction method SIMID succeeded in fully automating the control-oriented modelling. On the other hand, RW-SSID had its implementation halted after its results required an excessive number of tuning parameters. In addition, the results from the automated control-oriented modelling were successfully used for automating the tuning of saturation limits for reference filters.

Sammendrag

Kontrollorientert modellering er en metode som er i stand til å forenkle identifiseringen og tuningen av en matematisk modell for et fartøy. Metoden kan representere dynamikken til fartøy som opererer i et bredt utvalg av hastigheter, og ble tilpasset til ubemannede overflatefartøy (USV-er) av Breivik et al. (2008). Forbedringer av kontrollorientert modellering for USV-er har blitt foreslått av Eriksen and Breivik (2017), som utvidet modellstrukturen til å inkludere giring i stedet for bare jaging, og av Kvalvaag (2018), som delvis automatiserte identifikasjonsprosedyren. Metoden har blitt anvendt med suksess på hurtiggående USV-er og ble i prosjektet som var forløperen til denne oppgaven (de Freitas, 2019) identifisert som en god løsning for konvertering av overflatefartøy (SV-er) til USV-er.

Denne masteroppgaven fortsetter forbedringene i kontrollorientert modellering for å kunne utføre konverteringer med minimal kostnad, tidsbruk og nødvendig kunnskap. Dette gjøres ved å prøve å oppnå full automatisering av identifikasjonsprosedyren. Etter at datainnsamling i forsøk er gjennomført, er prosedyren allerede nesten helt automatisert. Unntaket er stadiet kjent som dataekstraksjon, der innsamlet data må transformeres til treghet- og dempningsresultater som kan tilpasses modellen. Derfor foreslås to nye metoder for dataekstraksjon: en som utfører systemidentifikasjon av stabil tilstand ved bruk av et rullende vindu for å skille stabile og transiente regioner basert på Dalheim and Steen (2020), kalt RW-SSID. Den andre identifiserer treghet- og dempningsdata samtidig ved bruk av optimalisering og unngår identifikasjon av stabil tilstand, kalt SIMID. I tillegg er disse metodene kombinert med metoder for automatisk fjerning av data som er ansett som upålitelige. Kriteriene som brukes for å definere upålitelige data er lavt signal-til-støy-forhold og ikke klarer å oppnå stabil tilstand.

De foreslåtte forbedringene ble testet ved hjelp av data samlet inn av Eriksen and Breivik (2017) i samarbeid med Maritime Robotics. Den samme casestudien USV ble brukt av Breivik et al. (2008) og Kvalvaag (2018). Resultatene viser at den foreslåtte dataekstraksjonsmetoden SIMID lyktes med å automatisere den kontrollorienterte modelleringen. Derimot ble implementeringen av RW-SSID stoppet fordi resultatene krevde et for stort antall tuningparametere. Resultatene fra den automatiserte kontrollorienterte modelleringen ble også brukt med suksess til å automatisere tuningen av metningsgrenser for referansefiltre.

Preface

This thesis was written as the final part of my master's education at NTNU in the MSc Marine Technology program with specialization in Marine Cybernetics. It explores the world of unmanned surface vessels and the motivation for the topic came from working in cooperation with Maritime Robotics. Conversions from surface vessels to unmanned are performed by the company and the challenge was to improve existing methods.

The work started with a project thesis (de Freitas, 2019) to identify control system solutions with potential to improve such conversions and control-oriented modelling was picked among other solutions to be further explored in this thesis. After reviewing the reference literature, I noticed how the procedure of control-oriented modelling still required some tuning and making it fully automated would bring clear benefits. This was successfully achieved with independent development of new methods. Restrictions due to COVID-19 impeded the execution of planned experimental tests, but this was solved without loss of quality by using data collected by Eriksen and Breivik (2017) also in cooperation with Maritime Robotics.

Caio Swan de Freitas

Trondheim, June 8th, 2020

Acknowledgment

I would like to thank my supervisors, Professor Roger Skjetne, Arild Hepsø and Stephanie Kemna, for supporting me over a year on project and thesis with much appreciated help in planning and feedback on my work.

I would also like to thank all employees at Maritime Robotics for the friendly and helpful workspace, making presence even when a forced home-office was necessary. Especially, I wish to acknowledge the help provided by Giorgio D. K. M. Kufoalor, who became an unofficial additional supervisor with invaluable advices.

Thanks to Bjørn-Olav H. Eriksen for providing me original data from his article that was vital after it was no longer possible to run experiments and became a good benchmark for results.

Finally, I would like to thank my fiancée Amanda Rossi, who used her completely opposite approach to academical writing to give me excellent tips to improve my text.

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

MASTER OF TECHNOLOGY THESIS DEFINITION (30 SP)

Name of the candidate:	Caio Swan de Freitas
Field of study:	Marine control engineering
Thesis title (Norwegian):	Kontrollorientert modellering for konvertering av overflatefartøy til ubemannede fartøy
Thesis title (English):	Control-oriented modelling for the conversion of surface vessels to unmanned vessel

Background

In recent years, unmanned surface vehicles (USVs) have increased in popularity as an alternative to reduce costs and operational risks for the offshore and ocean-related research industries, among others. This trend has led to the development of USV-specific designs. Another option that has the potential to reduce cost, is to convert existing (manned) surface vessels (SVs) to USVs for cases in which SVs are already applied. A key part of a USV is the software that makes it unmanned; conversions pose a challenge that this project intends to mitigate in terms of time, cost, and know-how to adapt and calibrate such software properly.

Considering underactuated USVs for target-tracking at high speeds, Breivik et al. (2008) proposed a novel velocity controller with the objective to deliver a simple yet advanced control system requiring minimal system identification and tuning trials. Eriksen and Breivik (2018) proposed improvements on this controller, while Kvalvaag (2018) proposed automatic solutions to the identification and tuning procedures. This methodology matches the identified necessities for conversions since it can cover the vessel's entire operational range and does not compromise the time, cost, or experience required for the conversion.

In this Master project, a control-oriented modelling and control law development will be investigated for some proposed future work topics from the cited articles, also considering necessary improvements. The work shall be conducted in cooperation with Maritime Robotics AS.

Work description

1. Perform a background and literature review to provide information and relevant references on:
   - Controller based on control-oriented modelling.
   - System identification, including automated model calibration.
   - Relevant existing tuning procedures.

   Write a list with abbreviations and definitions of terms and symbols, relevant to the literature study and project report.
2. Implement the methodology from previous works in the Marine Systems Simulator (Fossen and Perez, 2004) and test for target-tracking scenarios.
3. Investigate performance and necessary improvements when applying methodology to path following scenarios, as suggested in Eriksen and Breivik (2018).
4. Analyze impacts of identified model on reference filter for desired velocity. For instance, defining the acceleration limits of the reference filter as a function of the model, as suggested in Eriksen and Breivik (2017).
5. Investigate and propose improvements to automation, as suggested in Kvalvaag (2018).
6. Apply the resulting methodology on an experimental test with an USV, and analyze its impacts on a practical conversion.

Specifications

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

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

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

The report shall be organized in a logical structure to give a clear exposition of background, problem, design, results, and critical assessments. The text should be brief and to the point, with a clear language. Rigorous mathematical deductions and illustrating figures are preferred over lengthy textual descriptions. The report shall have font size 11 pts., and it is not expected to be longer than 70 A4-pages, 100 B5-pages, from introduction to conclusion, unless otherwise agreed upon. It shall be written in English (preferably US) and contain the elements: Title page, abstract, acknowledgement, project definition, list of symbols and acronyms, table of contents, introduction (project motivation, objectives, scope and delimitations), background/literature review, problem formulation, method, results, conclusions with recommendations for further work, references, and optional appendices. Figures, tables, and equations shall be numerated. The original contribution of the candidate and material taken from other sources shall be clearly identified. Work from other sources shall be properly acknowledged using quotations and a Harvard citation style (e.g. natbib Latex package). The work is expected to be conducted in an honest and ethical manner, without any sort of plagiarism and misconduct, which is taken very seriously by the university and will result in consequences. NTNU can use the results freely in research and teaching by proper referencing, unless otherwise agreed upon.

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

Start date: 13 January, 2020 **Due date:** As specified by the administration.

Supervisor: Roger Skjetne
Co-advisor(s): Arild Hepsø and Stephanie Kemna - both in Maritime Robotics AS.

Trondheim, 14 April, 2020

Digitally signed by Roger Skjetne
Date: 2020.04.14 18:57:22 +02'00'

Roger Skjetne
Supervisor

Table of Contents

Nomenclature

Acronyms

CG	Centre of Gravity
COG	Course Over Ground
DOF	Degree Of Freedom
IMO	International Maritime Organization
NED	North-East-Down
PID	Proportional Integral Derivative
RPM	Rotations Per Minute
RW-SSID	Rolling Window Steady-State IDentification
SI	System Identification
SIMID	SIMultaneous IDentification
SOG	Speed Over Ground
SSID	Steady-State IDentification
SV	Surface Vessel
USV	Unmanned Surface Vessel

Letters

α	Significance level for Student's two-tailed t-test
$\boldsymbol{\beta}_{\sigma_r}$	Vector of identified parameters for yaw damping component
$\boldsymbol{\beta}_{\sigma_U}$	Vector of identified parameters for surge damping component
$\boldsymbol{\beta}_{m_r}$	Vector of identified parameters for yaw inertia component
$\boldsymbol{\beta}_{m_U}$	Vector of identified parameters for surge inertia component
$\boldsymbol{\Delta\nu}$	$= \nu - \nu_0$, variation of velocity from the reference of the expansion $\boldsymbol{\nu}_0$

$\Delta\tau_i$	Size of i-th step command
δ	Rudder angle
$\delta_{f,i}$	SSID estimate of noise variance
ε_{LS}	Weighted least-squares fitting evaluation
ζ	Damping ratio
$\boldsymbol{\eta}$	$= [x, y, \psi]^\top$, position of the vessel regarding the inertial axis
λ	Regularization hyperparameter
λ_{number}	SSID filtering parameters
$\boldsymbol{\nu}$	$= [u, v, r]^\top$, velocity of the vessel in the body fixed axis orientation
$\boldsymbol{\nu}_0$	$= [u_0, 0, 0]$, constant velocity around which is the reference for the Taylor series expansion used to derive the Abkowitz model.
$\nu_{f,i}$	SSID estimate of noise variance
$\boldsymbol{\sigma}(\boldsymbol{x})$	Damping vector as a function of states
$\sigma_r(\boldsymbol{x})$	Damping yaw component as a function of states
$\sigma_U(\boldsymbol{x})$	Damping surge component as a function of states
σ_{b_1}	Standard deviation of linear function slope
σ_w	Standard deviation of white noise
ς	Scaled throttle
$\boldsymbol{\tau}$	Vector of commands
$\boldsymbol{\tau}_\delta$	Rudder angle command
$\boldsymbol{\tau}_m$	Motor throttle command
$\boldsymbol{\phi}_\sigma$	Vector of states for damping component
$\boldsymbol{\phi}_m$	Vector of states for inertia component
χ	COG
ψ	Vessel heading angle
ω_n	Natural frequency
a_r	Hyperparameter of inertia yaw component as a function of states
a_U	Hyperparameter of inertia surge component as a function of states
b_0	Linear function intercept
b_1	Linear function slope
b_r	Hyperparameter of inertia yaw component as a function of states

b_U	Hyperparameter of inertia surge component as a function of states
$\boldsymbol{C}(\boldsymbol{\nu})$	Coriolis-centripetal matrix as a function of vessel velocities
$\boldsymbol{D}(\boldsymbol{\nu})$	Damping matrix as a function of vessel velocities
$\mathcal{D}_{\boldsymbol{\sigma}}$	Damping data set for parameter identification
$\mathcal{D}_{m_r}$	Yaw inertia data set for parameter identification
$\mathcal{D}_{m_U}$	Surge inertia data set for parameter identification
Fn	Froude number
g	Gravity acceleration
H	Absolute height between initial and final estimated state
H_0, H_1	Statistical hypotheses for SSID
I_z	Inertia around yaw axis
K	Nomoto first-order and second-order model K parameter
k_i	Simplified damping parameter of x_i
L	Vessel length
$\boldsymbol{M}(\boldsymbol{x})$	Inertia matrix as a function of states
m	Vessel mass
m_i	Inertia parameter of x_i
$m_r(\boldsymbol{x})$	Inertia yaw component as a function of states
$m_U(\boldsymbol{x})$	Inertia surge component as a function of states
$N(\boldsymbol{\Delta\nu}, \delta)$	Yaw moment on Abkowitz model as function of $\boldsymbol{\Delta\nu}$ and δ
n_t	Number of timesteps in rolling window
$\boldsymbol{R}(\psi)$	Rotation matrix as a function of the heading ψ
R	Test statistic for SSID
r	Yaw rate
$R(\boldsymbol{\beta})$	Regularization function
$R(\mathrm{s})$	Reference filter transfer function
R_{lb}, R_{ub}	Lower and upper bounds for statistical test hypotheses
s	Transfer function variable
t_1	Variable for Student's t-test
$\mathrm{t}_{\alpha/2,n_t-2}$	Tabulated Student's t value
T	Nomoto first-order model T parameter

t	Time
T_1	Nomoto second-order model T_1 parameter
T_2	Nomoto second-order model T_2 parameter
T_3	Nomoto second-order model T_3 parameter
T_n	Natural period
t_s	Settling time for estimated state
U	$= \sqrt{u^2 + v^2}$, total linear speed
u	Surge speed
u_0	Constant surge speed on ν_0
v	Sway speed
$w(t)$	White noise with zero mean
w_i	Weight of results from i-th step on least-squares fitting
$\boldsymbol{x}$	Vector of current states
$\boldsymbol{x}_d$	Vector of desired states
$\boldsymbol{x}_{init}$	Vector of initial states
$\dot{\boldsymbol{x}}_{\max}$	Vector of maximum accelerations for reference filters
$X(\boldsymbol{\Delta\nu}, \delta)$	Surge force on Abkowitz model as function of $\boldsymbol{\Delta\nu}$ and δ
x_g	Surge distance from body fixed axis origin to centre of gravity
x_i	Main state during i-th step command
$x_{lim,low}$	Lower limit to check for noisy data
$x_{lim,up}$	Upper limit to check for noisy data
$Y(\boldsymbol{\Delta\nu}, \delta)$	Sway force on Abkowitz model as function of $\boldsymbol{\Delta\nu}$ and δ
Y_{vrr}	Hydrodynamic derivative from the Abkowitz model

Symbols

$\bar{(\cdot)}$	Average
$\hat{(\cdot)}$	Estimate
$(\cdot)^+$	State value at final timestep
$(\cdot)^-$	State value at first timestep
$\ddot{(\cdot)}$	Double derivative
$\dot{(\cdot)}$	Derivative
$\tilde{(\cdot)}$	Error between reference and desired value

Chapter 1

Introduction

1.1 Background and Motivation

The history behind seeking autonomy for vessels is not recent. In order to comprehend this statement, it is necessary to understand what autonomy means for ships. There is still an ongoing debate about final definitions, but the latest report on the matter by the International Maritime Organization (IMO) divides the autonomy of ships into four non-exclusive degrees (Maritime Safety Commitee, 2018):

- Degree 1: Ship with automated processes and decision support;
- Degree 2: Remotely controlled ship with seafarers on board;
- Degree 3: Remotely controlled ship without seafarers on board;
- Degree 4: Fully autonomous ship.

Initial attempts for the first degree of autonomy can be traced back to the beginning of the twentieth century. Back then, the first automation goal was to have an autopilot to steer the vessels. Daniels and Tucker (1952) describe how the Braine gear was invented in 1906 to automatically steer model yachts based on balancing the pressure on the mainsail and the pressure on the rudder. Another invention initially applied to model yachts in 1919 was the wind-vane steering system used to keep the boat on constant apparent wind (Førthmann, 1998). The same challenge led Minorsky (1922) to describe the development of a three-term controller that became known as the PID (proportional integral derivative) controller, which is nowadays the most widely used controller in most types of systems. In this last case, the application was targeted at commercial and navy ships.

Advancements in sensor, communication and controller technology over the following years made the second automation degree possible, but the transition to the third and fourth is still under way. In recent years, unmanned surface vessels (USVs) have increased in popularity as an alternative to manned vessels and other systems to reduce costs and operational risks for various industries. For instance, USVs are supporting scientific research about the oceans, offshore oil & gas exploration and military training (Liu et al., 2016). This trend has led to the development of many USV-specific vessel designs as it can be seen in a non-exhaustive list of 50 USV projects until 2011 in Liu et al. (2016), in which the majority are USV-specific designs. Another option is to convert existing (manned) surface vessels (SVs) to USVs. One such example is the Telemetron vessel shown in Figure 1.1. Such conversions have the potential to reduce costs even more for cases in which SVs are already successfully applied because there is no need for investments on a new design development. Moreover, Bertram (2008) states that a conversion augments the capabilities of a vessel by being able to operate both manned and unmanned in a proven platform. This represents a more affordable and lower risk solution than a new USV-specific vessel design.

Figure 1.1: USV Telemetron autonomously driving during tests in the Trondheimfjord with a safety pilot (Courtesy of Maritime Robotics).

One of the goals when developing USVs is to provide reliable manoeuvring irrespective of the vessel characteristics. The implementation of control system solutions and their tuning is a key task to achieve this goal, but conversions that are time and cost restricted pose some particular challenges on these tasks that this thesis intends to address. In these cases, commonly used methods, such as model-scale tests for modelling of control system solutions, are not viable due to the cost and the time required to execute these tests. In the project prior to this master thesis (de Freitas, 2019), a large selection of methods were investigated to identify control system solutions that can be implemented and tuned under these constraints. Alternatives were identified for thrust allocation, control law and reference filter, but covering all these topics in this thesis is infeasible, so this work focuses on control-oriented modelling. It is a proven solution already applied to real USVs, but improvements can still be made. For instance, fully automating the system identification procedure for a model to be used by the control law.

This master thesis was done in cooperation with Maritime Robotics AS to tackle the challenges faced on conversions and investigate viable alternatives to improve it. This has the potential to increase the safety and decrease the costs of transportation and operations at sea by making the unmanned technology available for a wider public at an even faster pace. Though the motivation for this study comes from cases of conversion from SV to USV, the methodology can also be applied to reduce time and cost for USV-specific designs.

1.2 Objectives, Scope and Limitations

The main objective of this thesis is to:

- Improve solutions to perform better conversions from SV to USV.

It is important to have a simple sentence to guide as the main objective, but underneath there are some secondary objectives that specify what better conversions were thought to mean in the context of this thesis:

- Minimize the cost.
- Minimize the time spent on trials and tuning.
- Maximize the performance of USVs, in terms of control precision and robustness.

Aiming to reach such objectives, the project prior to this thesis (de Freitas, 2019) analysed a series of control system solutions and selected the solutions considered best suited for this problem. Among these results, it was chosen for this thesis to give continuity to studies on control-oriented modelling with the focus of applying to conversions of SVs to USVs. Previous works on control-oriented modelling for USVs (Breivik et al., 2008; Eriksen and Breivik, 2017, 2018; Kvalvaag, 2018) identified potential future work, some of which were chosen to be addressed in this thesis.

As suggested in Kvalvaag (2018), the system identification procedure for the control-oriented model has been almost completely automated, but still requires some tuning. Two new methods are proposed and compared in this work in an attempt to fully automate the process. As suggested in Eriksen and Breivik (2017), the results from system identification were used for tuning the reference filters. The methods developed were applied to a high-speed USV actuated by an outboard engine (Figure 1.1), and it is discussed how these methods could be applied to other types of USVs.

This thesis is structured as follows:

- **Chapter 2 - Literature Review**: Relevant theory on control system solutions and mathematical models for vessels is followed by a thorough review of control-oriented modelling and the system identification procedure for this modelling. In addition, reference filters are explored further because system identification results were used to also tune them.
- **Chapter 3 - Methodology**: Description of two new methods of data extraction to fully automate the system identification procedure, two heuristic methods to automatically remove unreliable data from the procedure, and utilization of SI results to tune saturation limits of the reference filters.
- **Chapter 4 - Results**: Application of a complete system identification procedure to experimental data from testing a USV using new methods proposed in Chapter 3.
- **Chapter 5 - Discussion**: Analysis of the results from Chapter 4 and its impacts on conversions of different vessels.
- **Chapter 6 - Conclusion**: Summary and conclusion of the content of this thesis and a brief presentation of possible future works.

1.3 Contributions

The application of control-oriented modelling to USVs is not a novelty, but original methods were developed to fully automate the identification procedure. The foundation of methods for system identification in this application are credited to Breivik et al. (2008), Eriksen and Breivik (2017) and Kvalvaag (2018), and the procedure was almost automatic, still requiring tuning for data extraction and manual identification of unreliable data. Therefore, the original methods from this work were developed to automate these stages of identification. In addition, the resultant model was used to tune the reference filters of the vessel.

Using data collected by Eriksen and Breivik (2017), the complete system identification procedure was conducted and the following original methods were implemented:

- **RW-SSID**: Method for data extraction based on Dalheim and Steen (2020) using rolling windows for steady-state identification.
- **SIMID**: Method for data extraction based on control-oriented model structure doing simultaneous identification of inertia and damping parameters using optimization.
- **Removal of Unreliable Data**: Heuristic methods to automatically identify and remove data with low signal-to-ratio and data containing unreached steady-state.

- **Tuning of Reference Filter Saturation Limits**: Using the control-oriented model structure, this method uses the identified model to tune the saturation limit of surge acceleration, yaw rate and yaw acceleration for the system's reference filters.

Chapter 2

Literature Review

2.1 Control System Solutions

The most rudimentary way to control a vessel is to manually control each of its actuators. For example, using the steering wheel to define the angle of a rudder and the throttle lever to define the RPM (rotations per minute) of a thruster. However, this gets too complicated when a big number of actuators are available and necessary, requiring a more refined control system solution. Since it is a manual mode, this is totally dependent on the operator and provides no possibility of autonomy.

The next step in complexity is to add thrust allocation on the control system solution, as shown in Figure 2.1. In this case, instead of defining commands for each actuator, the operator defines commands for each controllable degree of freedom (DOF). The DOFs of a vessel are commonly defined in terms of a body fixed axis, as shown in Figure 2.2. The controllable DOFs usually are the planar DOFs - surge, sway and yaw - or a subset of them, while roll, pitch and heave are intrinsically stable due to the vessel's geometry and buoyancy forces, so there is no need to control them apart from in very specific cases. Therefore, in this solution the operator commands a throttle or force percentage for each controllable DOF, then the thrust allocation algorithm distributes this overall command as individual commands for each actuator.

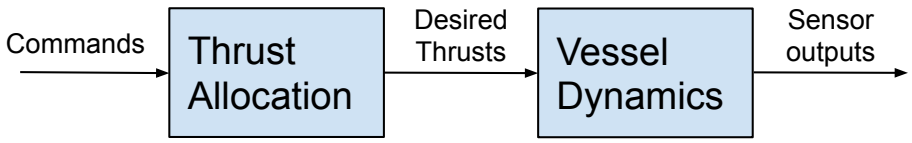


Figure 2.1: Block schematic of open-loop control system solution for the vessel system.

For example, on a cruise ship with two azimuth thrusters on the aft and a bow thruster, it means setting desired forces and moments on surge, sway and yaw and having the thrust allocation define each actuator's angle and rpm to reach the overall command. This allocation depends on parameters of the actuators, allocation objectives and implemented algorithms. In this case, the operator needs to give three commands, instead of five if manually controlling all actuators, because it would be necessary to control the angles of both azimuths, their RPMs and the RPM of the bow thruster. So it is clear to see the reduction of complexity brought by the thrust allocation.

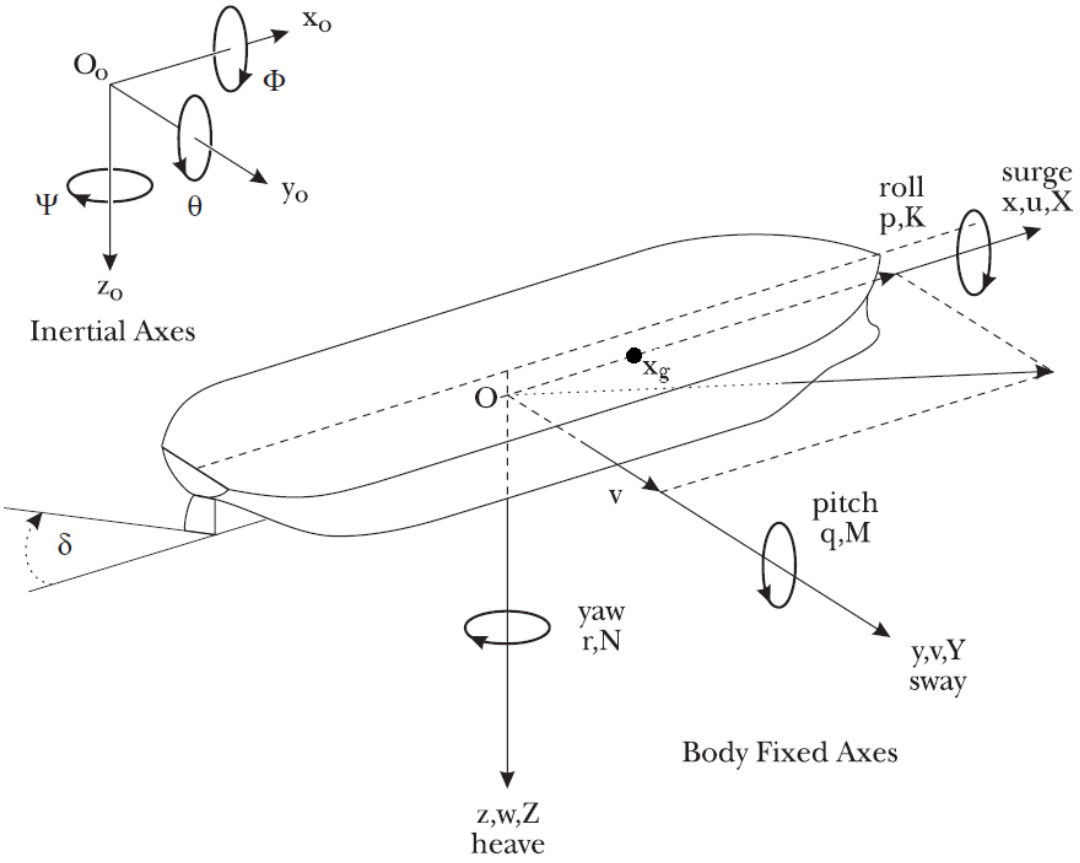


Figure 2.2: Coordinate system for vessels indicating inertial and body fixed axes (Blanke and Christensen, 1993)

The two previous control solutions are open-loop systems in which the operator needs experience to input proper commands so that the desired vessel dynamics are achieved. Although these control solutions are theoretically sufficient to convert a SV to unmanned, the performance is far from ideal. This is especially because the system is dependent on feedback from the operator, who is not on board. Hence, a closed-loop structure is necessary.

The translation of the operator's experience into algorithms can be divided into four new blocks for the system: observer, maneuver planning, reference filter and controller. This logic is shown in Figure 2.3 based on Sørensen (2011). In this closed-loop system, the operator's inputs are reduced to planning the mission at the maneuver planning block, so not requiring nearly as much continuous critical operation as when manually setting commands for the thrust allocation or directly for the actuators. The following paragraphs detail the function of each of these new blocks.

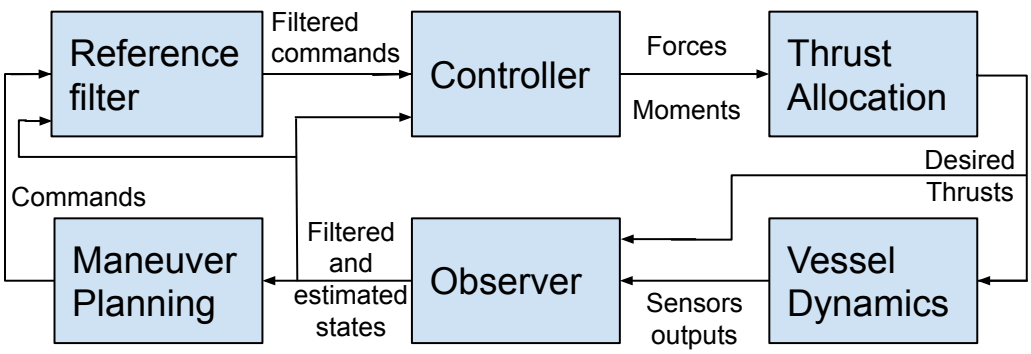


Figure 2.3: Block schematic of closed-loop control system solution for the vessel system.

Observer receives the states measured from the vessel dynamics and desired thrusts from the previous time step. When properly tuned, it filters sensor data and estimates observable unmeasured states before inputting them to other blocks. This is useful for filtering out random noise as well as high frequency waves, which usually do not need to be counteracted by the controller and would only lead to more wear and tear of the actuators. Moreover, estimating unmeasured states allows for further control possibilities and some degree of fault-tolerance if measurement signals are lost temporarily.

Maneuver planning receives the states from the observer and outputs commands to the reference filter and the controller. The transformation from states to commands depends on what the operator set as the mission plan. For example, if the plan is to follow a pre-defined path at a given speed, this block compares the path and the desired speed with the current states and outputs the necessary speed and heading based on its algorithm. This shifts the operator responsibility from closing the loop on Figure 2.1 to focusing on a higher-level control of the vessel, defining routes and objectives with far fewer updates than the low-level represented by Figure 2.3.

Reference filter, also known as reference model, is a key block to improve the performance of the controller. It filters step commands from maneuver planning into smooth continuous achievable commands in order to avoid saturation, and consequent wear and tear, of actuators. Further detailing about this block is discussed in Section 2.5 as the results from modelling the vessels dynamics can be used to, at least partially, tune the reference filter.

Controller calculates the total forces and moments to be distributed by the thrust allocation. This is done using the filtered commands from the reference filter and the states from the observer. There are countless alternatives for the controller algorithms and they were explored in de Freitas (2019). One of the conclusions was to use control-oriented models, which are described in Section 2.3 and further used in Chapter 3.

A good way to understand the schematic in Figure 2.3 is to follow an example. A typical setup is to have sensors measuring the vessel's speed over ground (SOG), course over ground (COG), position and heading. From these, the observer outputs filtered states as well as unmeasured but observable ones, such as accelerations and velocity decomposed on the body fixed axis. The maneuver planning has a mission set by the operator to follow a path at a given speed. Thus, it receives the states and outputs the necessary surge speed and heading based on its algorithm. The reference filter is input with both surge speed and heading commands from guidance and outputs their filtered values. The controller uses these in its algorithm to output total desired surge force and yaw moment. Finally, these are distributed among the thrusters by the thrust allocation so that the vessel follows the path with the given speed as determined in the maneuver planning block.

Since parts of each of these blocks can lead to an entire project of research, assumptions were made to decrease the scope of the initial study (de Freitas, 2019), focusing on parts that were considered more relevant for the conversion of SVs to USVs. All necessary states commonly available were considered measured and free of noise, neglecting the need for an observer. In addition, the commands were designed to match whatever was considered necessary for the specified control solution, neglecting the need for the maneuver planning block. The results provided recommendations of algorithms for thrust allocation, controller and reference filter. From these, it was decided to focus this work on control-oriented modelling, which is used by the suggested controller and can also be used to tune the reference filter.

Since control-oriented modelling for USVs is a recent technique first seen in Breivik et al. (2008), it is first presented in the following section the classical mathematical models used for modelling the dynamics of vessels. Then, Section 2.3 presents control-oriented modelling, comparing it to the classical models.

2.2 Classical Mathematical Models

This section summarises the most commonly used mathematical models to represent vessel dynamics and their possible variations. In order to keep consistency between the models presented, all of them

follow the same coordinate system shown in Figure 2.2 with all 6 DOFs. The inertial axis is fixed to the Earth on an arbitrary point on the water surface and uses North-East-Down (NED) coordinates. The axis x_0 points North, y_0 points East and z_0 down. The body-fixed axis follows the same convention with positive heave downwards from the axis' origin, positive surge towards the bow and positive sway to starboard (Perez and Fossen, 2007). The origin of the body fixed axis can be defined anywhere on the vessel, but it is common to place it at the vessel's geometric centre. It can also be placed on the centre of gravity (CG), but this is avoided because CG is not a fixed position.

Although realistically all DOFs are coupled, there are some assumptions that can simplify the models without serious negative consequences to the representation. This work focuses on the three planar DOFs - surge, sway and yaw - while the others - heave, roll and pitch - are considered to always be on the equilibrium point as they are inherently stable for vessels. That is because the coupling between planar DOFs and others is small, only being more relevant on extreme weather conditions. Moreover, the stability of heave, roll and pitch makes it rare to focus on making them controllable.

Intending demonstrate the relation between the inertial and the body fixed axis, the correspondence between planar velocities is shown in Equation (2.1). $\boldsymbol{\eta} = [x, y, \psi]^\top$ is the position of the vessel regarding the inertial axis. $\boldsymbol{\nu} = [u, v, r]^\top$ is the velocity of the vessel in the body fixed axis orientation. $\boldsymbol{R}(\psi)$ is the rotation matrix as a function of the heading ψ shown in Equation (2.2) (Fossen, 2011).

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

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} u\cos\psi - v\sin\psi \\ u\sin\psi + v\cos\psi \\ r \end{bmatrix} \tag{2.1b}$$

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

All of the mathematical models described in this section follow these definitions. However, the simplification to three planar DOFs is not sufficient to derive analytical models due to the complexity of the interaction between hulls and fluids. Therefore, more assumptions are necessary to represent this interaction.

One of the earliest mathematical models still in use to represent the vessel's dynamics is the first-order steering model by Nomoto et al. (1957) as presented in Equation (2.3). It relates the yaw rate of the vessel r with rudder angle commands δ as a function of parameters T and K. The vessel speed is considered constant and disturbances are neglected. Nomoto et al. (1957) also proposed a second-order model with more parameters, as presented in Equation (2.4), but with the same purpose. Even though these contain many simplifications to cut the tuning down to only a few parameters, they are so powerful that they are still used nowadays on autopilot implementations.

$$T\dot{r} + r = K\delta \tag{2.3}$$

$$T_1T_2\ddot{r} + (T_1 + T_2)\dot{r} + r = K\delta + KT_3\dot{\delta} \tag{2.4}$$

For a different approach, Abkowitz (1980) used third-order Taylor series expansion around a constant surge speed u_0 to propose a model including all planar DOFs. The variation of velocity $\boldsymbol{\Delta\nu}$ from the reference of the expansion $\boldsymbol{\nu}_0$ is presented in Equation (2.5). This leads to the hydrodynamic forces being represented by Equation (2.6) and vessel accelerations by Equation (2.7):

$$\boldsymbol{\Delta\nu} = \boldsymbol{\nu} - \boldsymbol{\nu}_0 = [u, v, r]^\top - [u_0, 0, 0]^\top = [\Delta u, v, r]^\top \tag{2.5}$$

$$X(\boldsymbol{\Delta\nu}, \delta) = X(\boldsymbol{\nu}_0, 0) + X_u\Delta u + X_{uu}\Delta u^2 + X_{uuu}\Delta u^3 + X_{vv}v^2 + (X_{rr} + mx_g)r^2 + X_{\delta\delta}\delta^2 + X_{vvu}v^2\Delta u + X_{rru}r^2\Delta u + X_{\delta\delta u}\delta^2\Delta u + (X_{vr} + m)vr + X_{v\delta}v\delta + X_{r\delta}r\delta + X_{vru}vr\Delta u + X_{v\delta u}v\delta\Delta u + X_{r\delta u}r\delta\Delta u \quad (2.6a)$$

$$Y(\boldsymbol{\Delta\nu}, \delta) = Y(\boldsymbol{\nu}_0, 0) + Y_u\Delta u + Y_{uu}\Delta u^2 + Y_v v + Y_{vvv}v^3 + Y_{vrr}vr^2 + Y_{v\delta\delta}v\delta^2 + Y_{vu}v\Delta u + Y_{vuu}v\Delta u^2 + (Y_r - mu)r + Y_{rrr}r^3 + Y_{rvv}rv^2 + Y_{r\delta\delta}r\delta^2 + Y_{ru}r\Delta u + Y_{ruu}r\Delta u^2 + Y_\delta\delta + Y_{\delta\delta\delta}\delta^3 + Y_{\delta vv}\delta v^2 + Y_{\delta rr}\delta r^2 + Y_{\delta u}\delta\Delta u + Y_{\delta uu}\delta\Delta u^2 + Y_{vr\delta}vr\delta \quad (2.6b)$$

$$N(\boldsymbol{\Delta\nu}, \delta) = N(\boldsymbol{\nu}_0, 0) + N_u\Delta u + N_{uu}\Delta u^2 + N_v v + N_{vvv}v^3 + N_{vrr}vr^2 + N_{v\delta\delta}v\delta^2 + N_{vu}v\Delta u + N_{vuu}v\Delta u^2 + (N_r - mx_g u)r + N_{rrr}r^3 + N_{rvv}rv^2 + N_{r\delta\delta}r\delta^2 + N_{ru}r\Delta u + N_{ruu}r\Delta u^2 + N_\delta\delta + N_{\delta\delta\delta}\delta^3 + N_{\delta vv}\delta v^2 + N_{\delta rr}\delta r^2 + N_{\delta u}\delta\Delta u + N_{\delta uu}\delta\Delta u^2 + N_{vr\delta}vr\delta \quad (2.6c)$$

$$\dot{u} = \frac{X(\boldsymbol{\Delta\nu}, \delta)}{m - X_{\dot{u}}} \quad (2.7a)$$

$$\dot{v} = \frac{(I_z - N_{\dot{r}})Y(\boldsymbol{\Delta\nu}, \delta) - (mx_g - Y_{\dot{r}})N(\boldsymbol{\Delta\nu}, \delta)}{(m - Y_{\dot{v}})(I_z - N_{\dot{r}}) - (mx_g - N_{\dot{v}})(mx_g - Y_{\dot{r}})} \quad (2.7b)$$

$$\dot{r} = \frac{(m - Y_{\dot{v}})N(\boldsymbol{\Delta\nu}, \delta) - (mx_g - N_{\dot{v}})Y(\boldsymbol{\Delta\nu}, \delta)}{(m - Y_{\dot{v}})(I_z - N_{\dot{r}}) - (mx_g - N_{\dot{v}})(mx_g - Y_{\dot{r}})} \quad (2.7c)$$

where m is the vessel mass, I_z is the inertia around yaw axis and x_g is the surge distance from body fixed axes origin to CG. These are directly obtained from the vessel properties, but the other parameters, commonly called hydrodynamic derivatives, need system identification (SI) techniques to have their values defined. The capital letters on the hydrodynamic derivatives correspond to which force it contributes to and the subscript letters are the speeds or rudder angle to be multiplied to it. For example, Y_{vrr} is multiplied by vr^2 and this multiplication contributes to the total sway force Y. These Taylor-series expansions do not contain all possible parameters, because many were neglected due to assumptions, such as considering a port starboard symmetry that decouples surge from sway and yaw (Abkowitz, 1980; Fossen, 2011). Therefore, although it is clear to see the increase of the number of parameters compared to the Nomoto models, the Abkowitz model also holds simplifications.

These are two of the most well-known mathematical models for vessels in literature and many others exist with different assumptions that have been used to represent different vessels. Fossen (2011) presents many of these alternatives and uses a matrix representation to simplify the notation as shown in Equation (2.8). This general matrix representation includes coriolis-centripetal $\boldsymbol{C}$ and damping $\boldsymbol{D}$ components on the left side and a general force vector $\boldsymbol{\tau}$ on the right side, but this is adaptable. For example, the Abkowitz model from Equations (2.6) and (2.7) is rewritten in Equation (2.9) with $\boldsymbol{C}$ and $\boldsymbol{D}$ considered incorporated in $\boldsymbol{\tau}$ and no disturbances or actuators other than the rudder.

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu}) = \boldsymbol{\tau}(\boldsymbol{\nu}, \delta) \tag{2.8}$$

$$\begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} X(\boldsymbol{\Delta\nu}, \delta) \\ Y(\boldsymbol{\Delta\nu}, \delta) \\ N(\boldsymbol{\Delta\nu}, \delta) \end{bmatrix} \tag{2.9}$$

2.3 Control-Oriented Model

Classical mathematical models for vessels are designed to be applicable at service speed in displacement regime (Fossen, 2005). The speed regime of a vessel is defined by its Froude number as defined in Equation (2.10):

$$Fn = \frac{U}{\sqrt{gL}} \tag{2.10}$$

where U is speed, g gravity acceleration and L vessel length. A Froude number below 0.3 defines a displacement regime. Classical mathematical models were designed for such regime, because they were designed for large vessels in open-water voyages keeping roughly the same speed for days and that cannot reach Froude numbers above 0.3. However, that is not the case for the smaller USVs considered in this work. They are expected to operate in conditions requiring more maneuverability and precision of control at varied speeds, including Froude numbers above 0.3 (Breivik et al., 2008).

Inspired by aircraft control theory and targeted to underactuated vessels, a version of control-oriented model was presented by Breivik et al. (2008) as a feedforward component for a gain-scheduled PID controller to fit USVs with a varied range of operation. This method intends to provide a model that is simple to fit to a vessel, as opposed to the classical models, that are demanding to fit robustly (Breivik et al., 2008). The controller was applied to a straight-line target tracking problem for a USV and was therefore divided into total linear speed ($U = \sqrt{u^2 + v^2}$) and yaw rate (r) controllers. The speed part combines a feedforward term, derived exclusively from experimental trials, and a PI term to compensate for uncertainties on the feedforward, while the yaw rate part only contains the PI term. The model consists of a scaled throttle (ς) as a function of surge speed (u) in a third order polynomial shown in Equation (2.11a) and a linear transformation from ς to the actual throttle input (τ) shown in Equation (2.11b). Breivik et al. (2008) used σ to define scaled throttle instead of ς, but it was changed in this work to avoid confusion with other uses of σ.

$$\varsigma(u) = c_{u3}u^3 + c_{u2}u^2 + c_{u1}u + c_{u03} \tag{2.11a}$$

$$\tau(\varsigma) = c_{\varsigma 1}\varsigma + c_{\varsigma 0} \tag{2.11b}$$

The constants (c) from the functions are obtained through least-squares curve fitting. More on SI in Section 2.4. Both τ and σ range from 0 to 1, clearly showing the more direct approach from the control-oriented model, instead of the classical models using forces and moments.

This model is further developed in Eriksen and Breivik (2017) by applying it to both speed and yaw rate and including inertia terms as shown in Equation (2.12). The motor throttle is defined only for positive commands as $\tau_m \in [0, 1]$ and the rudder input as $\tau_\delta \in [-1, 1]$. This formulation seems similar to the classic models, with $\boldsymbol{M}(\boldsymbol{x})$ as a inertia matrix and $\boldsymbol{\sigma}(\boldsymbol{x})$ as a damping term, but it uses normalized commands instead of forces and moments for $\boldsymbol{\tau}$, making this a non-first principles model. The similar

denomination of these components is kept as a simplification.

$$\boldsymbol{M}(\boldsymbol{x})\dot{\boldsymbol{x}} + \boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{\tau} \tag{2.12a}$$

$$\boldsymbol{M} = \begin{bmatrix} m_U(\boldsymbol{x}) & 0 \\ 0 & m_r(\boldsymbol{x}) \end{bmatrix}, \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_U(\boldsymbol{x}) \\ \sigma_r(\boldsymbol{x}) \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} U \\ r \end{bmatrix}, \quad \boldsymbol{\tau} = \begin{bmatrix} \tau_m \\ \tau_\delta \end{bmatrix} \tag{2.12b}$$

While Breivik et al. (2008) modelled σ as a third order polynomial function of surge speed, Eriksen and Breivik (2017) modelled both $\boldsymbol{M}$ and $\boldsymbol{\sigma}$ as fourth order functions of both states as shown in Equation (2.13). Inertia components also have an extra asymptotic basis function added to improve capability to fit to the vessel's dynamics. The $\boldsymbol{\beta}$ components are vectors of parameters that are found during SI to define the model to a respective vessel. In addition, the two hyperparameters a and b in $\boldsymbol{\phi}_m$ also need to be identified during SI independently for m_U and m_r. Hyperparameters are parameters that need to be defined before performing regression to find the other parameters.

$$m_U(\boldsymbol{x}) = \boldsymbol{\phi}_m(\boldsymbol{x})\boldsymbol{\beta}_{m_U}, \quad m_r(\boldsymbol{x}) = \boldsymbol{\phi}_m(\boldsymbol{x})\boldsymbol{\beta}_{m_r} \tag{2.13a}$$

$$\boldsymbol{\phi}_m(\boldsymbol{x}) = [1, U, r, U^2, Ur, r^2, U^3, U^2r, Ur^2, r^3, U^4, U^3r, U^2r^2, Ur^3, r^4, \tanh\left(a(U-b)\right)] \tag{2.13b}$$

$$\sigma_U(\boldsymbol{x}) = \boldsymbol{\phi}_\sigma(\boldsymbol{x})\boldsymbol{\beta}_{\sigma_U}, \quad \sigma_r(\boldsymbol{x}) = \boldsymbol{\phi}_\sigma(\boldsymbol{x})\boldsymbol{\beta}_{\sigma_r} \tag{2.13c}$$

$$\boldsymbol{\phi}_\sigma(\boldsymbol{x}) = [1, U, r, U^2, Ur, r^2, U^3, U^2r, Ur^2, r^3, U^4, U^3r, U^2r^2, Ur^3, r^4] \tag{2.13d}$$

Finally, in Kvalvaag (2018) a reduction of ϕ vectors is proposed based on expected symmetries of the dynamics. The damping always opposes the direction of movement, so considering the boat starboard-port symmetric and the rudder action unbiased to any side, the yaw rate components in $\boldsymbol{\phi}_{\sigma_r}$ have to be an odd function. This reduction is shown in Equation (2.14). On the other hand, these same symmetries result in the yaw rate components in $\boldsymbol{\phi}_{\sigma_U}$ having to be an even function (Equation (2.15)). For inertia components, both are also even functions for r due to the same symmetries and to keep inertia parameters positive (Equation (2.16)). Since only positive surge speeds are considered, no reduction is proposed for U components.

$$\boldsymbol{\phi}_{\sigma_r}(\boldsymbol{x}) = [1, U, r, U^2, Ur, U^3, U^2r, r^3, U^4, U^3r, Ur^3] \tag{2.14}$$

$$\boldsymbol{\phi}_{\sigma_U}(\boldsymbol{x}) = [1, U, U^2, r^2, U^3, Ur^2, U^4, U^2r^2, r^4] \tag{2.15}$$

$$\boldsymbol{\phi}_{m_U}(\boldsymbol{x}) = [1, U, U^2, r^2, U^3, Ur^2, U^4, U^2r^2, r^4, \tanh\left(a_U(U-b_U)\right)] \tag{2.16a}$$

$$\boldsymbol{\phi}_{m_r}(\boldsymbol{x}) = [1, U, U^2, r^2, U^3, Ur^2, U^4, U^2r^2, r^4, \tanh\left(a_r(U-b_r)\right)] \tag{2.16b}$$

2.4 System Identification

According to Ljung (1998), system identification is the process of transforming observed input and output data from a system into a mathematical model. Such process being divided in four stages:

1. Observed data;
2. Candidate models;
3. Criterion of fit;
4. Validation.

Each of these could be performed with many alternative methods, so they are explained in this thesis with examples from what was done in previous works about control-oriented models. The next sections follow the system identification structure used by the control-oriented model references shown in Figure 2.4, which is an extended subdivision of the list above by Ljung (1998). All performed the identification of the mathematical model for a small USV equipped with a rotatable outboard engine.

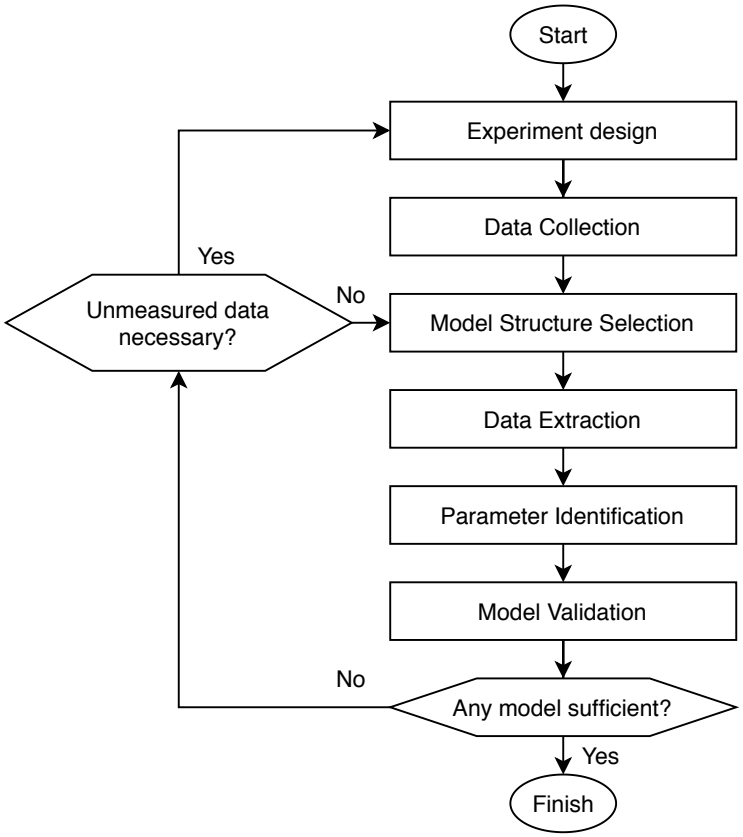


Figure 2.4: System identification procedure (Adapted from: Kvalvaag, 2018).

2.4.1 Experiment Design and Data Collection

The first step on the observed data stage is to acquire the necessary data. In order to do so, the operational range of the system must be known as well as the available commands and outputs to be measured. This information is used to design the experiments required to collect the necessary data and choose the sensors necessary to collect such data. Previous knowledge of general model structures applicable to the system being handled can be useful to minimize possible iterations.

For the small USVs used, the available commands are throttle and rudder angle commands. It is desired to identify damping and inertia components for surge speed and yaw rate, so two sets of tests were designed:

1. For each throttle command, vary rudder angle from zero to maximum stepwise and using the same steps back to zero.

2. For each rudder angle command, vary throttle from zero to maximum stepwise and using the same steps back to zero.

Each step should be long enough to allow for steady-state periods, which are used to identify damping components, while transient periods are used to identify inertia components. Since the rudder action is

considered symmetrical, there is no need to test both sides. This setup for experiment design is demonstrated in Figure 2.5, showing the expected boundaries of outputs in red and the inside area being covered by the step commands. Blue lines correspond to constant throttles and yellow lines to constant rudder angles.

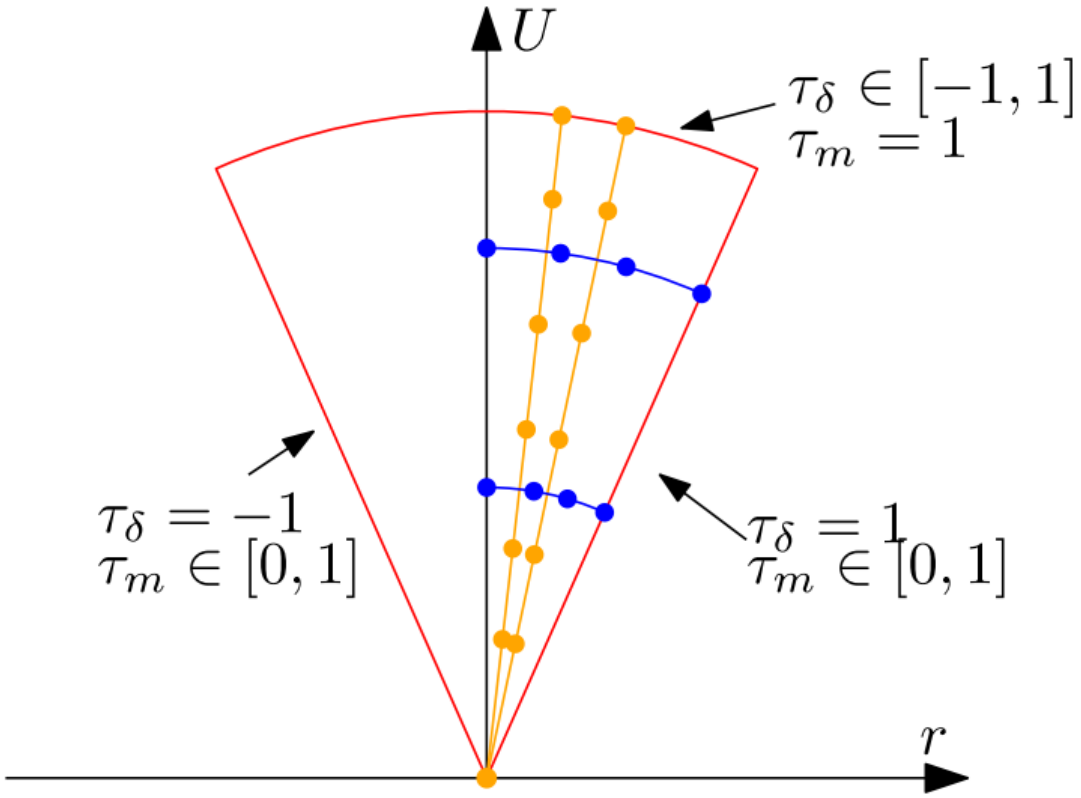


Figure 2.5: Red boundaries representing limits of inputs for the vessel and input sequence of commands resulting from experiment design represented by yellow (changing throttle) and blue (changing rudder angle) dotted lines (Eriksen and Breivik, 2017).

Kvalvaag (2018) also considered in the experiment design the possible deadbands of input commands as shown in Figure 2.6. This includes direct deadbands of throttle commands (green and red areas) and safety deadbands for a combination of high throttle and big rudder angle to avoid dangerous sharp turns at high speeds. Such deadbands can also be adapted to the model structure as done in Equation (2.11b) by Breivik et al. (2008).

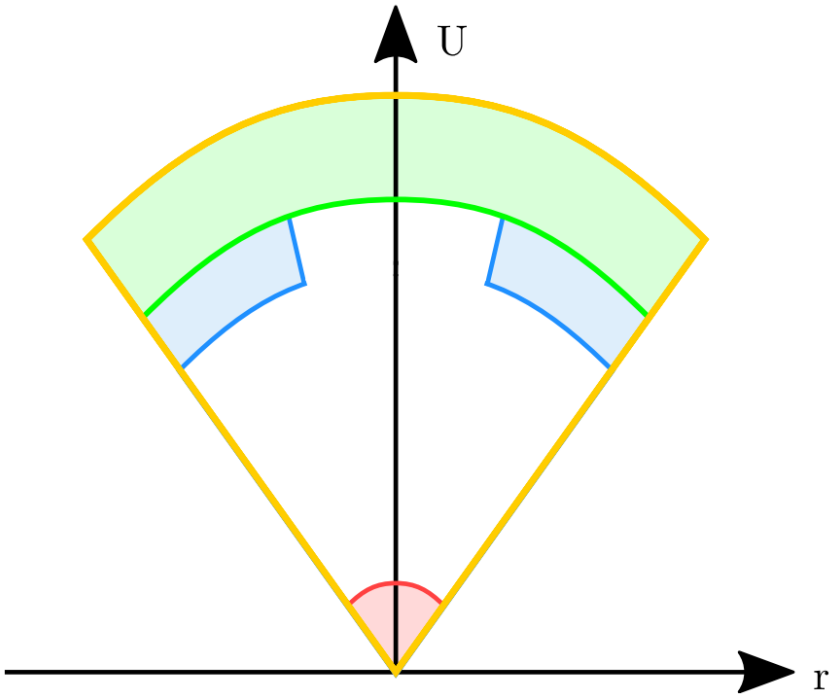


Figure 2.6: Deadbands of input commands from experiment design. The green and red regions represent areas where changes in commanded throttle do not affect motor response and the blue region is a user-defined region to avoid sharp turns at high speeds for safety (Kvalvaag, 2018).

Finally, position and heading sensors are necessary for the collection of data, so that speed and yaw rate can be derived. Speed can also be acquired directly, for instance with sensors that measure water flow under the hull. Besides, more than the expected speed and yaw rate, it is important to also acquire the feedback signals from rudder angle and throttle, because their actuation is not instantaneous to the received command.

2.4.2 Model Structure Selection

There exist three types of model structures (Ljung, 1998):

1. Black-box structures, not based in physics principles;
2. Structures from physical modelling;
3. Structures from semi-physical modelling.

While black-box models totally depend on the data collected to be defined, physical and semi-physical models can be defined prior to the experiment design stage, minimizing the necessity of iterations. Anyway, it cannot be guarantee beforehand that no iterations will be necessary, as they may occur due to unexpected dynamics not properly fitting to the model structure chosen.

The control-oriented models, that can be considered semi-physical models, used by Breivik et al. (2008), Eriksen and Breivik (2017) and Kvalvaag (2018) are presented in Section 2.3 and all of them used only one model structure without iterations. Although various models could be tested simultaneously, the next sections consider the choice of a singular model on this step to simplify explanation.

2.4.3 Data Extraction

After the model structure is defined, the data expected by the model for parameter identification must be extracted from the collected data. This procedure is totally dependent on the model chosen, so this is

illustrated by an example from the control-oriented moddeling for USVs literature:

Steady-State Identification

As explained in Section 2.4.1, transient and steady-state periods are used to identify inertia and damping components respectively. Thus, these periods must be identified in the data as exemplified in Figure 2.7.

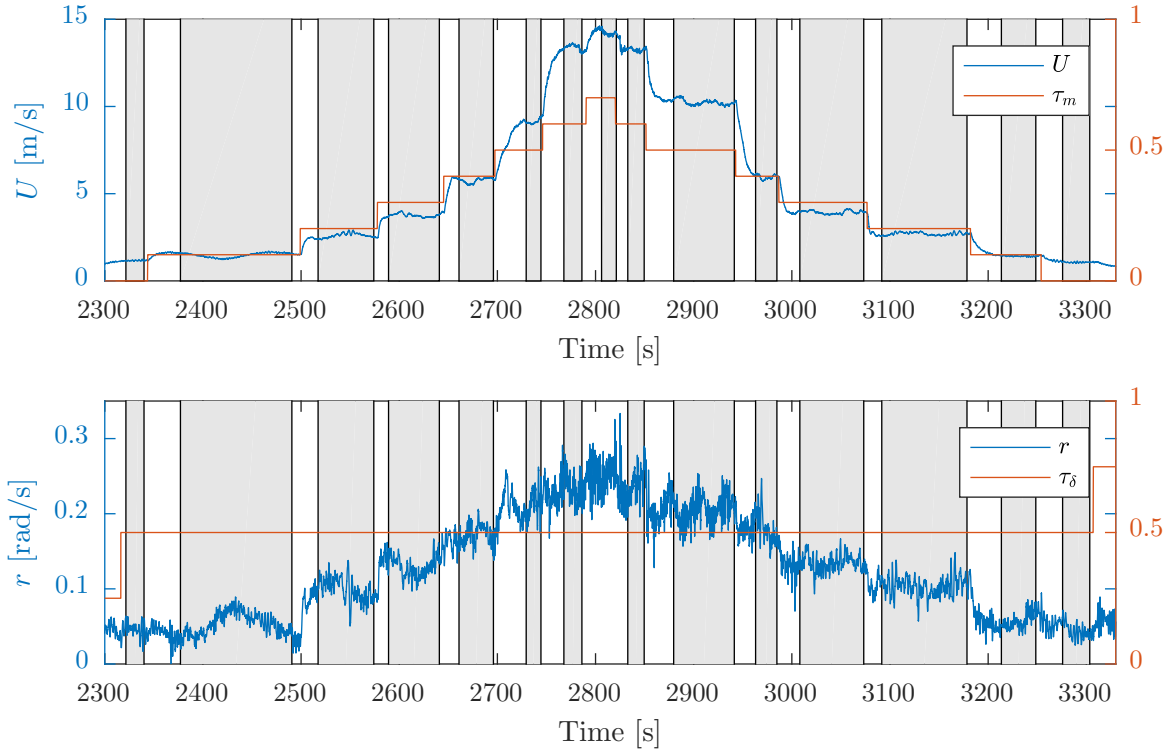


Figure 2.7: Example sequence of step commands and splits of the data between grey areas of steady-state and white areas of transient (Eriksen and Breivik, 2017).

While Eriksen and Breivik (2017) manually defined the transition instants, Kvalvaag (2018) automated the steady-state identification (SSID). Intending to achieve this automation, a test statistic (R) was defined as in Equation (2.17) and Equation (2.18) based on Cao and Rhinehart (1995). x_i represents the i-th state of the system and the subscript f indicates a filtered state.

$$R = \frac{(2-\lambda_1)\nu_{f,i}^2}{\delta_{f,i}^2} \tag{2.17}$$

where,

$$\nu_{f,i}^2 = \lambda_2(x_i - x_{f,i-1})^2 + (1-\lambda_2)\nu_{f,i-1}^2 \tag{2.18a}$$

$$\delta_{f,i}^2 = \lambda_3(x_i - x_{i-1})^2 + (1-\lambda_3)\nu_{f,i-1}^2 \tag{2.18b}$$

$$x_{f,i} = \lambda_1 x_i + (1-\lambda_1)x_{f,i-1} \tag{2.18c}$$

The null hypothesis, H_0, to identify that the system is in steady-state and the alternate hypothesis, H_1, for transient state are defined in Equation (2.19). R_{lb} and R_{lb} are lower and upper bounds of the

hypotheses respectively.

$$H_0 : R \leq R_{lb} \tag{2.19a}$$
$$H_1 : R \geq R_{ub} \tag{2.19b}$$

While this method succeeds in minimizing the workload necessary by avoiding manual identification of transitions, it is not fully automated. It is necessary to tune five parameters: λ_1, λ_2, λ_3, R_{lb} and R_{ub} for each state.

Damping data

With the separation of steady-state and transient periods complete, results from each period are extracted to be used for parameter identification. Damping data is extracted from steady-state periods since, according to the model in Equation (2.12), the lack of acceleration leads to damping equal to inputs (Equation (2.20)).

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{\tau} \tag{2.20}$$

Then, each steady-state period i has its states averaged ($\bar{\boldsymbol{x}}_i$) and assigned to the corresponding damping ($\boldsymbol{\sigma}_i$) in the data set ($\mathcal{D}_{\boldsymbol{\sigma}}$) as shown in Equation (2.21).

$$\mathcal{D}_{\boldsymbol{\sigma}} = \begin{bmatrix} \bar{\boldsymbol{x}}_1 & \bar{\boldsymbol{x}}_2 & ... & \bar{\boldsymbol{x}}_n \\ \boldsymbol{\sigma}_1 & \boldsymbol{\sigma}_2 & ... & \boldsymbol{\sigma}_n \end{bmatrix} \tag{2.21}$$

As there is an assumption of port-starboard symmetry on the rudder-generated forces, the response for zero rudder angle in steady-state must be $\sigma_r = 0$, but some offset may be present in the results due to disturbances, so $\sigma_r = 0$ must be forced in the data set for these cases (Eriksen and Breivik, 2017). Another modification to the data sets due to this assumption is to mirror results supposing opposite rudder angle. This would lead every column from damping data set with rudder angle different than zero ($\mathcal{D}_{\boldsymbol{\sigma},c} = [\bar{U}, \bar{r}, \sigma_U, \sigma_r]^\top$) to insert in the data set its port-starboard symmetric set ($\mathcal{D}_{\boldsymbol{\sigma},sym} = [\bar{U}, -\bar{r}, \sigma_U, -\sigma_r]^\top$).

Inertia data

Inertia data is extracted separately for surge and yaw, because it depends on whether the step was in throttle or rudder angle. Considering state x_i for a step in τ_i and inertia parameter m_i, Eriksen and Breivik (2017) proposed a linearized version of the control-oriented model (Equation (2.12)) to be used for each transient period as shown in Equation (2.22). $(\cdot)^+$ represents the value of a variable at the final timestep of the region and $(\cdot)^-$ at the first timestep.

$$m_i \Delta \dot{x}_i + k_i \Delta x_i = \Delta \tau_i \tag{2.22a}$$
$$k_i = \frac{\sigma_{x_i}^+ - \sigma_{x_i}^-}{{x_i}^+ - {x_i}^-} \tag{2.22b}$$
$$\Delta \tau_i = \tau_i - \tau_i^- \tag{2.22c}$$
$$\Delta x_i = x_i - x_i^- \tag{2.22d}$$

The only unknown in Equation (2.22a) is the inertia parameter m_i. It can be identified by curve fitting the acquired data x_i from trials to the ordinary differential equation's solution (Equation (2.23)). An example is shown in Figure 2.8. The fitting can be performed by doing an optimization as Equation

(2.24), minimizing the squared error between data x_i and the estimate $\hat{x}_i$ as a function of m_i. As a modification in notation to make explanations and equations clearer, $\hat{(\cdot)}$ represents that the variable is estimated, not measured.

$$\hat{x}_i(t) = \frac{\Delta\tau_i}{k_i}\left(1 - e^{\frac{-k_i t}{m_i}}\right) + {x_i}^- \tag{2.23}$$

$$\hat{m}_i = \arg\min_{m_i} \sum_{t=0}^{t^+ - t^-} \left[x_i(t) - \hat{x}_i(t)\right]^2 \tag{2.24}$$

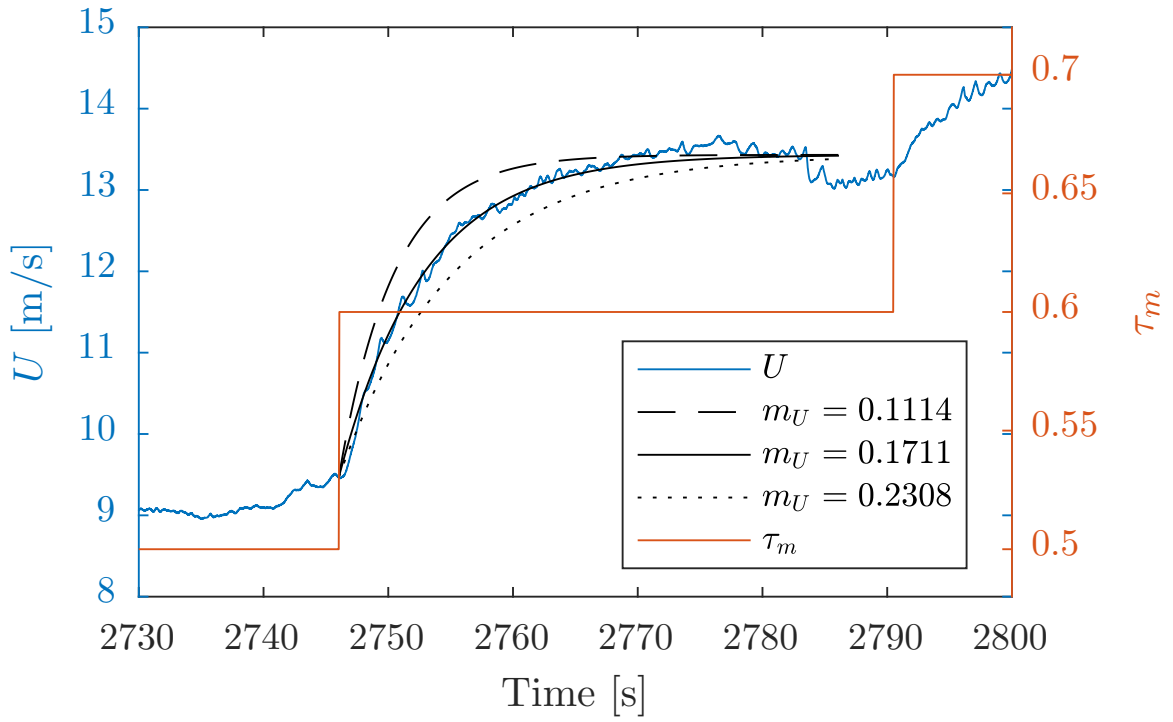


Figure 2.8: Extraction of inertia measurement m_U for a step in the surge speed, showing how the result is affected by changes in m_U (Eriksen and Breivik, 2017).

After the inertia parameter is identified for all transient periods, the data sets can be prepared. Instead of the average for the states for the entire time series, as done for the damping set, this data set only averages the first and last states of the period (Equation (2.25)) due to the linearization in Equation (2.22a), and place it in the midrange value for the states. Finally, the inertia sets are presented in Equation (2.26).

$$\bar{\boldsymbol{x}}_i^* = \frac{\boldsymbol{x_i}^+ - \boldsymbol{x_i}^-}{2} \tag{2.25}$$

$$\mathcal{D}_{m_U} = \begin{bmatrix} \bar{\boldsymbol{x}}_1^* & \bar{\boldsymbol{x}}_2^* & \dots & \bar{\boldsymbol{x}}_n^* \\ m_{U,1} & m_{U,2} & \dots & m_{U,n} \end{bmatrix} \tag{2.26a}$$

$$\mathcal{D}_{m_r} = \begin{bmatrix} \bar{\boldsymbol{x}}_1^* & \bar{\boldsymbol{x}}_2^* & \dots & \bar{\boldsymbol{x}}_n^* \\ m_{r,1} & m_{r,2} & \dots & m_{r,n} \end{bmatrix} \tag{2.26b}$$

The port-starboard symmetry on the rudder-generated forces discussed in the previous section also leads to extra columns for the data sets. Opposite rudder angle should produce the same inertia results, so every column from inertia data set with rudder angle different than zero inserts in the data set its port-starboard symmetric set. For instance, using surge inertia data sets, $\mathcal{D}_{m_U,c} = [\bar{U}, \bar{r}, m_U]^\top$ leads to also adding ($\mathcal{D}_{m_U,sym} = [\bar{U}, -\bar{r}, m_U]^\top$).

2.4.4 Parameter Identification

At this stage, the data extracted is used to identify the parameters of the chosen model. In the control-oriented model case, this means data sets $\mathcal{D}$ are used to define the parameter vectors $\boldsymbol{\beta}$ from Equation (2.13). Both Eriksen and Breivik (2017) and Kvalvaag (2018) use similar solutions for this, a weighted linear least squares with regularizarion and cross-validation.

The common weighted least squares method is formulated as shown in Equation (2.27). Applying it to the control-oriented model formulation, it can be used to identify $\boldsymbol{\beta}$ by considering y_i the normalized damping and inertia terms from the data sets and $\hat{y}_i$ the normalized estimations by the model as a function of the states from the data sets. Equation (2.28) gives the example of applying this formulation to identify $\boldsymbol{\beta}$ for the surge inertia part of the model using $\mathcal{D}_{m_U}$. Although it could be possible to make estimates for the weights w_i, Kvalvaag (2018) did not consider it a good option and followed Eriksen and Breivik (2017) heuristic approach of setting all equal to one and changing to smaller values in case of clear outliers. The normalization of states from data sets is done because U and r have different domains, but should have the same impact in the results of least-squares method (Eriksen and Breivik, 2017).

$$\varepsilon_{LS} = \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2 \tag{2.27}$$

$$\varepsilon_{LS} = \sum_{i=1}^{n} w_i (m_{U,i} - \boldsymbol{\phi}_m(\bar{\boldsymbol{x}}_i^*)\boldsymbol{\beta}_{m_U})^2 \tag{2.28}$$

In this formulation, there is an analytical solution for $\boldsymbol{\beta}$ that minimizes ε_{LS} as shown in Equation (2.29) still following the previous example.

$$\boldsymbol{\beta}_{m_U} = (\boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{Y} \tag{2.29a}$$

$$\boldsymbol{X} = [\bar{\boldsymbol{x}}_1^*, \bar{\boldsymbol{x}}_2^*, ..., \bar{\boldsymbol{x}}_n^*] \tag{2.29b}$$

$$\boldsymbol{Y} = [m_{U,1}, m_{U,2}, ..., m_{U,n}] \tag{2.29c}$$

$$\boldsymbol{W} = [w_1, w_2, ..., w_n] \tag{2.29d}$$

A recurring problem for this fitting technique in a case with few data points and considerable model complexity is overfitting and this can compromise predictions, as shown in an original example in Figure 2.9. According to Hastie et al. (2009), the prediction accuracy can be improved by shrinking regression coefficients in order to minimize overfitting, a method also known as regularization. A general formulation for Equation (2.27) including regularization is shown in Equation (2.30).

$$\varepsilon_{LS} = \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2 + \lambda R(\boldsymbol{\beta}) \tag{2.30}$$

The regularization weight λ controls the amount of shrinkage of parameters in $\boldsymbol{\beta}$ (Hastie et al., 2009) and becomes a hyperparameter of the problem. There are different options for the regularization function, for example Lasso (Equation (2.31)) and Ridge (Equation (2.32)).

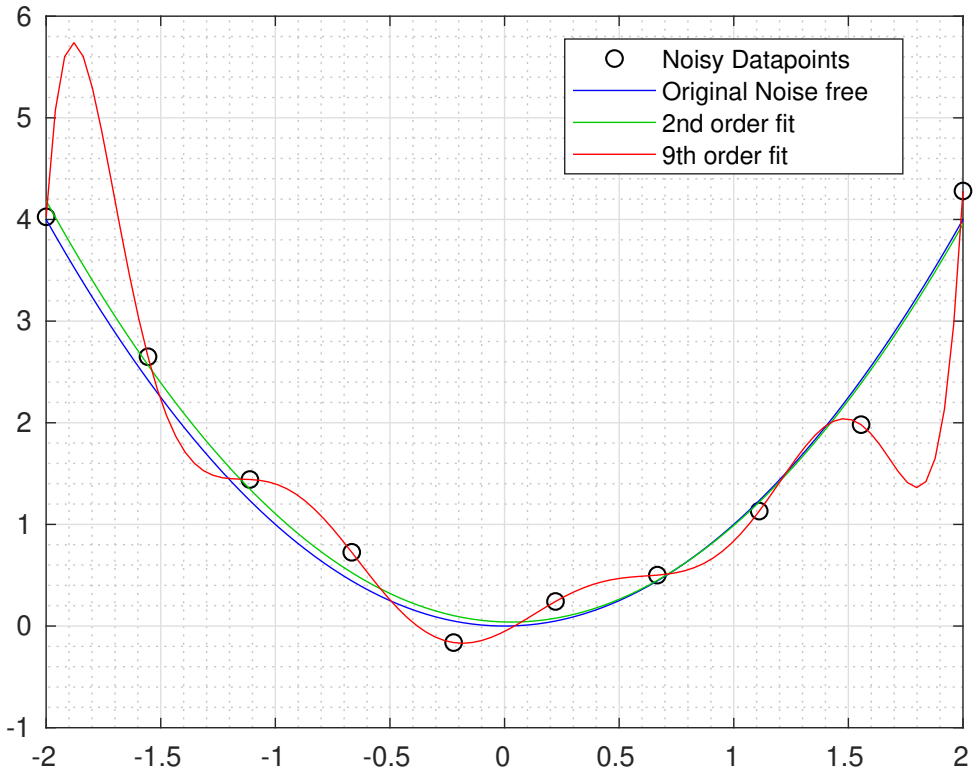


Figure 2.9: Example of the consequences of overfitting and importance of model reduction. Second order polynomial shows a good fit to the original data, while the ninth order polynomial is overfitted to points, not corresponding well to original data.

$$R(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_1 = \sum_i |\beta_i| \tag{2.31}$$

$$R(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_2 = \sqrt{\sum_i |\beta_i|^2} \tag{2.32}$$

The identification of hyperparameters, in the control-oriented model case λ from regularization and a and b from $\boldsymbol{\phi}_m$, was performed by the reference literature using cross-validation. As an example, the cross-validation method used by Eriksen and Breivik (2017) was leave-one-out cross-validation. This method excludes one data point to be used as validation for the parameters identified and the rest is used for training, repeating the process for all data points and looking for an overall minimum validation error.

2.4.5 Model Verification

Finally, the last stage of system identification is model verification. This is a crucial step to decide if results found on previous stages are satisfactory or if another iteration of the process is necessary. The verification can be performed by using a different trial, different to the one used for identification, and comparing the results for the states acquired and the output of simulating the identified model with the same inputs.

Figure 2.10 shows the surge speed results of model verification by Eriksen and Breivik (2017). It is clear to see that the trial was different from the one used for identification as described in Section 2.4.1 and the results from simulation were considered satisfactorily close to real data, apart from some small offsets. Therefore, in this case there was no iteration on the SI process and the model was approved.

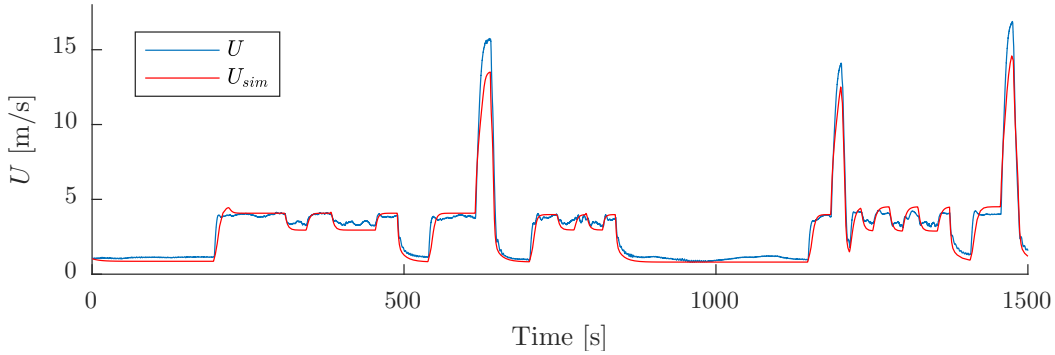


Figure 2.10: Surge speed results for model verification comparing acquired trial data in blue and data from simulating resulting model from SI in red (Eriksen and Breivik, 2017).

2.5 Reference Filter

The reference filter is responsible for a smooth transition between different commands that come from maneuver planning. This filtering smooths the steps on commands with two objectives. The first is to avoid saturation, and consequent wear and tear, of actuators. The second is to minimize integral wind-up problems in the controller. These problems happens when commands are changed much faster than the vessel can respond and the error caused by this reaction delay accumulates on the integral component of the controller, leading to undesired overshooting.

As in the previous sections, the reference filter can be modelled using different techniques. One of the options is a mathematical model of the vessel's dynamics as complete as possible to properly filter commands into feasible desired states. This is done by generating smooth reference trajectories using model simulations coupled with a controller. Examples are presented in Fossen (2011).

Another alternative is to model the filter like a mass-damper-spring system directly as a function of natural frequency ω_n and damping ratio ζ. Equation (2.33) presents the transfer function $R(\mathrm{s})$ of this approach, that keeps the tuning as simple as possible. With the purpose of also having smooth second derivatives, a low pass filter is typically cascaded with the mass-damper-spring system as presented in Equation (2.34) (Fossen, 1994, 2011). The low pass filter has period $T_n = 1/\omega_n$.

$$R(\mathrm{s}) = \frac{\omega_n^2}{\mathrm{s}^2 + 2\zeta\omega_n \mathrm{s} + \omega_n^2} \tag{2.33}$$

$$R(\mathrm{s}) = \frac{\omega_n^2}{(1 + T_n \mathrm{s})(\mathrm{s}^2 + 2\zeta\omega_n \mathrm{s} + \omega_n^2)} \tag{2.34a}$$

$$= \frac{\omega_n^3}{\mathrm{s}^3 + (2\zeta + 1)\omega_n \mathrm{s}^2 + (2\zeta + 1)\omega_n^2 \mathrm{s} + \omega_n^3} \tag{2.34b}$$

One possible additional improvement for both cases is to add saturating elements to the first and second derivatives of the filtered state. This seems especially necessary because the linear system has an exponential convergence, but this should be applicable to every step amplitude (Fossen, 2011). This means that every filtered step converges in the same period of time, but this is not realistic for this application, so the saturating elements are necessary. The saturation is tuned depending on the vessel's limitations, which can be identified during SI, as suggested by Eriksen and Breivik (2017) to be done while identifying the control-oriented model and explored in Section 3.5. Then, the other parameters are tuned depending on how fast the transition is desired, but also trying to avoid frequent saturation. Figure 2.11 exemplifies this by showing heading step commands and their respective filtered and filtered

+ saturated results. For step commands, the derivatives are infinite, but it is possible to compare them between the same filters with and without saturation as shown in Figure 2.12 for yaw rate limited to $\pm 8\,deg/s$ and Figure 2.13 for yaw angular acceleration limited to $\pm 2\,deg/s^2$.

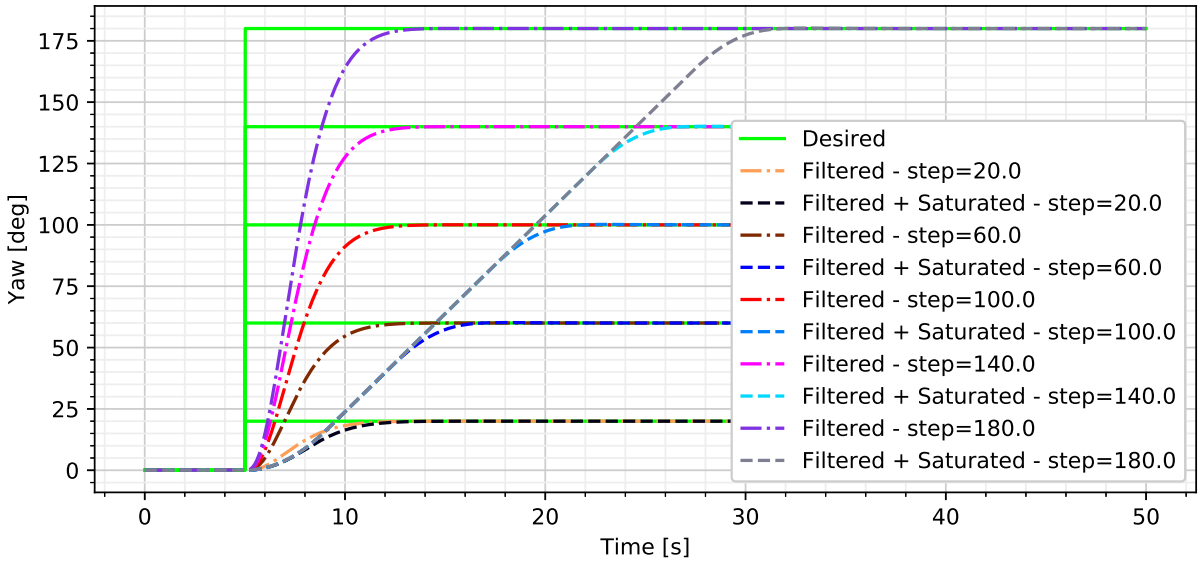


Figure 2.11: Comparison between heading step commands and their correspondent filtered and filtered + saturated results for mass-damper-spring based reference filter. Step commands instantly reach desired new values, filtered results converge all at the same time, and filtered + saturated results converge in a more realistic behaviour due to saturation limits (de Freitas, 2019).

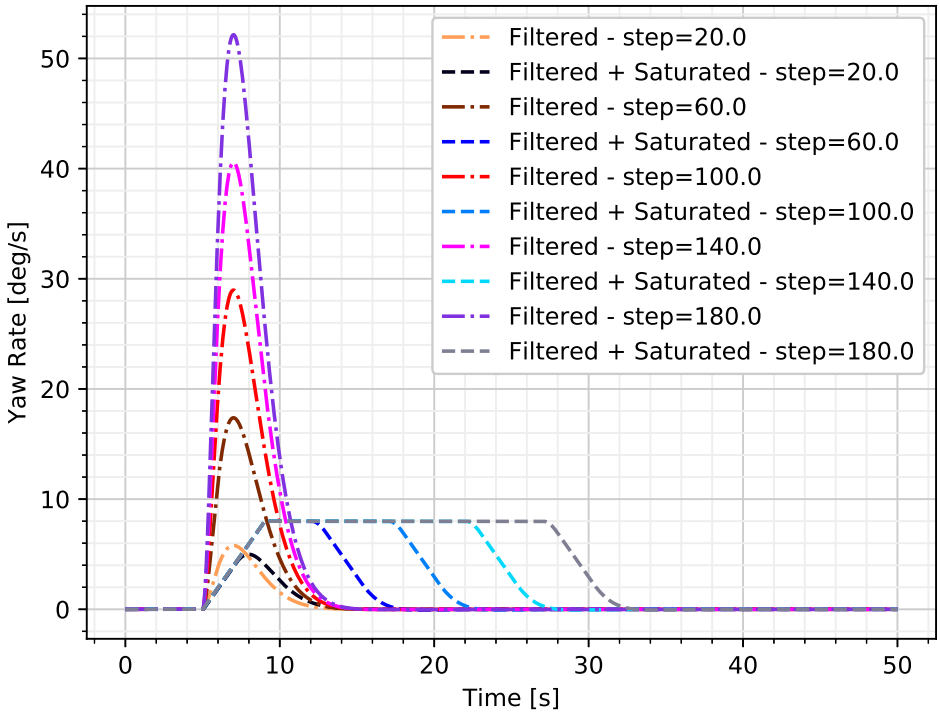


Figure 2.12: Comparison between filtered and filtered + saturated results for yaw rate for mass-damper-spring based reference filter in the same steps from Figure 2.11. Yaw rate saturation is defined at $\pm 8\,deg/s$. Step commands are not shown because they are not continuous, so cannot be derived and have their results here (de Freitas, 2019).

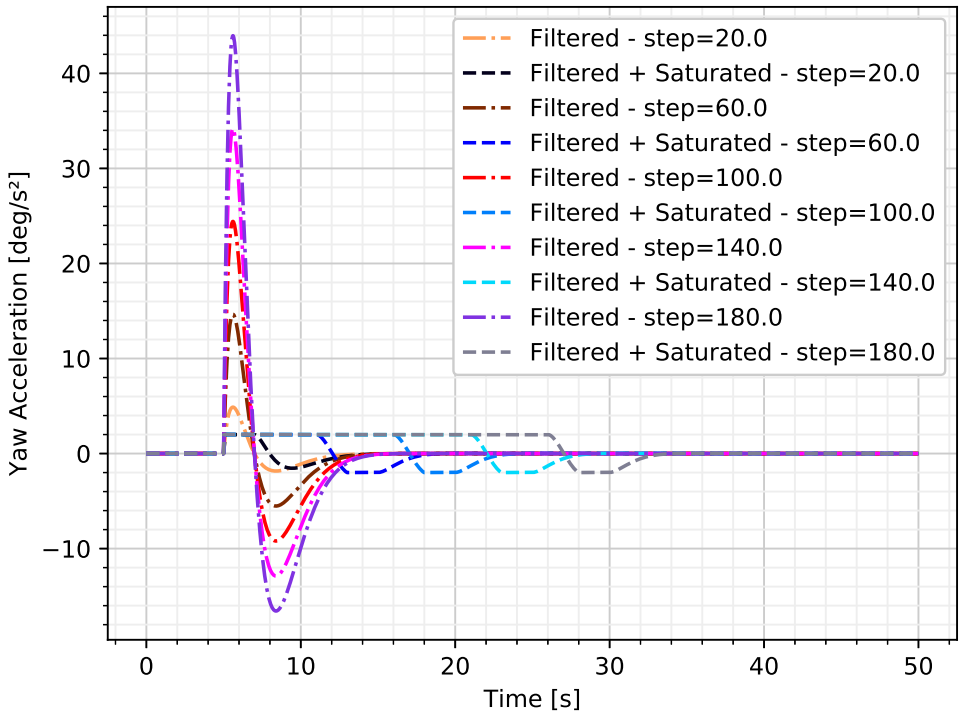


Figure 2.13: Comparison between filtered and filtered + saturated results for yaw angular acceleration for mass-damper-spring based reference filter in the same steps from Figure 2.11. Yaw angular acceleration saturation is defined at $\pm\, 2\, deg/s^2$. Step commands are not shown because they are not continuous, so cannot be double derived and have their results here (de Freitas, 2019).

Breivik et al. (2008) also uses similar saturating elements, but on another structure for the reference filter in which the saturation elements are intrinsic. Instead of using a lowpass filter, an approximation using hyperbolic tangents is proposed, as shown in equations (2.35) to (2.39) for the surge speed. The tunable parameters on this reference filter are the maximum surge speed achievable by the USV $u_{\max}$, the acceleration limit $\alpha(t)$ that must be designed between zero and the maximum achievable by the USV $\alpha_{\max}$, and the agility gain $k_{p,\tilde{\rho}}$ for the step commands. Figures 2.14 and 2.15 show the behaviour of this filter for varied steps of surge speed.

$$u_r(t) = u_{\max} \tanh\left(\frac{\rho_r(t)}{u_{\max}}\right) \tag{2.35}$$

$$\dot{\rho}_r(t) = \alpha(t) \tanh\left(\frac{k_{p,\tilde{\rho}}\tilde{\rho}(t)}{\alpha(t)}\right) \tag{2.36}$$

$$\tilde{\rho}(t) = \rho_d(t) - \rho_r(t) \tag{2.37}$$

$$\rho_d(t) = u_{\max} \tanh^{-1}\left(\frac{u_d(t)}{u_{\max}}\right) \tag{2.38}$$

$$\rho_r(0) = u_{\max} \tanh^{-1}\left(\frac{u(0)}{u_{\max}}\right) \tag{2.39}$$

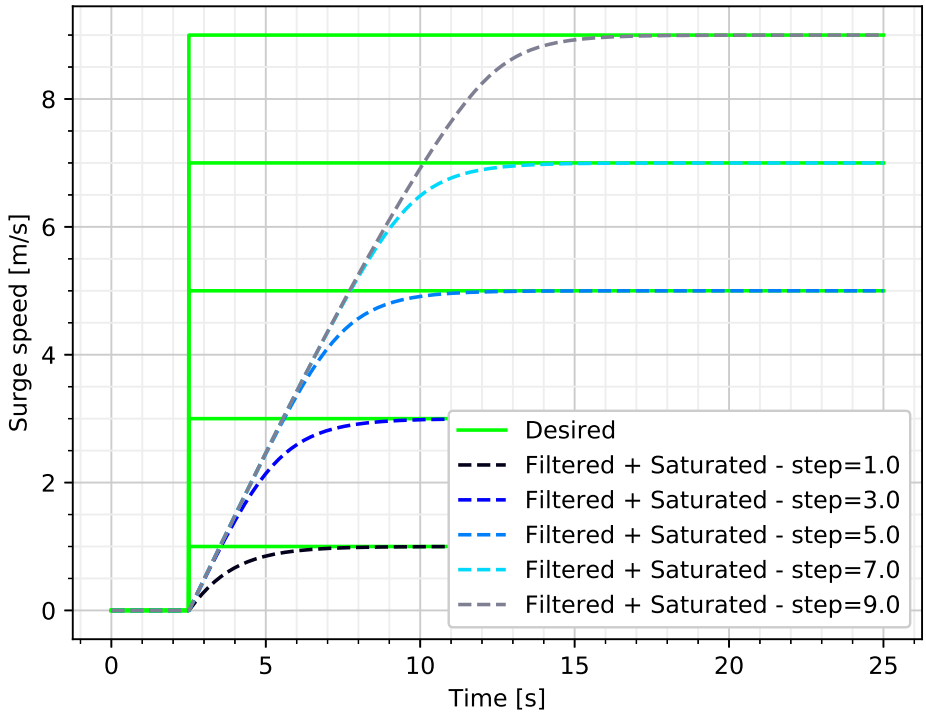


Figure 2.14: Comparison between surge speed step commands and their correspondent filtered + saturated results for reference filter based on hyperbolic tangents. Step commands instantly reach desired new values, and filtered + saturated results converge in a more realistic behaviour due to saturation limits (de Freitas, 2019).

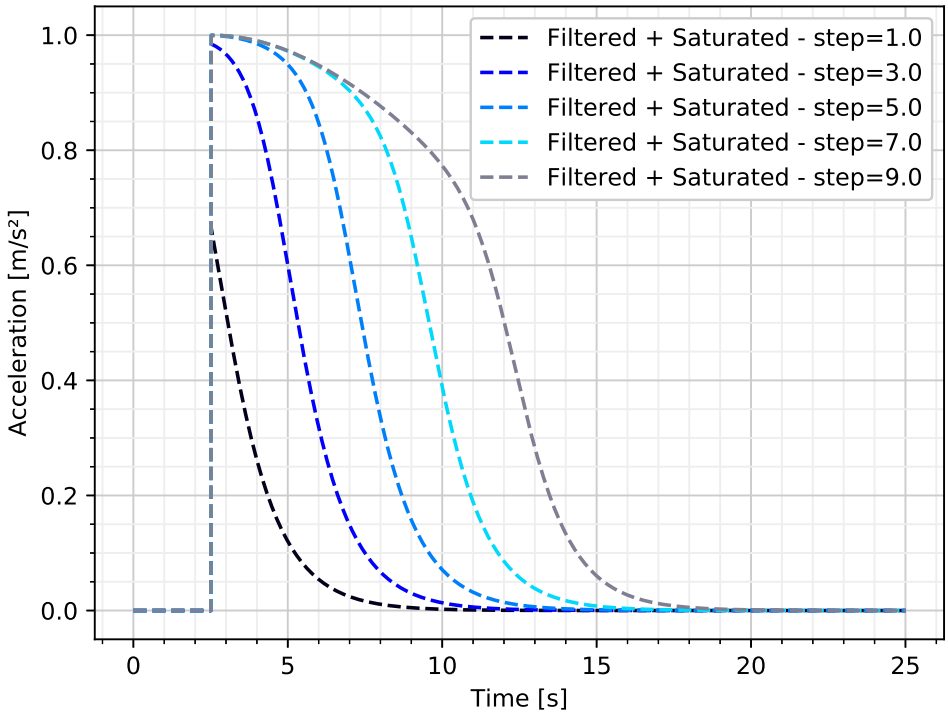


Figure 2.15: Filtered + saturated results for surge acceleration for reference filter based on hyperbolic tangents in the same steps as Figure 2.14. Step commands are not shown because they are not continuous, so cannot be derived and have their results here (de Freitas, 2019).

Chapter 3

Methodology

3.1 Problem Formulation

Control-oriented modelling is presented in Chapter 2 as a good option to be used during a conversion of SVs to USVs. In contrast to classical mathematical models for vessels, it is simpler to identify, while not compromising performance and being suitable for both displacement and non-displacement regimes (Breivik et al., 2008). Since the objective of this thesis is to perform better conversions, this chapter presents methods, newly developed or adapted, to improve control-oriented modelling by further automating SI using techniques different from what was previously tried and identifying tuning parameters for the system reference filter from the SI results.

The methods from the reference literature were implemented in the Marine Systems Simulator (Fossen and Perez, 2004) to test with different vessel dynamics and identify opportunities for improvements. Insights from the simulations led to the new methods described in this chapter. In addition, the simulations also showed that the resulting controller using the control-oriented model had no remarkable issues with straight path following scenarios, so while this topic was proposed in this thesis' work description, it was not further explored.

Some parts of SI are not covered in this chapter because no changes were implemented on them compared to the reference literature as presented in Chapter 2. The model structure used was the same as proposed by Kvalvaag (2018) with even and odd functions for inertia and damping parameters (Equations 2.14, 2.15 and 2.16). There was an intention of running new experiments with a different vessel, but due to the restrictions imposed by the government following COVID-19 pandemic at the beginning of 2020, these experiments were cancelled. Instead, data collected by Eriksen and Breivik (2017) was used for SI and model validation. Specifications of the Telemetron USV (Figure 1.1), used for data collection, are presented in Table 3.1.

Component	Description
Vessel hull	Polarcirkel Sport 845
Length	8.45 m
Beam	2.71 m
Weight	1675 kg
Propulsion system	Yamaha 225 HP outboard engine
Motor control	Electro-mechanical actuation of throttle valve
Rudder control	Hydraulic actuation of outboard engine angle with proportional-derivate (PD) feedback control
Navigation system	
Identification experiments	Kongsberg Seatex Seapath 330+
Control experiments	Hemisphere Vector VS330

Table 3.1: Telemetron USV specifications (Eriksen and Breivik, 2017).

3.2 Rolling Window Steady-State Identification (RW-SSID)

Kvalvaag (2018) presented an automation for SSID on USV data based on Cao and Rhinehart (1995), as described in Section 2.4. However, it was still necessary to manually tune 10 parameters, 5 for surge data and 5 for yaw data. In an attempt to minimize the amount of tuning parameters in this process, RW-SSID is presented in this section, a different SSID method using rolling windows to distinguish steady-state and transient regions based on Dalheim and Steen (2020).

The assumption of this method is that data regions of the system can be modelled as shown in Equation (3.1), the state x is a linear function of time added to a zero-mean white noise component w:

$$x = b_0 + b_1 t + w(t) \tag{3.1}$$

With the objective of determining if a region consists only of steady states, the function parameters and standard deviations of the noise and the linear slope must be calculated. The linear slope b_1 can be estimated using least squares error, as shown in Equation (3.2). The intercept b_0 estimation is calculated in Equation (3.3) using $\hat{b}_1$. The standard deviation of the noise $\hat{\sigma}_w$ is calculated in Equation (3.4) and used to calculate the standard deviation of the linear slope $\hat{\sigma}_{b_1}$ in Equation (3.5). The number of timesteps in the data region is defined by n_t, differing from the notation n used by Dalheim and Steen (2020) to differentiate it from other uses in this work.

$$\hat{b}_1 = \frac{\sum_{t=1}^{n_t} tx - \frac{1}{n_t}\sum_{t=1}^{n_t} t \sum_{t=1}^{n_t} x}{\sum_{t=1}^{n_t} t^2 - \frac{1}{n_t}(\sum_{t=1}^{n_t} t)^2} \tag{3.2}$$

$$\hat{b}_0 = \frac{1}{n_t}\left(\sum_{t=1}^{n_t} x - \hat{b}_1 \sum_{t=1}^{n_t} t\right) \tag{3.3}$$

$$\hat{\sigma}_w = \sqrt{\frac{1}{n_t - 2}\sum_{t=1}^{n_t}(x - \hat{b}_1 t - \hat{b}_0)^2} \tag{3.4}$$

$$\hat{\sigma}_{b_1} = \frac{\hat{\sigma}_w}{\sqrt{\sum_{t=1}^{n_t}(t - \bar{t})^2}} \tag{3.5}$$

The null-hypothesis of the region being a steady-state region is tested using a Student's t-distribution. Steady-state would mean a constant state over time, so a two-tailed t-test is applied to $\hat{b}_1$ to check how

significantly it is different from zero. This test is performed as shown in Equation (3.6), where $\mathrm{t}_{\alpha/2,n_t-2}$ is a tabulated value from the Student's t-distribution. α is the significance level and $n_t - 2$ the number of DOFs. If the test is true, the null hypothesis is rejected, i.e. the region is not a steady-state region (Dalheim and Steen, 2020).

$$\mathrm{t}_1 = \frac{|\hat{b}_1|}{\hat{\sigma}_{b_1}} \tag{3.6a}$$

$$\mathrm{t}_1 > \mathrm{t}_{\alpha/2,n_t-2} \tag{3.6b}$$

The approach to cover all data is to use a rolling window and the size n_t of the window is a tuning parameter. Longer windows are better for detecting long-term drift, while shorter windows are better for detecting fast changing unsteady behaviour. The rolling window is designed to have a maximum overlap during propagation, so it is shifted at single timesteps until all data is covered (Dalheim and Steen, 2020).

The detection of transition from one type of state to another has a simplification compared to Dalheim and Steen (2020) due to the nature of the system analysed. The transition from steady-state to transient is simple to identify, because it happens when the step command for either throttle or rudder angle is given. Therefore, there is no need to analyse this transition, as it is already known. On the other hand, the transition from transient to steady-state is smooth and completely depends on the dynamics of the vessel, making it harder to identify, especially in the presence of external disturbances. Consequently, in this transition the SSID technique is necessary.

Given these circumstances, the transition from transit to steady-state is then defined as the final point of the first window after a step in command to have a false result in Equation (3.6), confirming the null-hypothesis. This procedure differs from Dalheim and Steen (2020), who define a transition to steady-state from the first point of the window. This change was made based on findings by Kvalvaag (2018) that premature identification of a transition has negative impact on the results and it is better to focus on certainty of transition rather than accuracy.

After the identification, the extraction of data for the data sets remains the same as the reference literature for both inertia and damping sets, $\mathcal{D}_{m_U}$, $\mathcal{D}_{m_r}$, $\mathcal{D}_{\sigma_U}$ and $\mathcal{D}_{\sigma_r}$. This procedure is explained in Section 2.4.3.

An example case for RW-SSID is shown in Figure 3.1. The data was created analytically and white noise was added. There are throttle step commands at time 0 and 20 seconds. Various values for the tuning parameters α and size of rolling window n_t were tested to show their impact on results. The trade-off for tuning these parameters is that premature SSID leaves few data to properly identify inertia parameters in the transient region, while late SSID leaves few data to properly identify damping parameters in the steady-state region.

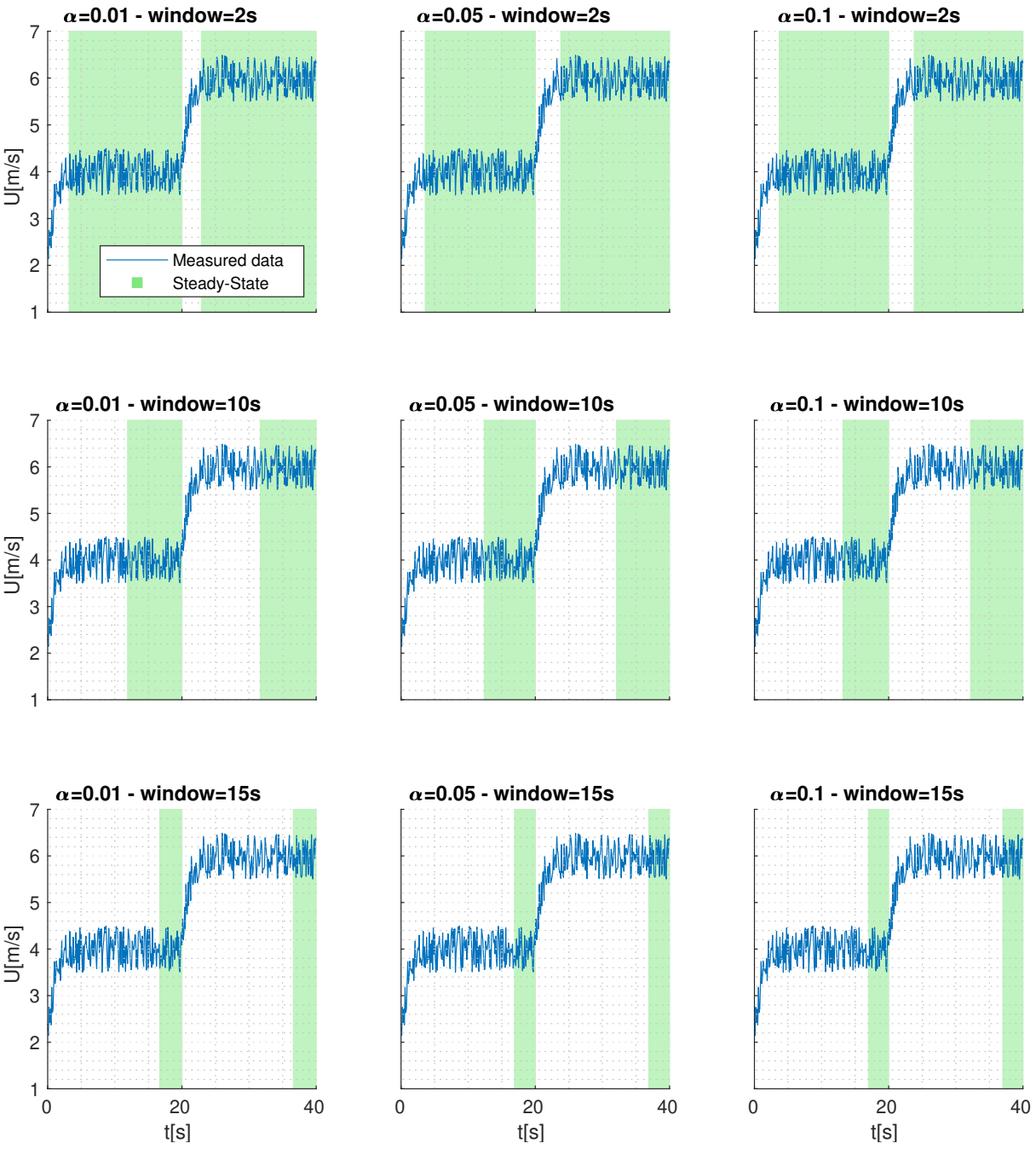


Figure 3.1: Applying RW-SSID to example data created with white noise with varied α and window size. Examples with bigger window lengths have smaller steady-state regions because this region only starts at the end of the first window that identifies steady-state. The impact caused by tuning α is much smaller, but still perceptible with smaller α leading to larger steady-state regions.

3.3 Simultaneous Identification (SIMID)

Another alternative for data extraction, original to this work, is to simultaneously identify inertia and damping data points after every step by utilizing optimization techniques, skipping the SSID process. This is based on the method used by Eriksen and Breivik (2017) and Kvalvaag (2018) to identify inertia parameters as discussed in Section 2.4.3. The equations used there are repeated below.

$$m_i \Delta \dot{x}_i + k_i \Delta x_i = \Delta \tau_i \tag{2.22a}$$

$$k_i = \frac{\sigma_{x_i}^+ - \sigma_{x_i}^-}{{x_i}^+ - {x_i}^-} \tag{2.22b}$$

$$\Delta \tau_i = \tau_i - \tau_i^- \tag{2.22c}$$

$$\Delta x_i = x_i - x_i^- \tag{2.22d}$$

$$\hat{x}_i(t) = \frac{\Delta \tau_i}{k_i} \left(1 - e^{\frac{-k_i t}{m_i}} \right) + {x_i}^- \tag{2.23}$$

In this case, instead of using optimization to estimate only m_i, x_i^+ also becomes an estimated parameter denoted $\hat{x}_i^+$, as shown in Equation (3.8):

$$\hat{m}_i, \hat{x}_i^+ = \underset{m_i, x_i^+}{\arg\min} \sum_{t=0}^{t_f - t_i} [x_i(t) - \hat{x}_i(t)]^2 \tag{3.8}$$

The expected advantages of this approach are:

- Extending the region for inertia identification has no negative impact on the results, in fact the opposite holds because a continuous curve of the state for the entire step is identified, while dividing regions in SSID does not guarantee this continuity as exemplified by Kvalvaag (2018) results when discussing measurement accuracy.
- No parameters need to be tuned for a SSID.

One disadvantage of this method is the possibility of the optimization getting stuck on a local minimum. This risk is mitigated by using the last state of the data as the starting point for x_i^+ in the optimization, since it is expected that the optimized value is close to it. Another action taken to minimize problems and simplify the optimization, which was developed during iterations over the method, was to use k_i as the parameter to be optimized instead of x_i^+. This gives the optimization a clear bound for every problem, because k_i is always positive as it is possible to see by inspecting Equation (2.22b). States with bigger values always have bigger damping, so $\sigma_{x_i}^+ - \sigma_{x_i}^-$ always has the same sign as ${x_i}^+ - {x_i}^-$, meaning k_i is always positive. Moreover, this removes discontinuities from the search space that could happen by using x_i^+ as the parameter for optimization and having ${x_i}^-$ inside the search space, which could lead to a division by zero. Equation (3.9) represents the final optimization scheme for this method.

$$\hat{m}_i, \hat{k}_i = \underset{m_i, k_i}{\arg\min} \sum_{t=0}^{t_f - t_i} [x_i(t) - \hat{x}_i(t)]^2 \tag{3.9a}$$

$$\hat{x}_i^+ = \frac{\sigma_{x_i}^+ - \sigma_{x_i}^-}{\hat{k}_i} + {x_i}^- \tag{3.9b}$$

The extraction of the data sets also require changes. First explaining a necessary additional notation, y_i refers to the secondary state, that is considered steady while the other varies. For example, if there is a step command in throttle, x_i is the surge speed U and y_i is the yaw rate r, while the opposite is considered during a step command in the rudder angle.

The state $\hat{x}_i^+$ is not only the last value of the state after the transient period, but also its mean for the entire steady-state period, so the same is done for y_i^+. Although there is the assumption for the model that y_i is steady over the entire window between commands, practice shows that this is not always the case. For instance, it is clear at high speed that the surge speed drops after an increase in the rudder

angle. A suggestion for future work is to incorporate these coupled effects in the model, but in this work only a treatment of y_i is done to avoid distortions.

The solution is to define $\hat{y}_i^+$ as the average of y_i after the moment $\hat{x}_i$ has passed the settling time, which is when $\hat{x}_i$ gets closer to $\hat{x}_i^+$ than 2% of $(\hat{x}_i^+ - x_i^-)$, as exemplified in Figure 3.2. The settling time t_s can be analytically calculated for Equation (2.23) as shown in Equation (3.11) using the definition of $\hat{x}_i^+$ in Equation (3.10) and the estimated parameters $\hat{m}_i$ and $\hat{k}$.

$$\hat{x}_i^+ = \lim_{t\to\infty} \hat{x}_i(t) = \frac{\Delta\tau_i}{\hat{k}_i} + {x_i}^- \tag{3.10}$$

$$\hat{x}_i(t_s) = \hat{x}_i^+ - 0.02(\hat{x}_i^+ - x_i^-) = \frac{\Delta\tau_i}{\hat{k}_i}\left(1 - e^{\frac{-\hat{k}_i t_s}{\hat{m}_i}}\right) + {x_i}^- \tag{3.11a}$$

$$\frac{\Delta\tau_i}{\hat{k}_i} + {x_i}^- - 0.02\frac{\Delta\tau_i}{\hat{k}_i} = \frac{\Delta\tau_i}{\hat{k}_i}\left(1 - e^{\frac{-\hat{k}_i t_s}{\hat{m}_i}}\right) + {x_i}^- \tag{3.11b}$$

$$0.98\frac{\Delta\tau_i}{\hat{k}_i} = \frac{\Delta\tau_i}{\hat{k}_i}\left(1 - e^{\frac{-\hat{k}_i t_s}{m_i}}\right) \tag{3.11c}$$

$$e^{\frac{-\hat{k}_i t_s}{\hat{m}_i}} = 0.02 \tag{3.11d}$$

$$\frac{-\hat{k}_i t_s}{\hat{m}_i} = \ln 0.02 \tag{3.11e}$$

$$t_s = \frac{-\hat{m}_i}{\hat{k}_i}\ln 0.02 \approx 3.9\frac{\hat{m}_i}{\hat{k}_i} \tag{3.11f}$$

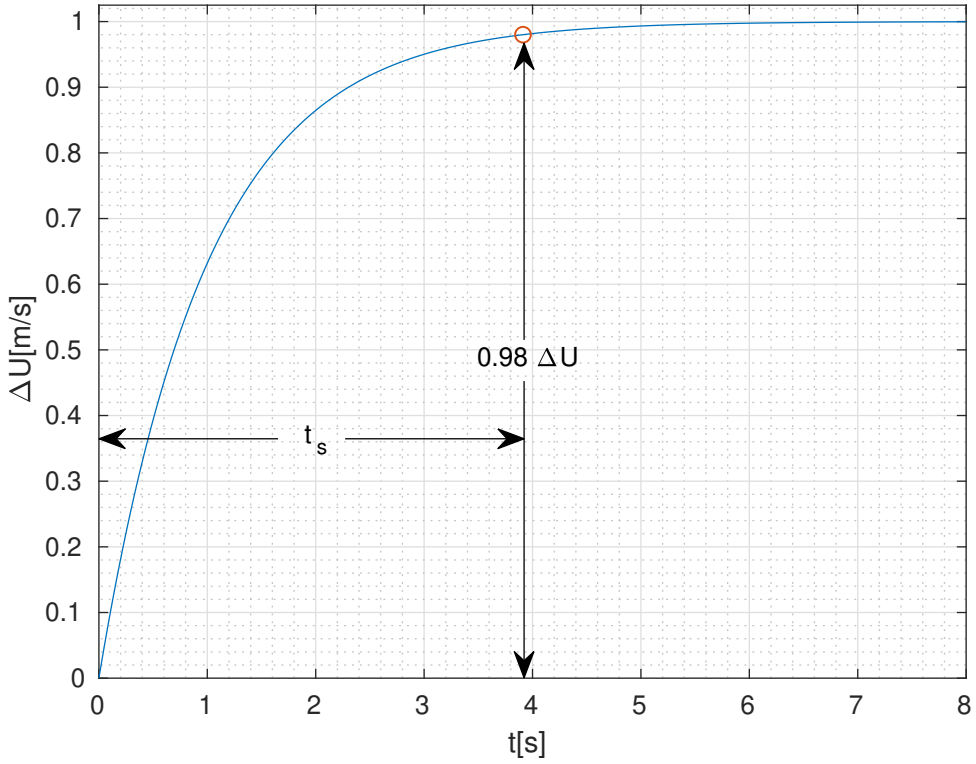
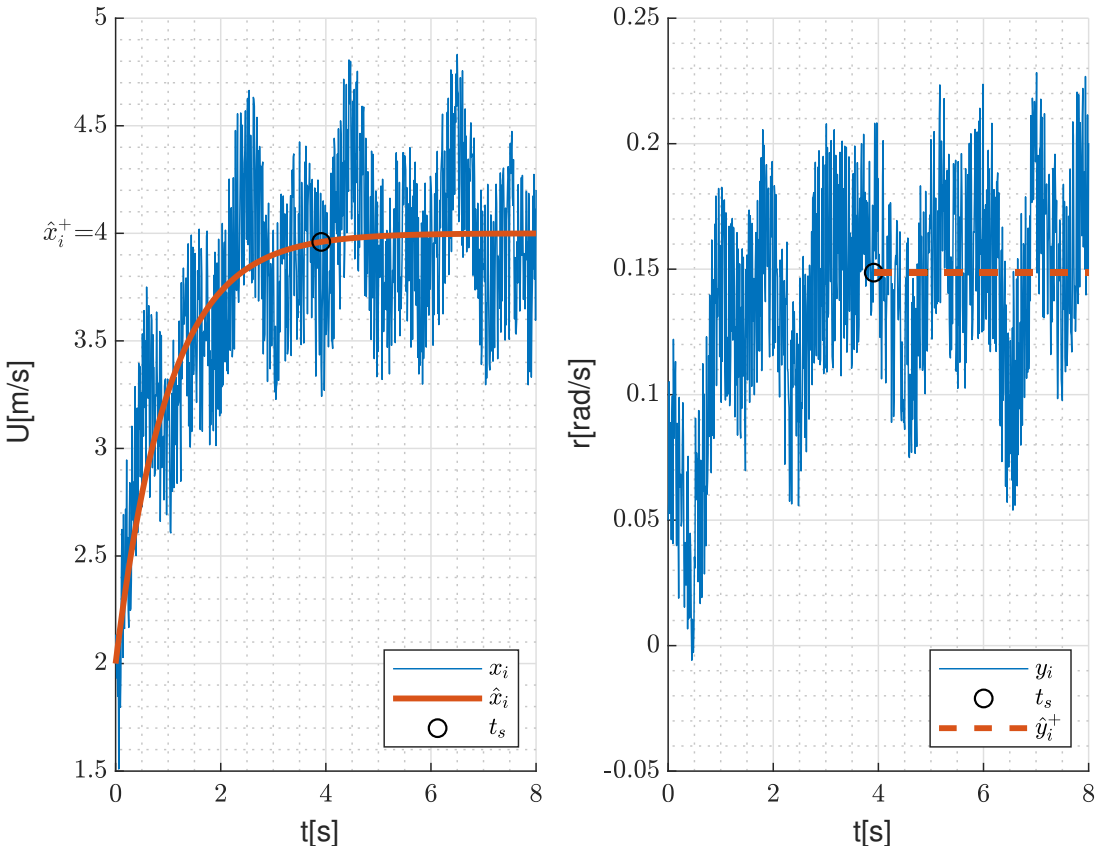


Figure 3.2: Example of step response and its respective settling time.

A visual example of the identification of the estimated parameters ($\hat{x}_i$, $\hat{x}_i^+$ and $\hat{y}_i^+$) is provided in

Figure 3.3 for a case of throttle step command. Summarizing the sequence of steps of this method:

1. Select measured data between two step commands;
2. Perform optimization (Equation (3.9)) on surge speed if throttle step or on yaw rate if rudder angle step;
3. Using parameters identified in optimization ($\hat{m}_i$ and $\hat{x}_i^+$), calculate settling time (Equation (3.11)), followed by calculation of $\hat{y}_i^+$.



(a) Simulated surge speed for throttle step of 20 % and estimated surge speed with starting point at $2\,m/s$.

(b) Simulated yaw rate for throttle step of 20 % and estimated yaw rate after reaching settling time.

Figure 3.3: Example of response for throttle step command, its measured data and estimated parameters.

Finally, the data sets can be built using the results achieved on each step. Damping data sets include results from all steps, while inertia data sets are divided in a set for throttle steps $\mathcal{D}_{m_U}$ and a set for

rudder angle steps $\mathcal{D}_{m_r}$.

$$\mathcal{D}_{\boldsymbol{\sigma}} = \begin{bmatrix} \hat{U}_1^+ & \hat{U}_2^+ & ... & \hat{U}_n^+ \\ \hat{r}_1^+ & \hat{r}_2^+ & ... & \hat{r}_n^+ \\ \boldsymbol{\sigma}_1 & \boldsymbol{\sigma}_2 & ... & \boldsymbol{\sigma}_n \end{bmatrix} \tag{3.12a}$$

$$\mathcal{D}_{m_U} = \begin{bmatrix} \frac{\hat{U}_{1_U}^+ + U_{1_U}^-}{2} & \frac{\hat{U}_{2_U}^+ + U_{2_U}^-}{2} & ... & \frac{\hat{U}_{n_U}^+ + U_{n_U}^-}{2} \\ \frac{\hat{r}_{1_U}^+ + r_{1_U}^-}{2} & \frac{\hat{r}_{2_U}^+ + r_{2_U}^-}{2} & ... & \frac{\hat{r}_{n_U}^+ + r_{n_U}^-}{2} \\ m_{U,1_U} & m_{U,2_U} & ... & m_{U,n_U} \end{bmatrix} \tag{3.12b}$$

$$\mathcal{D}_{m_r} = \begin{bmatrix} \frac{\hat{U}_{1_r}^+ + U_{1_r}^-}{2} & \frac{\hat{U}_{2_r}^+ + U_{2_r}^-}{2} & ... & \frac{\hat{U}_{n_r}^+ + U_{n_r}^-}{2} \\ \frac{\hat{r}_{1_r}^+ + r_{1_r}^-}{2} & \frac{\hat{r}_{2_r}^+ + r_{2_r}^-}{2} & ... & \frac{\hat{r}_{n_r}^+ + r_{n_r}^-}{2} \\ m_{r,1_r} & m_{r,2_r} & ... & m_{r,n_r} \end{bmatrix} \tag{3.12c}$$

3.4 Removal of Unreliable Data

Even when following the experiment design precisely, problems with the acquired data may create issues in the final model. During tests with the data extraction methods described in the previous sections, it was noticed that such problems had two main causes:

- Large relative noise in the data due to low signal-to-noise ratio.
- Not enough time on a step command to reach steady-state.

The data extraction methods assume the data does not have such problems, so there is a need to remove unreliable data to guarantee consistency in the results. In order to keep the SI process fully automated, the removal of data with problems was also automated. The heuristic methods used are described in the following sections.

3.4.1 Identification of Noisy Data

This method is applicable to both data extraction methods from the previous sections. Kvalvaag (2018) identified that a low signal-to-noise ratio is caused by step commands that are too small to be distinguishable from the disturbances of sea conditions and is particularly common at low speed. This can be mitigated by experiment design and by assuring calm enough sea conditions during trials, but it is not possible to guarantee all data is free from this problem since step commands that are too big would lower the performance of the model. Therefore, an automated method is necessary to avoid adding the data extracted from problematic regions to the data sets and to keep the entire process automated.

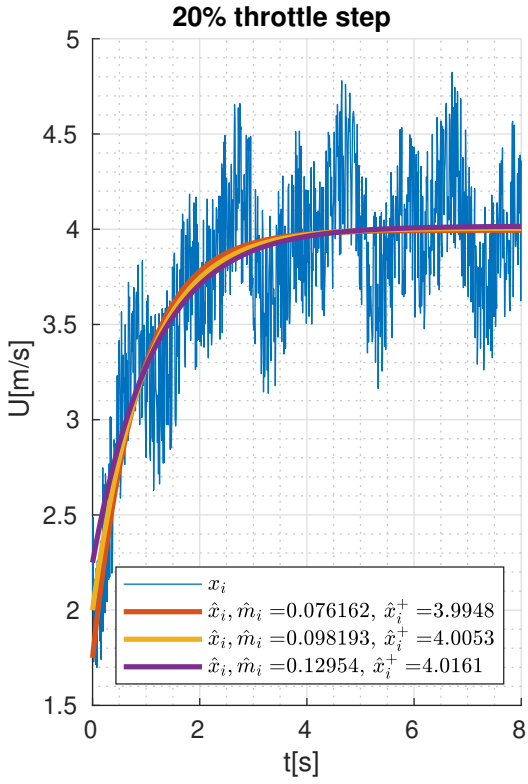
The solution utilized requires the data extraction to be performed first. Then, the results are used to calculate the boundaries in which the data should be confined to confirm its signal-to-noise ratio is large enough. As a means of calculating these boundaries, the height H of the signal is calculated as the absolute difference between the first measured state x_i^- and the estimated steady-state $\hat{x}_i^+$, as shown in Equation (3.13). The upper and lower limits ($x_{lim,up}$ and $x_{lim,low}$) for the criteria are defined in Equation (3.14) as the average between x_i^- and $\hat{x}_i^+$ plus height for upper limit and less height for lower limit.

$$H = |\hat{x}_i^+ - x_i^-| \tag{3.13}$$

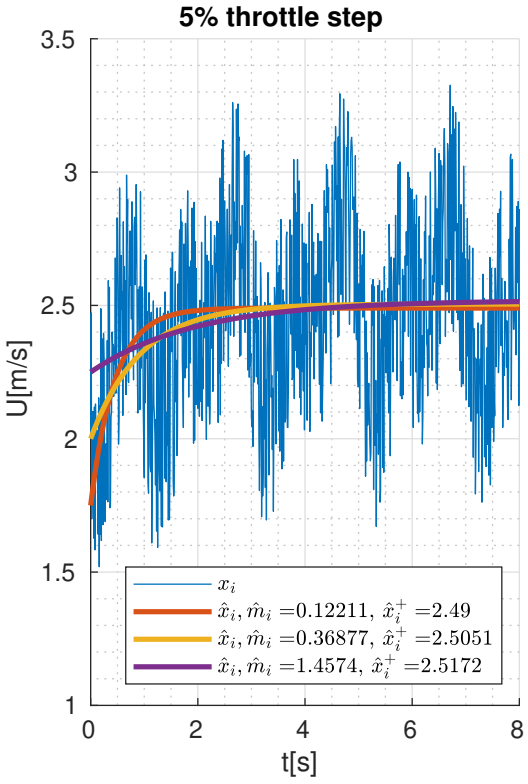
$$x_{lim,up} = \frac{\hat{x}_i^+ + x_i^-}{2} + H \tag{3.14a}$$

$$x_{lim,lo} = \frac{\hat{x}_i^+ + x_i^-}{2} - H \tag{3.14b}$$

Two examples are presented in Figure 3.4 with different throttle steps to demonstrate the impact of the boundaries and disturbances in each case. Figure 3.4a corresponds to a $20\,\%$ step in throttle and results in approximately $2\,m/s$ in height of the signal. Figure 3.4b corresponds to a $5\,\%$ step in throttle and results in approximately $0.5\,m/s$ in height of the signal. The noise is the same in both cases and has an amplitude of approximately $1.5\,m/s$. From visual inspection, it is possible to notice how hard it is to identify the step curve in Figure 3.4b due to the low signal-to-noise ratio. Applying the methodology, the $20\,\%$ step case easily fulfils the criteria, while Figure 3.5 shows the boundaries for the $5\,\%$ step case, which are trespassed by the data.



(a) Simulated surge speed for throttle step of $20\,\%$ and estimated surge speed with starting point at $1.75\,m/s$, $2\,m/s$ and $2.25\,m/s$.



(b) Simulated surge speed for throttle step of $5\,\%$ and estimated surge speed with starting point at $1.75\,m/s$, $2\,m/s$ and $2.25\,m/s$.

Figure 3.4: Examples to showing how the bigger signal-to-noise ratio in Figure 3.4b results in a bigger spread of estimated results when compared to Figure 3.4a with a lower signal-to-noise ratio.

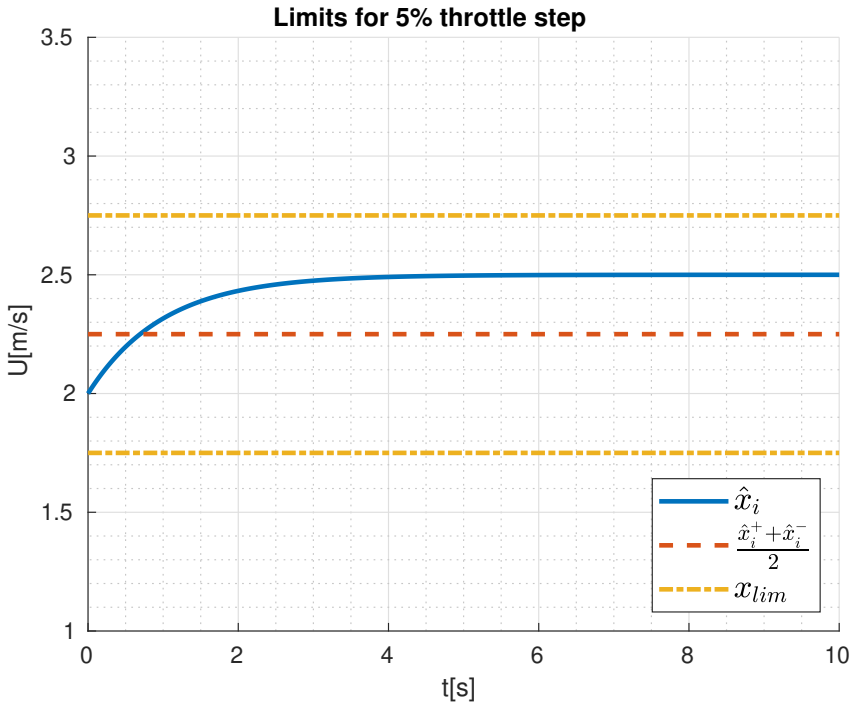


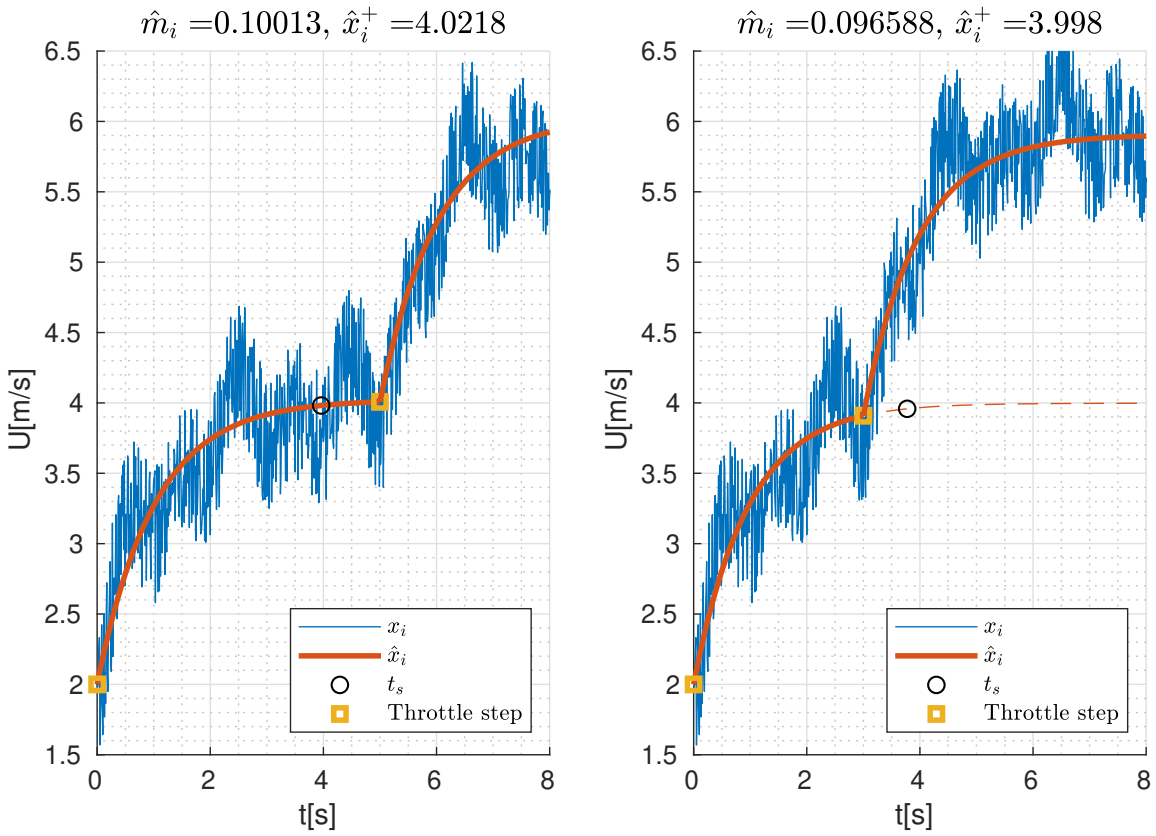
Figure 3.5: Calculated boundaries for estimated state $\hat{x}_i^+$ from Figure 3.4b starting at $2\,m/s$ with $5\,\%$ throttle step.

Both cases were created to serve as examples, so it is known that the initial state is $2\,m/s$, but initial states of $1.75\,m/s$ and $2.25\,m/s$ were also presented to show how a case with low signal-to-noise ratio has a wider spread in the results, especially for the inertia parameter $\hat{m}_i$. The spread in the first measured state x_i^- was considered when choosing it to calculate the height and the limits, because it can be an outlier, but not further explored.

3.4.2 Identification of Unreached Steady-State

This identification concerns only SIMID, since the RW-SSID is already embedded with SSID. This identification is performed just before building the data sets (Equation (3.12)), because it depends on the estimated parameters found by the optimization and subsequent calculations.

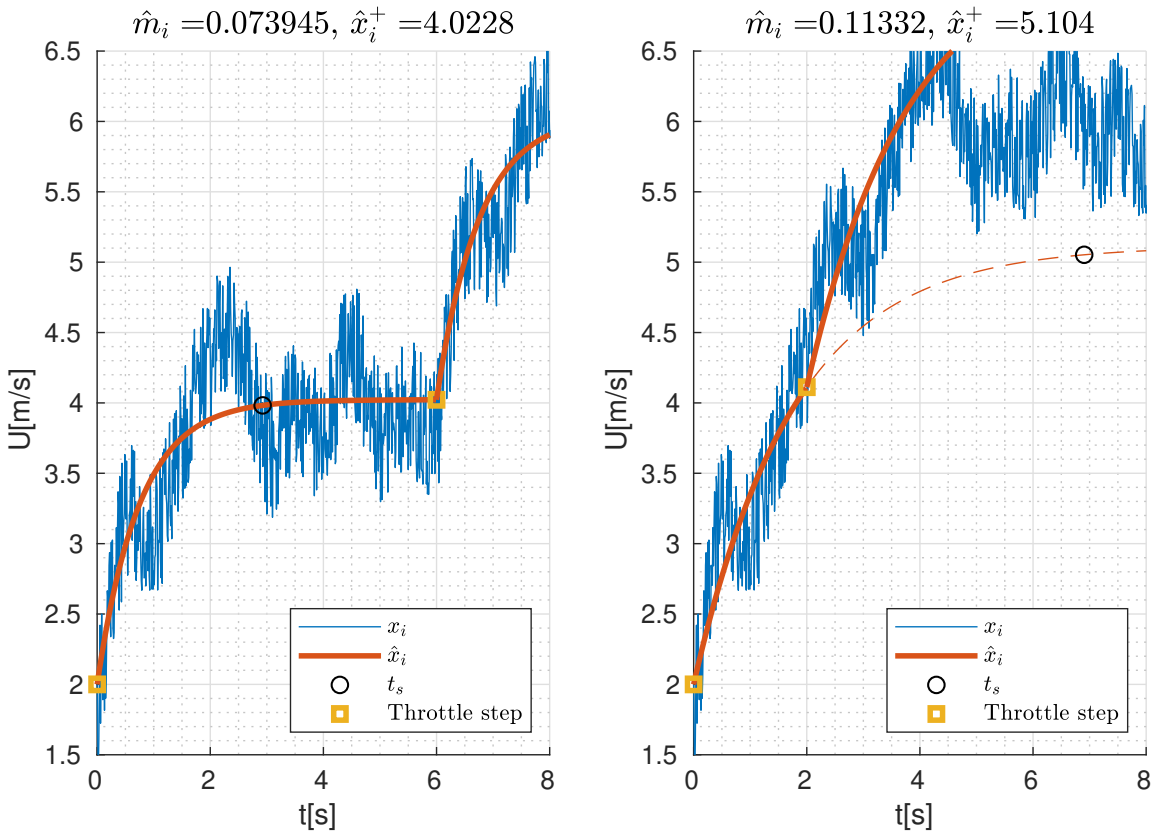
The criteria defined to judge whether steady-state is reached by the vessel is to check if the settling time t_s of the estimated state $\hat{x}_i$ happens before the next step command. Figures 3.6 and 3.7 show examples where the criteria is met (Figures 3.6a and 3.7a) and where it is not (Figures 3.6b and 3.7b). In these cases, only the examples where the criteria is met would be included in the data sets.



(a) Simulated surge speed for two subsequent throttle steps of 20 % at time 0 and 5 seconds and estimated surge speed with starting point at 2 m/s.

(b) Simulation of the same system as Figure 3.6a, but changing second throttle step to happen at 3 seconds.

Figure 3.6: Examples of settling time for estimated state before and close after next step command.



(a) Simulated surge speed for two subsequent throttle steps of 20 % at time 0 and 6 seconds and estimated surge speed with starting point at 2 m/s.

(b) Simulation of the same system as Figure 3.7a, but changing second throttle step to happen at 2 seconds.

Figure 3.7: Examples of settling time for estimated state before and far after next step command.

A situation of unreached steady-state is discarded as it may include in the data sets results with distortions in both $\hat{m}_i$ and $\hat{x}_i^+$. The dashed line in Figures 3.6b and 3.7b is a projection based on the available data, but not necessarily what would happen if the vessel was given enough time to reach steady-state. One assumption of the model structure used is that the states do not overshoot, but it does occasionally occur in practice. The pair of examples are of the same system being interrupted by a step at different moments and their respective results for the parameters. All steps are of 20 % in throttle.

It is possible to see how the impact of a premature subsequent step is much worse in Figure 3.7b than in Figure 3.6b, but it is only possible to affirm this because it is an artificially created example and the expected behaviour is known from Figures 3.6a and 3.7a. Therefore, results from Figures 3.6b and 3.7b would not be included in the data sets since they do not meet the criteria established. The necessity of this data cleansing method is minimized by performing the experiment with sufficient time between steps.

3.5 Reference Filter Saturation Limits

It is presented in Section 2.5 the benefits of having a reference filter in the control system solution. The algorithms presented can have their performance improved by properly tuning saturation limits, and the following sections describe how these saturation limits can be identified using the SI results.

3.5.1 Acceleration Saturation

Based on the model structure of control-oriented modelling, the acceleration saturation limits can be defined as a function of current and desired states for both surge and yaw. The first step is to modify Equation (2.12) to isolate the acceleration term, becoming Equation (3.16).

$$\boldsymbol{M}(\boldsymbol{x})\dot{\boldsymbol{x}} + \boldsymbol{\sigma}(\boldsymbol{x}) = \boldsymbol{\tau} \tag{2.12a}$$

$$\boldsymbol{M} = \begin{bmatrix} m_U(\boldsymbol{x}) & 0 \\ 0 & m_r(\boldsymbol{x}) \end{bmatrix}, \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_U(\boldsymbol{x}) \\ \sigma_r(\boldsymbol{x}) \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} U \\ r \end{bmatrix}, \quad \boldsymbol{\tau} = \begin{bmatrix} \tau_m \\ \tau_\delta \end{bmatrix} \tag{2.12b}$$

$$\dot{\boldsymbol{x}} = \frac{-\boldsymbol{\sigma}(\boldsymbol{x}) + \boldsymbol{\tau}}{\boldsymbol{M}(\boldsymbol{x})} \tag{3.16}$$

A control law defines the values for $\boldsymbol{\tau}$ based on current (x) and desired (x_d) states. A common algorithm for control laws is to combine a model-based feedforward component with a feedback component that corrects modelling imprecision. However, since this is a search for the saturation limits of the vessel, modelling imprecision is not so relevant, given that it is small, because the vessel would be capable of more acceleration to compensate for the imprecision.

The maximum possible acceleration should not be used as the saturation limit, because it can lead to unwanted and probably unsafe behaviour. For instance, it is not expected on USVs to command full throttle until reaching desired speed and then lower throttle to keep that speed. The normal procedure is to command throttle to the position of the final desired speed. Setting $\boldsymbol{\tau}$ using only the feedforward component matches this normal procedure and the experiment design, where step commands were used, minimizing errors in the model. Equation (3.17) presents this formulation for $\boldsymbol{\tau}$ and Equation (3.18) the consequence of using it in Equation (3.16).

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{x}_d)\dot{\boldsymbol{x}}_d + \boldsymbol{\sigma}(\boldsymbol{x}_d) \tag{3.17}$$

$$\dot{\boldsymbol{x}} = \frac{-\boldsymbol{\sigma}(\boldsymbol{x}) + \boldsymbol{M}(\boldsymbol{x}_d)\dot{\boldsymbol{x}}_d + \boldsymbol{\sigma}(\boldsymbol{x}_d)}{\boldsymbol{M}(\boldsymbol{x})} \tag{3.18}$$

Although not strictly necessary, the desired acceleration is removed from the equation, because none of the cases tested in this work include desired acceleration different than zero. This simplifies the notation from Equation (3.18) to Equation (3.19).

$$\dot{\boldsymbol{x}} = \frac{\boldsymbol{\sigma}(\boldsymbol{x}_d) - \boldsymbol{\sigma}(\boldsymbol{x})}{\boldsymbol{M}(\boldsymbol{x})} \tag{3.19}$$

Another simplification proposed is to fix the acceleration parameter during a change in desired state, just as the method for building the inertia data sets considers fixed inertia parameters during a step command. This simplifies implementation afterwards because otherwise some changes would need to be implemented in the example reference filters presented in Section 2.5, and it also enables a better comparison to SI results because of the similarity of the methods. The maximum absolute acceleration parameter ($\dot{\boldsymbol{x}}_{\max}$) is calculated as shown in Equation (3.20), the difference between damping for initial state $\boldsymbol{x}_{init}$ and desired state $\boldsymbol{x}_d$ divided by the inertia parameter relative to the mean of these states.

$$\dot{\boldsymbol{x}}_{\max} = \frac{\boldsymbol{\sigma}(\boldsymbol{x}_d) - \boldsymbol{\sigma}(\boldsymbol{x}_{init})}{\boldsymbol{M}(\frac{\boldsymbol{x}_d + \boldsymbol{x}_{init}}{2})} = \frac{\boldsymbol{\Delta\tau}}{\boldsymbol{M}(\frac{\boldsymbol{x}_d + \boldsymbol{x}_{init}}{2})} \tag{3.20}$$

For the purpose of comparing the results obtained for $\dot{\boldsymbol{x}}_{\max}$ with the data acquired from the tests, two new data sets are built in Equation (3.21), one for surge acceleration ($\mathcal{D}_{\dot{U}_{\max}}$) and another for yaw

acceleration ($\mathcal{D}_{\dot{r}_{\max}}$).

$$\mathcal{D}_{\dot{U}_{\max}} = \begin{bmatrix} \frac{\hat{U}^+_{1_U}+U^-_{1_U}}{2} & \frac{\hat{U}^+_{2_U}+U^-_{2_U}}{2} & \dots & \frac{\hat{U}^+_{n_U}+U^-_{n_U}}{2} \\ \frac{\hat{r}^+_{1_U}+r^-_{1_U}}{2} & \frac{\hat{r}^+_{2_U}+r^-_{2_U}}{2} & \dots & \frac{\hat{r}^+_{n_U}+r^-_{n_U}}{2} \\ \dot{U}_{\max,1_U} & \dot{U}_{\max,2_U} & \dots & \dot{U}_{\max,n_U} \\ \boldsymbol{\sigma}(\boldsymbol{x}^+_{1_U}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{1_U}) & \boldsymbol{\sigma}(\boldsymbol{x}^+_{2_U}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{2_U}) & \dots & \boldsymbol{\sigma}(\boldsymbol{x}^+_{n_U}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{n_U}) \end{bmatrix} \tag{3.21a}$$

$$\mathcal{D}_{\dot{r}_{\max}} = \begin{bmatrix} \frac{\hat{U}^+_{1_r}+U^-_{1_r}}{2} & \frac{\hat{U}^+_{2_r}+U^-_{2_r}}{2} & \dots & \frac{\hat{U}^+_{n_r}+U^-_{n_r}}{2} \\ \frac{\hat{r}^+_{1_r}+r^-_{1_r}}{2} & \frac{\hat{r}^+_{2_r}+r^-_{2_r}}{2} & \dots & \frac{\hat{r}^+_{n_r}+r^-_{n_r}}{2} \\ \dot{r}_{\max,1_r} & \dot{r}_{\max,2_r} & \dots & \dot{r}_{\max,n_r} \\ \boldsymbol{\sigma}(\boldsymbol{x}^+_{1_r}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{1_r}) & \boldsymbol{\sigma}(\boldsymbol{x}^+_{2_r}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{2_r}) & \dots & \boldsymbol{\sigma}(\boldsymbol{x}^+_{n_r}) - \boldsymbol{\sigma}(\boldsymbol{x}^-_{n_r}) \end{bmatrix} \tag{3.21b}$$

The first two rows are exactly the same as the inertia data sets. It uses the notation from SIMID as an example, since it is also valid for RW-SSID. The third row stores the maximum absolute acceleration extracted from each step command as shown in Equation (3.22) by deriving the estimated state $\hat{x}_i$ at $t = 0$. The final row registers the step in terms of damping to compare the results with proper equivalents in $\dot{\boldsymbol{x}}_{\max}$.

$$\hat{x}_i(t) = \frac{\Delta\tau_i}{k_i}\left(1 - e^{\frac{-k_i t}{m_i}}\right) + {x_i}^- \tag{2.23}$$

$$\dot{\hat{x}}_i(t) = \frac{\Delta\tau_i}{m_i} e^{\frac{-k_i t}{m_i}} \tag{3.22a}$$

$$\dot{x}_{\max,i} = \max_{t\geq 0} |\dot{\hat{x}}_i(t)| = |\dot{\hat{x}}_i(0)| = \frac{|\Delta\tau_i|}{m_i} e^{\frac{-k_i \cdot 0}{m_i}} = \frac{|\Delta\tau_i|}{m_i} \tag{3.22b}$$

3.5.2 Yaw Rate Saturation

Instead of controlling rate as the reference state, yaw is usually controlled through the heading or course over ground states. Hence, it is also useful to define the saturation limit for the yaw rate. It can be found as the steady-state yaw rate r as a function of surge speed U when the rudder angle command τ_δ is at its maximum.

This function could be defined from Equation (2.13) and using the SI results for $\boldsymbol{\beta}_{\boldsymbol{\sigma}_r}$, but it was decided to fit a model just to the data specific to this problem. It is simpler than using a parametric function, while also resulting in a better fit since only target data is used.

$$\sigma_r(\boldsymbol{x}) = \boldsymbol{\phi}_\sigma(\boldsymbol{x})\boldsymbol{\beta}_{\sigma_r} \tag{2.13c}$$

$$\boldsymbol{\phi}_\sigma(\boldsymbol{x}) = [1, U, r, U^2, Ur, r^2, U^3, U^2 r, Ur^2, r^3, U^4, U^3 r, U^2 r^2, Ur^3, r^4] \tag{2.13d}$$

Figure 3.8 shows all data acquired by Eriksen and Breivik (2017), also used in this work, and highlights the data applicable to defining the yaw rate saturation. The model used to fit the data is a fourth order polynomial to keep the pattern of the overall model structure and the procedure to fit also includes regularization and cross-validation as presented in Section 2.4.4 to avoid overfitting.

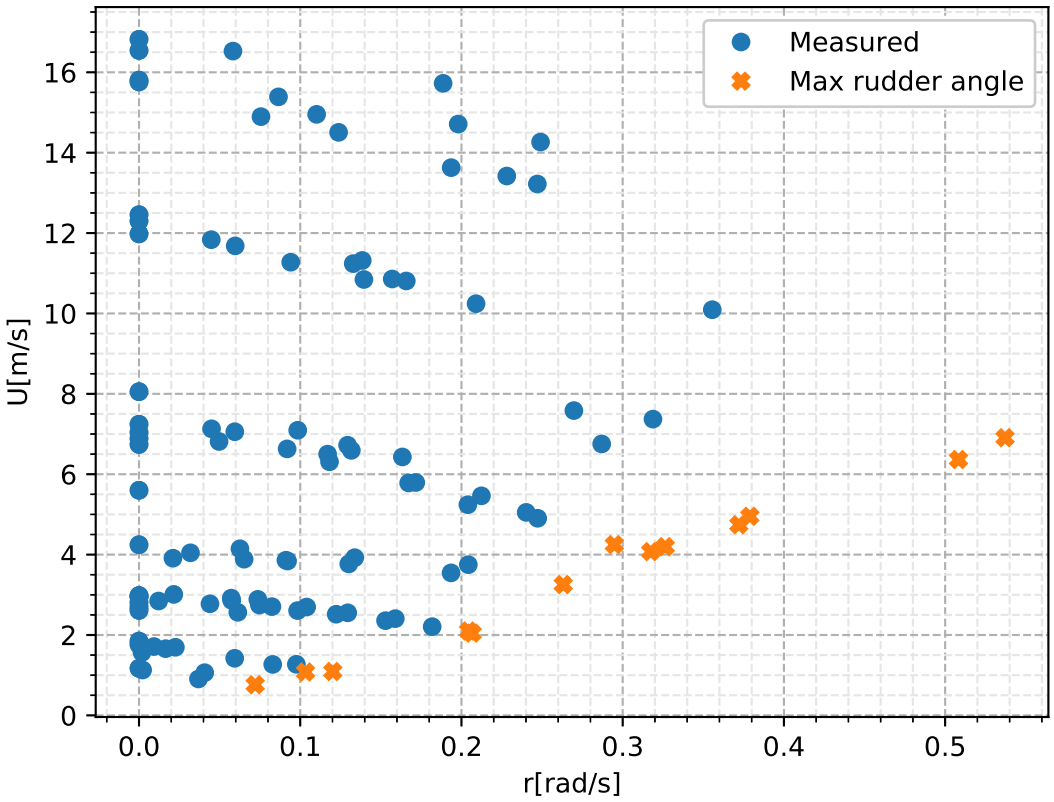


Figure 3.8: Steady-state results for surge speed U and yaw rate r highlighting the results achieved while at maximum rudder angle.

The point in Figure 3.8 with the biggest yaw rate r represents the limit of the combination of maximum throttle and the maximum rudder angle, so the USV is not capable of a bigger r. Therefore, the fitted polynomial used to define the yaw rate saturation must be truncated at this biggest yaw rate.

Chapter 4

Results and Discussion

The results obtained are presented in the same sequence for SI as presented in Figure 2.4 and followed by reference filter saturation limits. The discussion for each result is done right after its presentation and a final section of general remarks is included.

4.1 Experiment Design, Data Collection and Model Structure Selection

As explained in Chapter 3, these stages of SI were not performed in this work, but taken from the reference literature as the basis for testing new methods for SI of control-oriented models. The experiment and data collection procedures are explained in Section 2.4.1 and the model structure is detailed in Section 2.3. Overview plots of the collected data are presented in Figures 4.1 and 4.2.

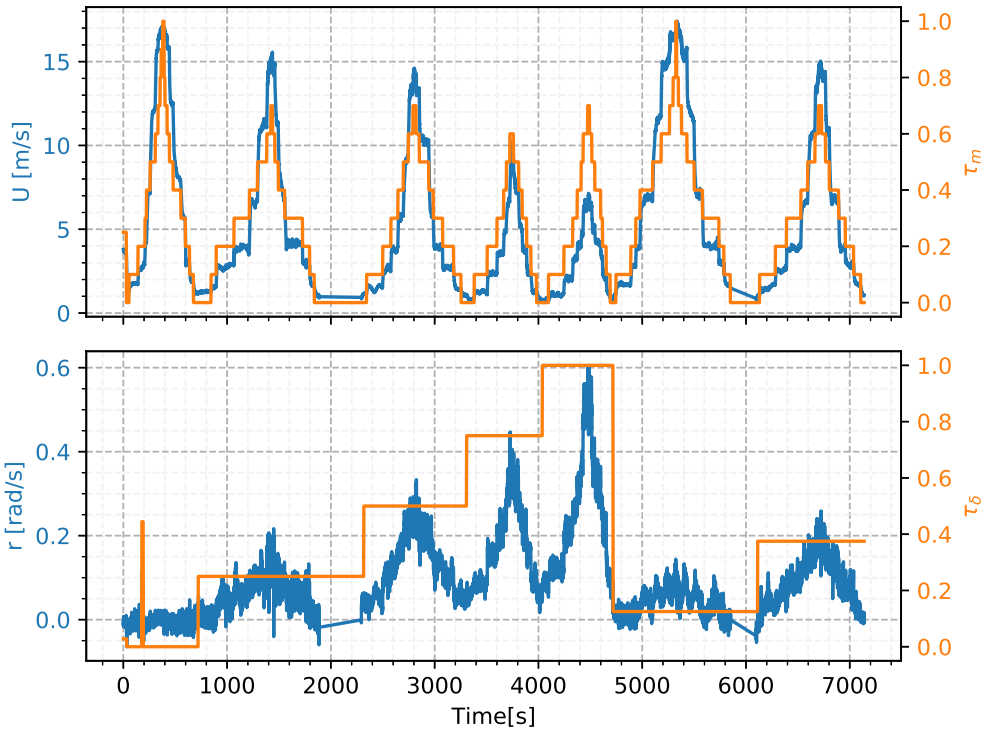


Figure 4.1: Data collected by Eriksen and Breivik (2017) during sequences of constant rudder angle for changing throttle.

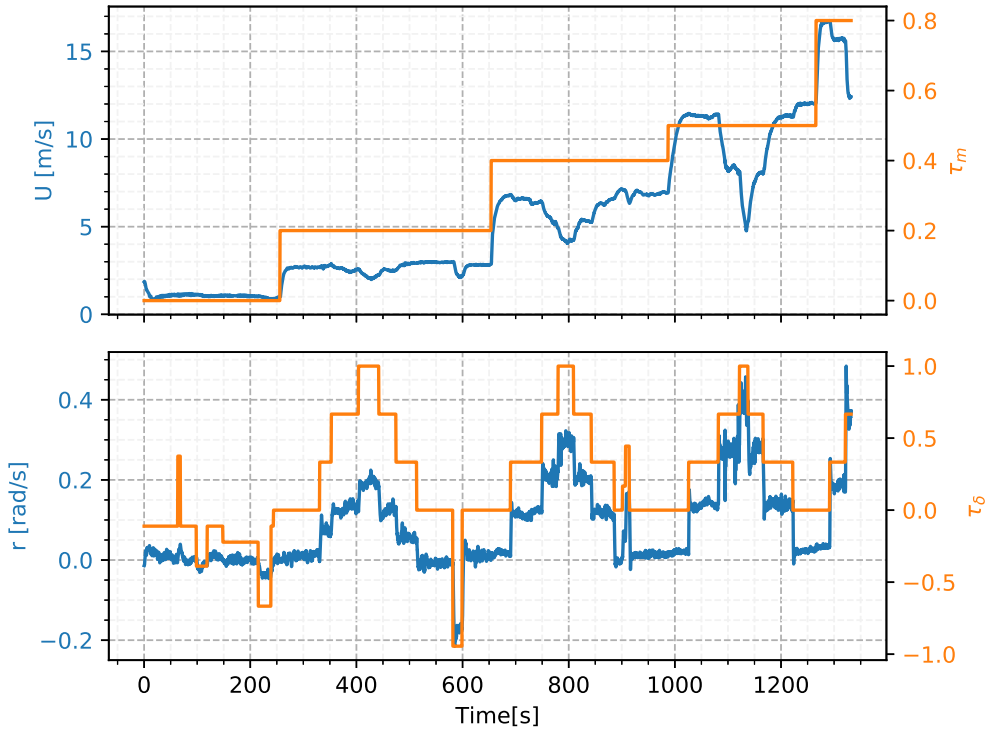


Figure 4.2: Data collected by Eriksen and Breivik (2017) during sequences of constant throttle for changing rudder angle.

Although these stages of SI were not performed in this work, there is one topic to discuss from the results. A noticeable outcome from the data collection plots (Figures 4.1 and 4.2) not discussed by Eriksen and Breivik (2017) and Kvalvaag (2018) is that a zero throttle command ($\tau_m = 0$) still outputs speed to the USV. This is due to normalizing the throttle command inside the geared range. A combustion outboard engine has a deadband in the actual throttle in which the gear is neutral, returning no propulsion. The gear clutches in as soon as the throttle deadband is passed, outputting the engine's minimum RPM and consequently the USV's minimum controllable speed. The relation between normalized and actual throttle was handled by Breivik et al. (2008) as seen in Equation (2.11b). It was not handled in this work because experimental tests using the model as the feedforward part of a controller were not performed.

4.2 Data Extraction

The data extraction stage is performed for every step command, but the detailed results are presented only for a few example steps in order to avoid an overflow of the text. Both new extraction methods are applied and their overall results presented.

4.2.1 Rolling Window Steady-State Identification (RW-SSID)

This data extraction method delivered the example results in Figures 4.3 to 4.5 after tuning of the size of the rolling window n_t to 10 seconds and the significance level α to 1%. While Figures 4.3 and 4.4 show a nice fit to the data, Figure 4.5 shows a premature steady-state identification. This can also be seen by using the same data from Figure 4.5, but with a rolling window size extended to 12 seconds, as shown in Figure 4.6.

Such discrepancies showed the necessity of tuning the size of the rolling window according to each case instead of one value for all as initially expected. This would make the near full automation of SI

impracticable, so it was decided not to continue with this method further in the SI procedure. This is further discussed in Section 4.2.3 when comparing to SIMID.

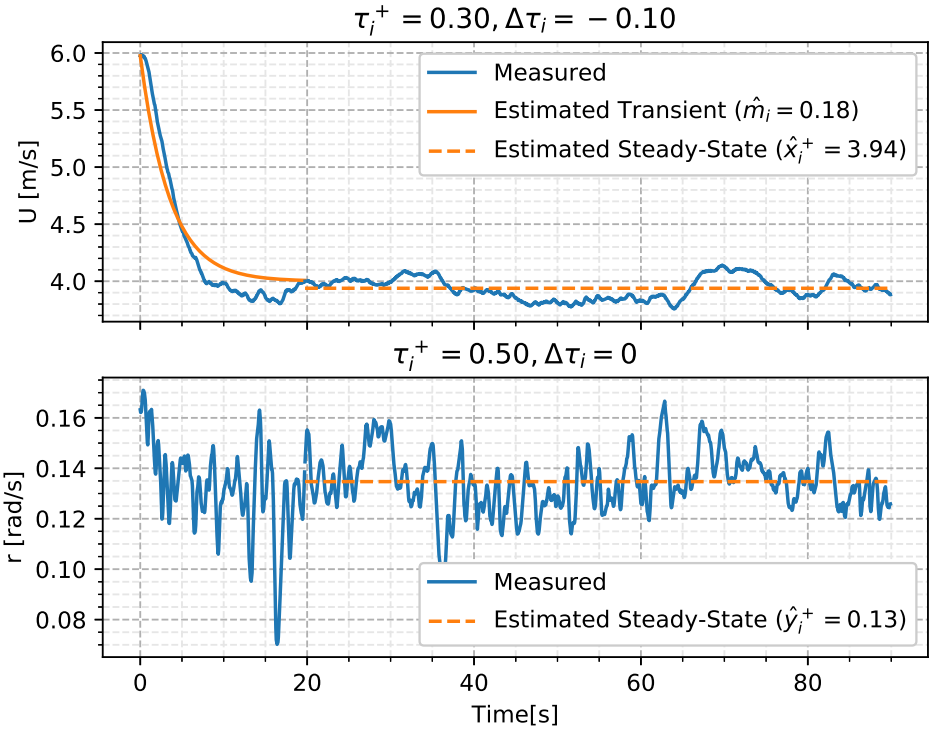


Figure 4.3: Example of throttle step and its extracted results with RW-SSID ($\alpha = 1\%, n_t = 10s$).

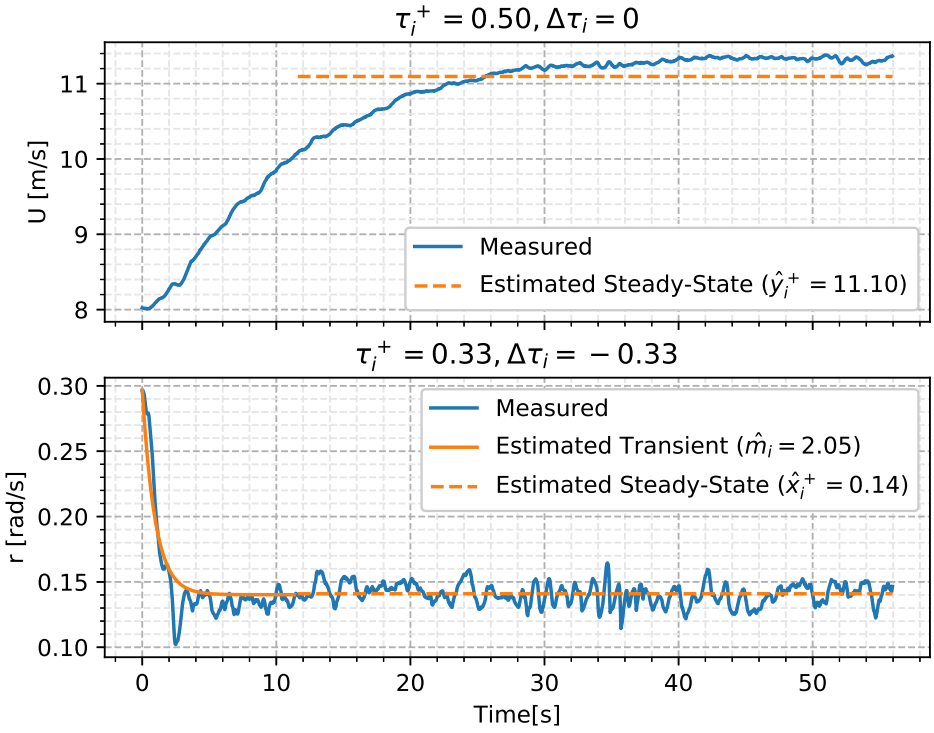


Figure 4.4: Example of rudder angle step and its extracted results with RW-SSID ($\alpha = 1\%, n_t = 10s$).

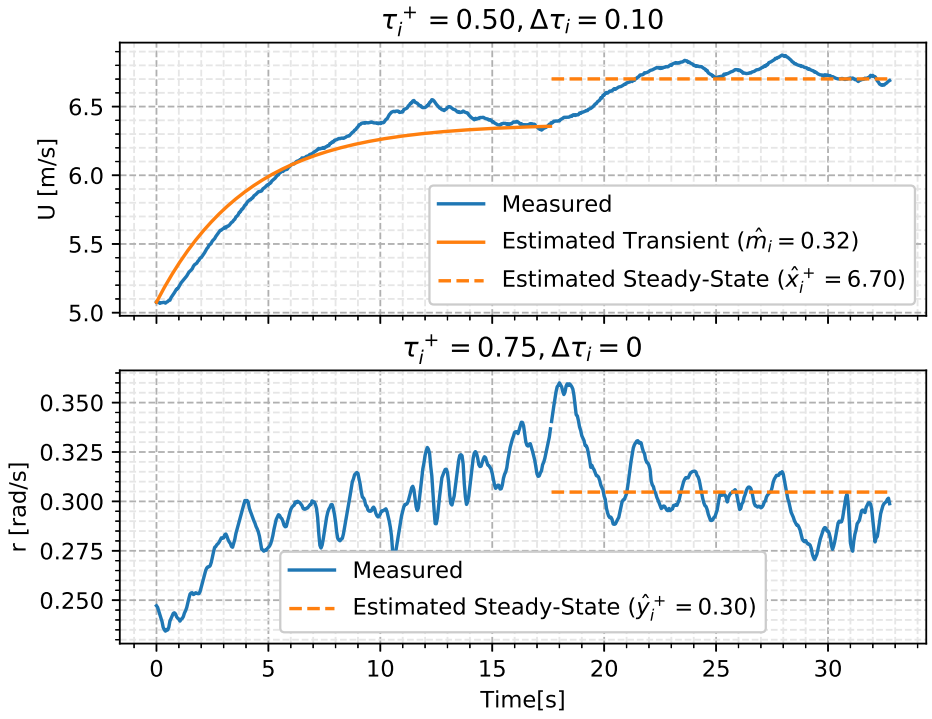


Figure 4.5: Example of throttle step and its extracted results with RW-SSID ($\alpha = 1\%, n_t = 10s$).

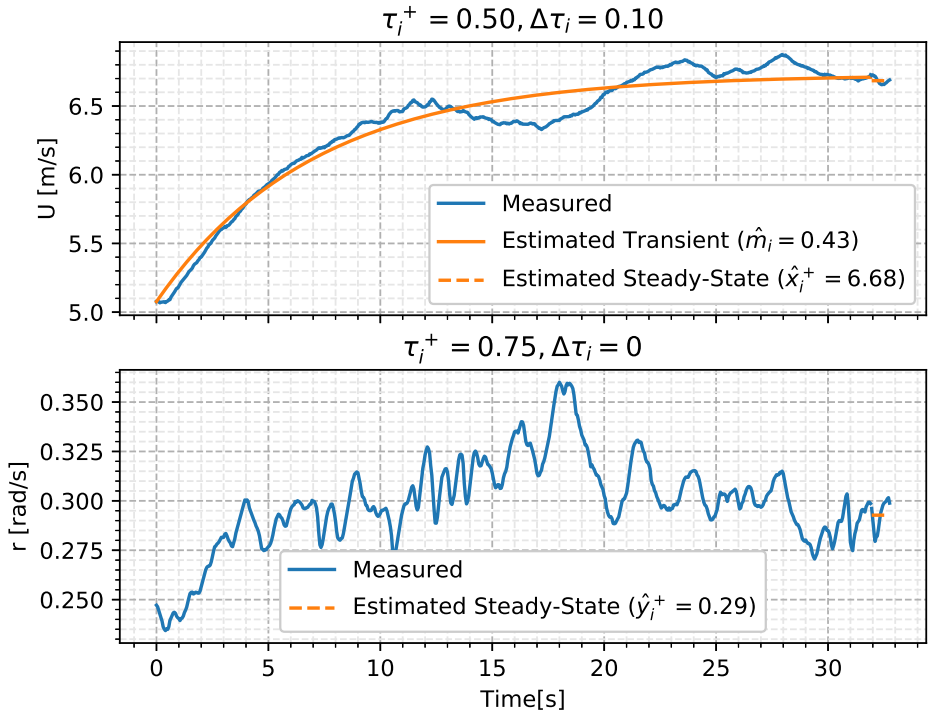


Figure 4.6: Example of throttle step, the same as Figure 4.5, and its extracted results with RW-SSID ($\alpha = 1\%, n_t = 12s$). Compared to Figure 4.5, the distinction between transient and steady-state regions is found at 32 seconds instead of 17.5 seconds, exemplifying the problem found with tuning the rolling window length n_t.

4.2.2 Simultaneous Identification (SIMID)

Figures 4.7 to 4.9 present the results of applying SIMID to the same three steps used to exemplify RW-SSID results.

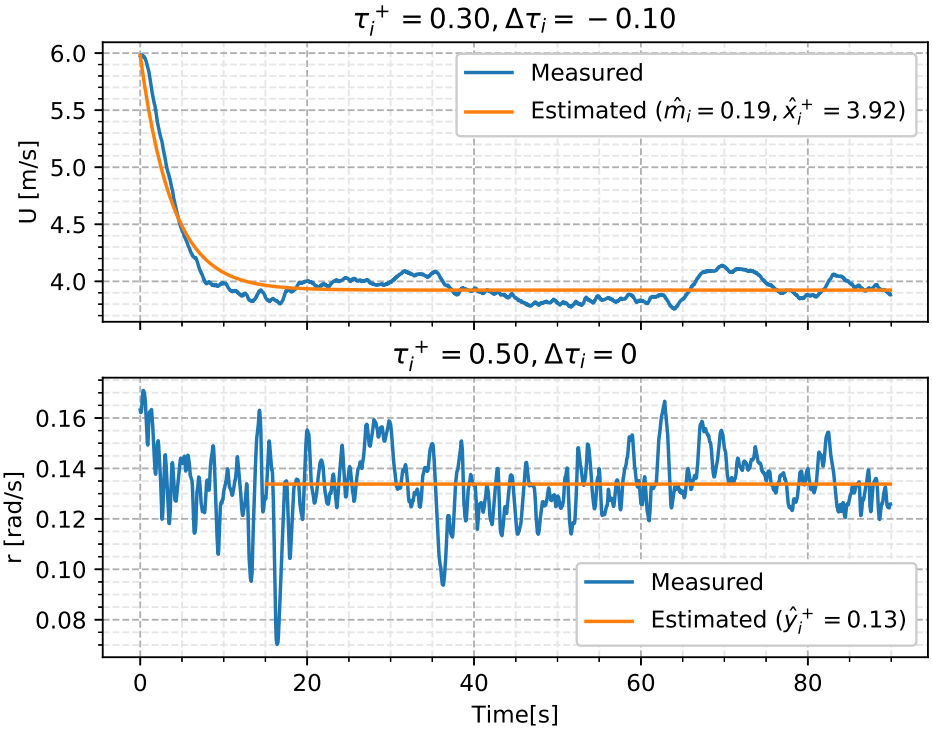


Figure 4.7: Example of throttle step, the same as Figure 4.3, and its extracted results with SIMID.

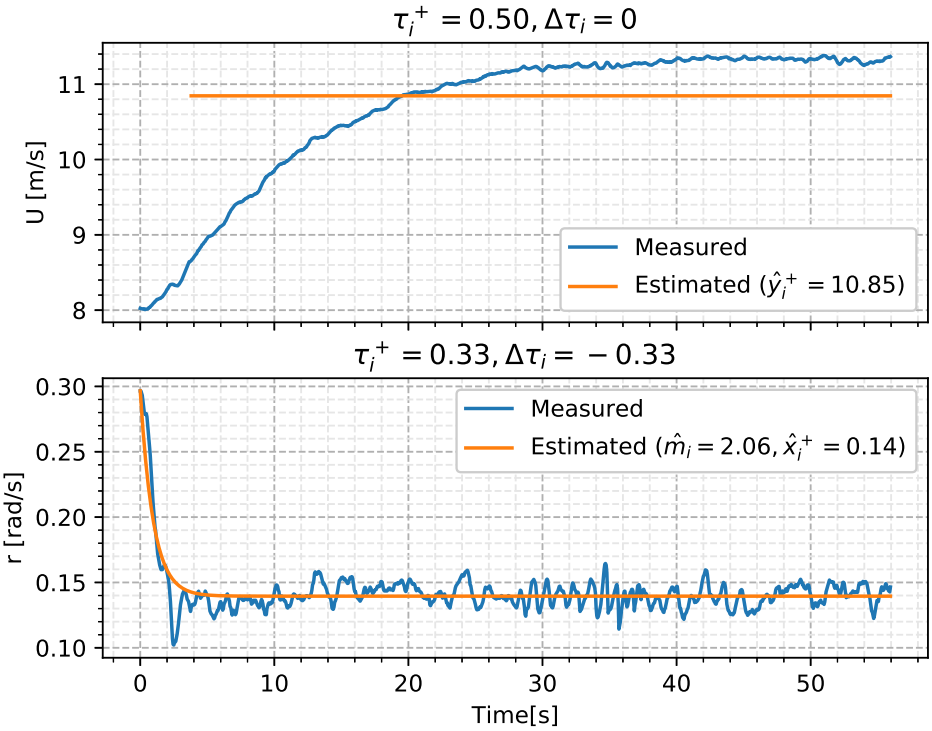


Figure 4.8: Example of rudder angle step, the same as Figure 4.4, and its extracted results with SIMID.

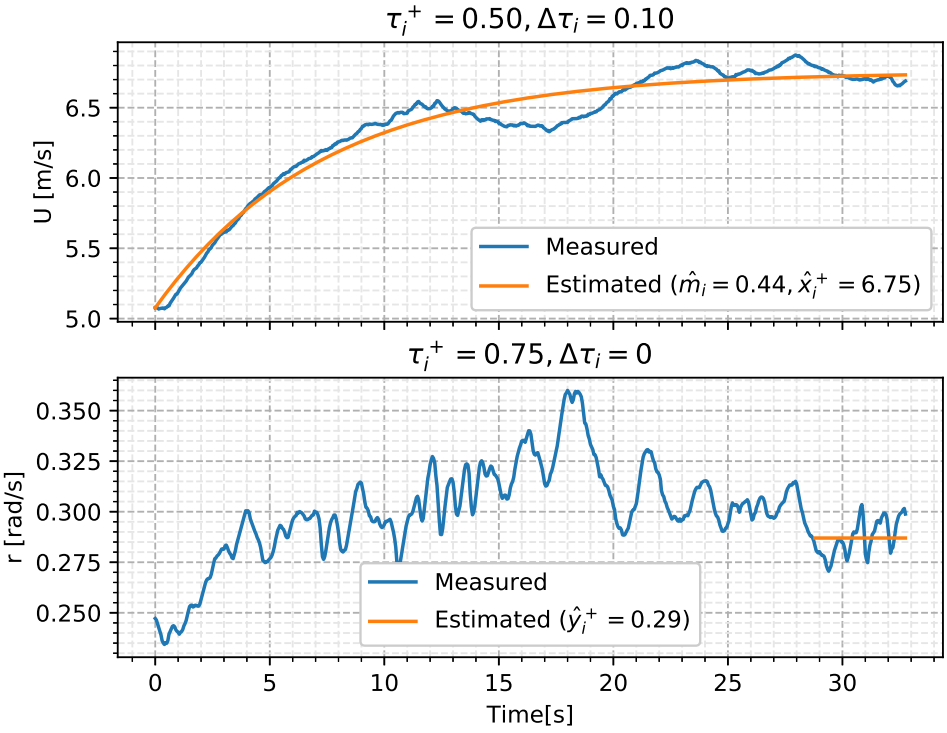


Figure 4.9: Example of throttle step, the same as Figure 4.5, and its extracted results with SIMID.

The previous examples were considered valid to be included in the data sets, but some other steps were not. Figure 4.10 presents a case in which steady-state was not reached by surge speed after a throttle command and Figure 4.11 presents a case in which signal-to-noise ratio is too low, causing the measured data to cross established validation boundaries as defined in Section 3.4.1. Both cases were automatically identified and marked invalid using the methods in Section 3.4.

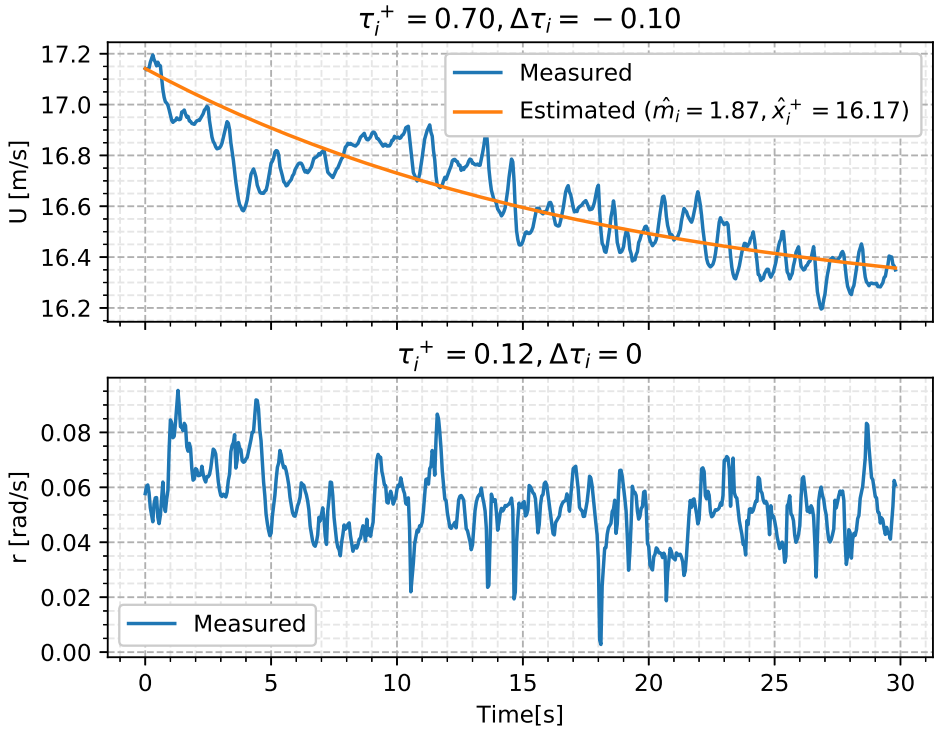


Figure 4.10: Example of throttle step and its extracted results with SIMID, invalid due to not reaching steady-sate.

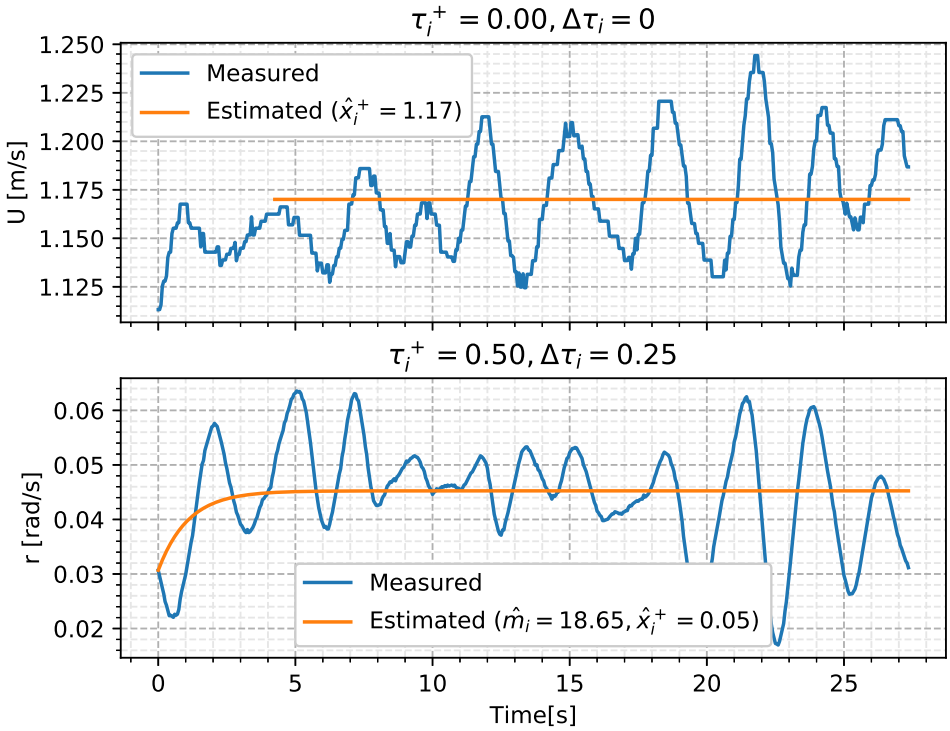


Figure 4.11: Example of rudder angle step and its extracted results with SIMID, invalid due to the signal-to-noise ratio being too low.

The overview of all data collected as presented in Figures 4.1 and 4.2, but including the distinction of valid and invalid data after applying the methods for removal of unreliable data, is presented in Figures 4.12 and 4.13.

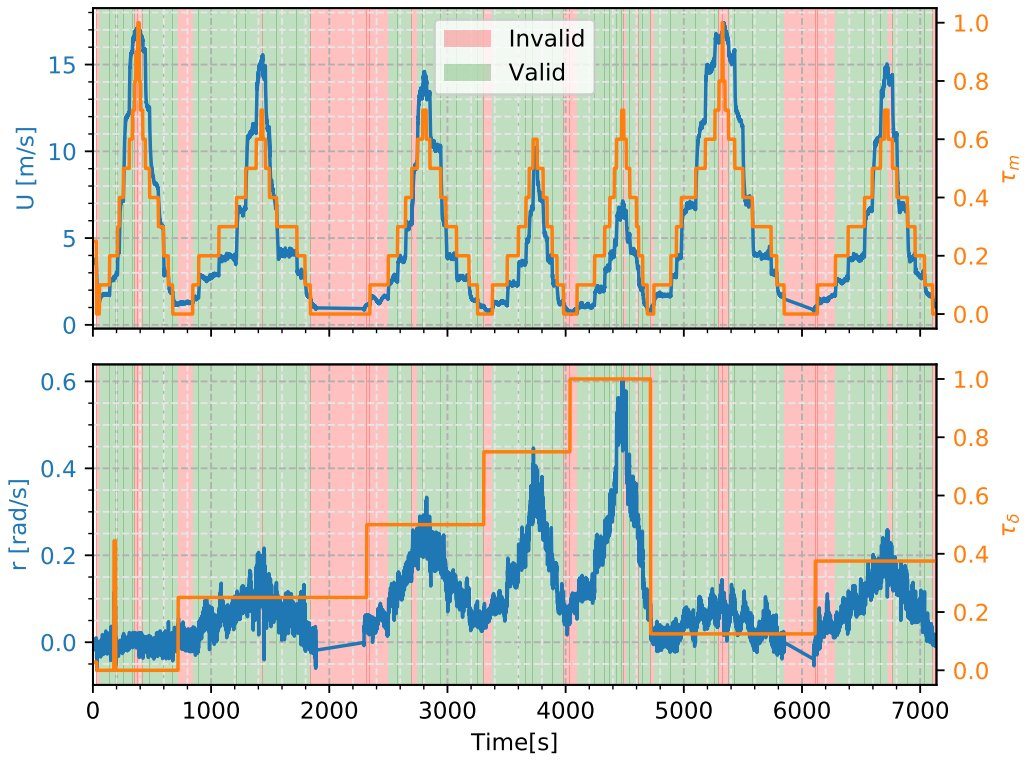


Figure 4.12: Distinction between valid and invalid data after applying SIMID to data from Figure 4.1.

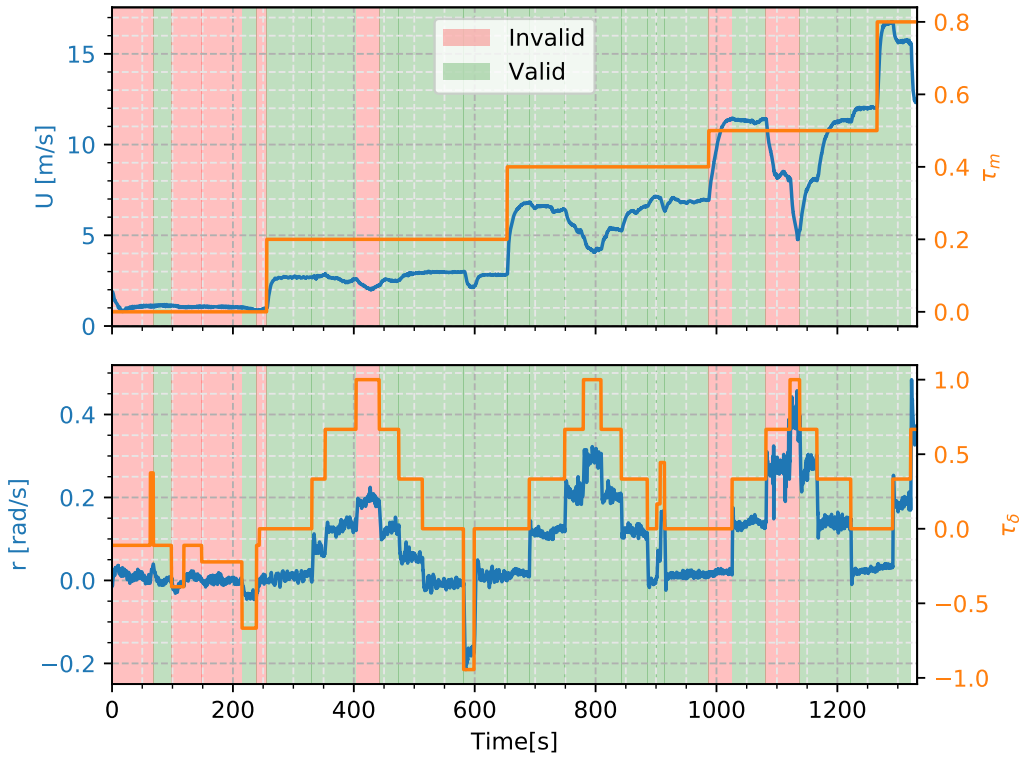


Figure 4.13: Distinction between valid and invalid data after applying SIMID to data from Figure 4.2.

A tendency noted in Figures 4.1 and 4.2 is that most invalid throttle steps were removed due to surge speed not reaching steady-state, while most invalid rudder angle steps were removed due to a low signal-to-noise ratio. Despite the removal of some part of the acquired data, the results in the next sections show how the valid data was enough to cover the entire operational range of the USV and produce good results.

The final data sets extracted after using SIMID are presented in Figure 4.14. The four subfigures include extracted damping and inertia parameters for surge speed and yaw rate as a function of both states. The color map from purple, at the lowest values, to yellow, at the highest, follow the vertical axis in all plots to improve the 3D visualization of the data.

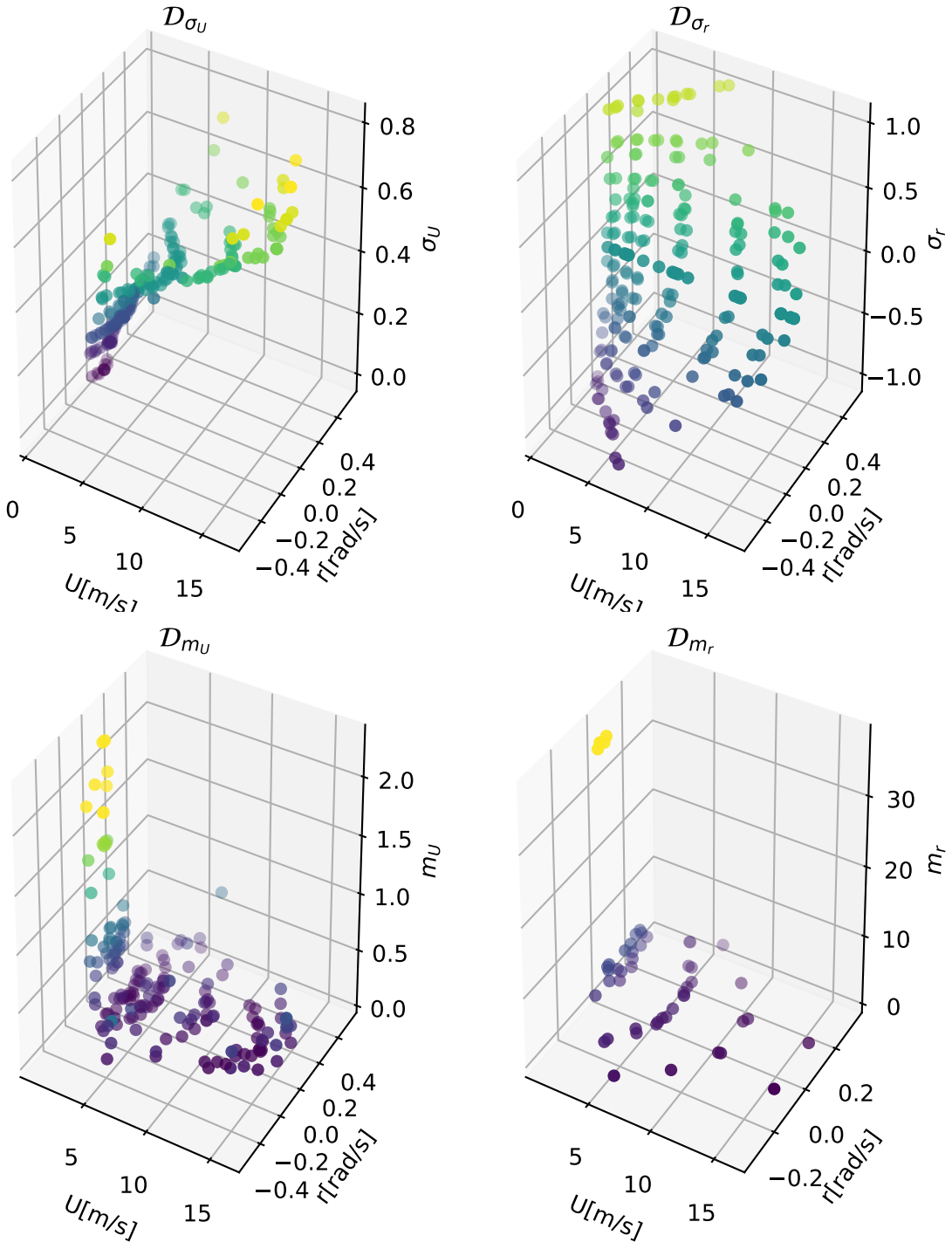


Figure 4.14: Damping and inertia data sets for surge speed and yaw rate extracted using SIMID.

4.2.3 Comparison of Methods

The fact that the rolling window would need individual tuning for each step in command positioned this method far away from the objective of full automation, which was achieved by SIMID. Hence, RW-SSID was considered impractical for data extraction, and its results were not used further in the SI procedure.

Aiming to understand why RW-SSID performed poorly, the article that served as its basis (Dalheim

and Steen, 2020) was revisited and some explanations were induced. Dalheim and Steen (2020) applied their steady-state identification method to data collected with many minutes of spacing between steps, while the data for this work has an average of 60 seconds between steps. Any system identification method can have better results when more data is available, but in this case such small time period between steps made RW-SSID impracticable. Longer time periods would open up the possibility of longer rolling windows, which would prevent premature identification, as shown in Figure 4.5. It makes it easier to identify if a disturbance in data during steady-state really is only a disturbance or if the vessel is still in transient dynamics. However, minimizing time is a constraint of the problem and a time period of many minutes would take too much time to perform all steps designed for the experimental trials. Therefore, RW-SSID was considered not fit to this problem and its development was halted.

On the other hand, SIMID performed up to expectations. Data extraction was fully automated with this method and the results were consistent when analysed individually as exemplified in Figures 4.7 to 4.9. Such automation was also made possible by using the methods for automatic removal of data considered unreliable. A rigorous analysis of removed data show that some parts of it could still have been used, for instance the steady-state results in Figure 4.11. However, a generalized rule to include such cases could not be created and this is a requirement for the automation to work properly, so valuable data was intentionally left out of the SI procedure. The key reason for this decision is that including faulty data is worse than leaving some good data out, because faulty data deviates the model results and, although it would be ideal to use as much good data as possible, leaving some out causes no clear impact on the results unless the amount of extracted data is very small, which is not the case.

Another positive indication of the success of the automation of data extraction by SIMID is that the final data sets extracted are very similar to the data sets extracted by Eriksen and Breivik (2017) and Kvalvaag (2018). Even so there is still room for improvement. It can be noted in Figure 4.8 that after the rudder angle step change the surge speed U takes much longer to converge than the yaw rate r. Since it is a rudder angle change, the assumption of the model is that U is constant, which is clearly not the case with this coupled result, causing an offset in the result. Despite this problem, model validation shows good results, but could possibly be improved if following these suggestions that impact data extraction:

- **Longer periods between steps:** Not too long since time is a constraint as previously discussed, but a longer overall period would mean a longer period of steady-state. Therefore, the offset as seen in Figure 4.8 would be minimized as more data in U steady-state would be used for averaging.
- **Identification of coupled motion:** This would mean dropping the assumption of decoupled inertia, transforming the inertia matrix from Equation (2.12b) to Equation (4.1) and consequently requiring additional inertia data sets.

$$\boldsymbol{M} = \begin{bmatrix} m_{UU}(\boldsymbol{x}) & m_{Ur}(\boldsymbol{x}) \\ m_{rU}(\boldsymbol{x}) & m_{rr}(\boldsymbol{x}) \end{bmatrix} \tag{4.1}$$

4.3 Parameter Identification

The parameters for the control-oriented model were identified using data sets presented in Figure 4.14. The results can be represented by generating surfaces with the identified parameters as shown in Figure 4.15.

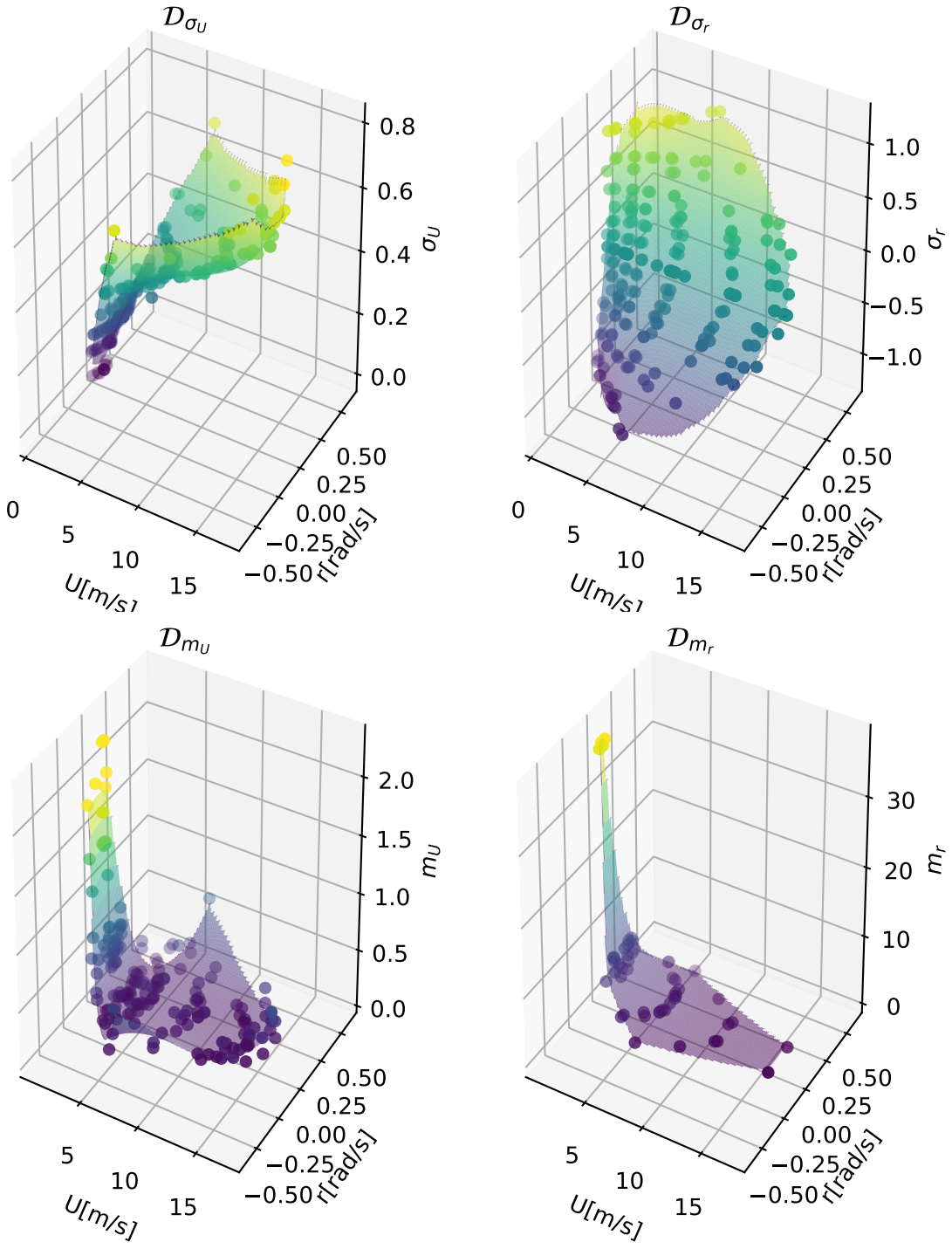


Figure 4.15: Damping and inertia data sets extracted using SIMID and respective surfaces from parameter identification results cropped on convex area around points.

It is important to highlight that the surfaces are plotted limited to the smallest convex area around the data points and this is the area where the model is valid. A comparison is plot in Figure 4.16 using the same model identified and same color map range, but showing the surface for a rectangular area between minimum and maximum surge speed and yaw rate.

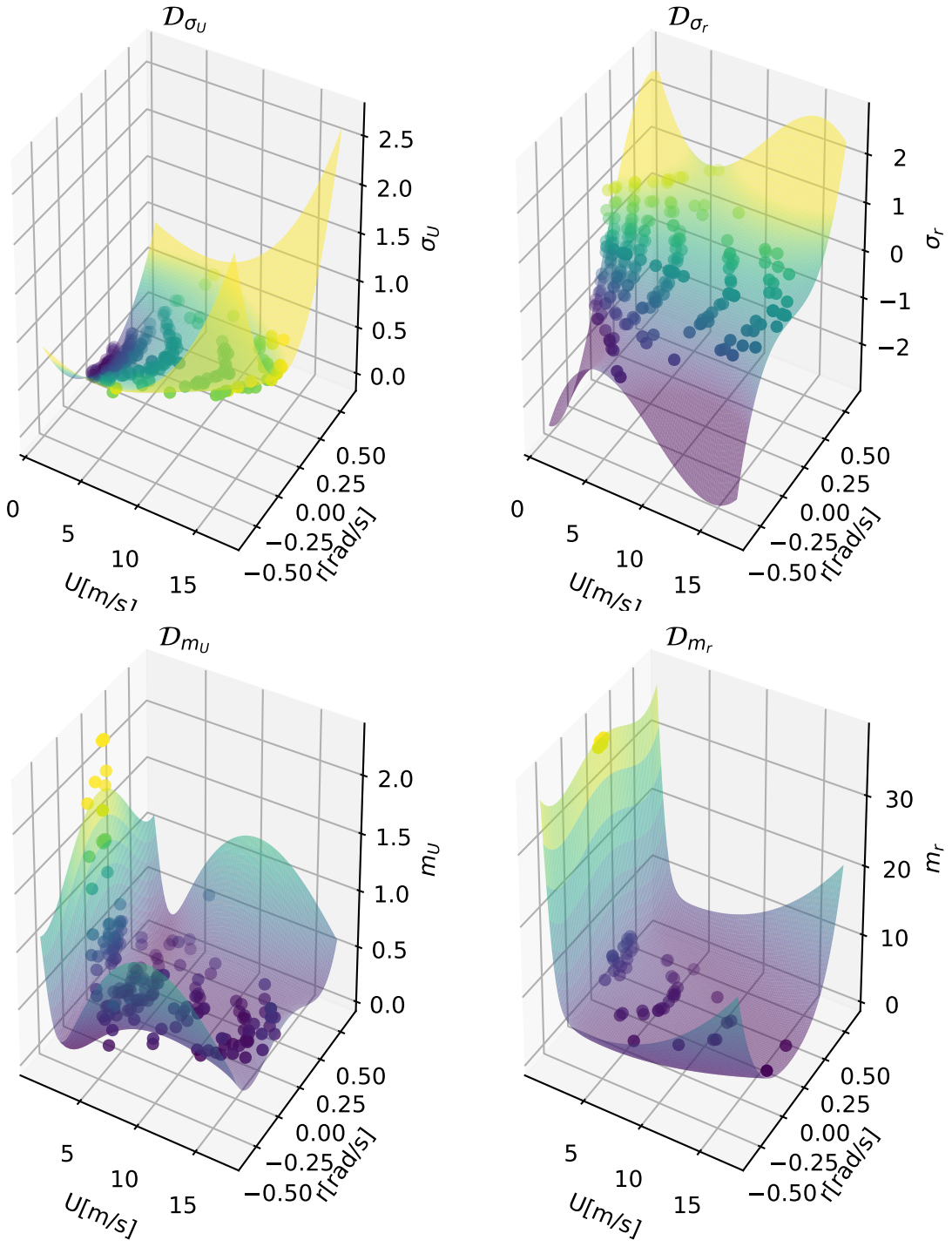


Figure 4.16: Damping and inertia data sets extracted using SIMID and respective surfaces from parameter identification results plotted between minimum and maximum surge speed and yaw rate.

In order to provide a better comparison for the problems of using extrapolation for the models, parameter identification was performed for only half of the data set points, randomly selected. Results are shown in Figure 4.17 with the surface constrained to the convex area covered by the points and in Figure 4.18 with the surface limited to minimum and maximum surge speed and yaw rate.

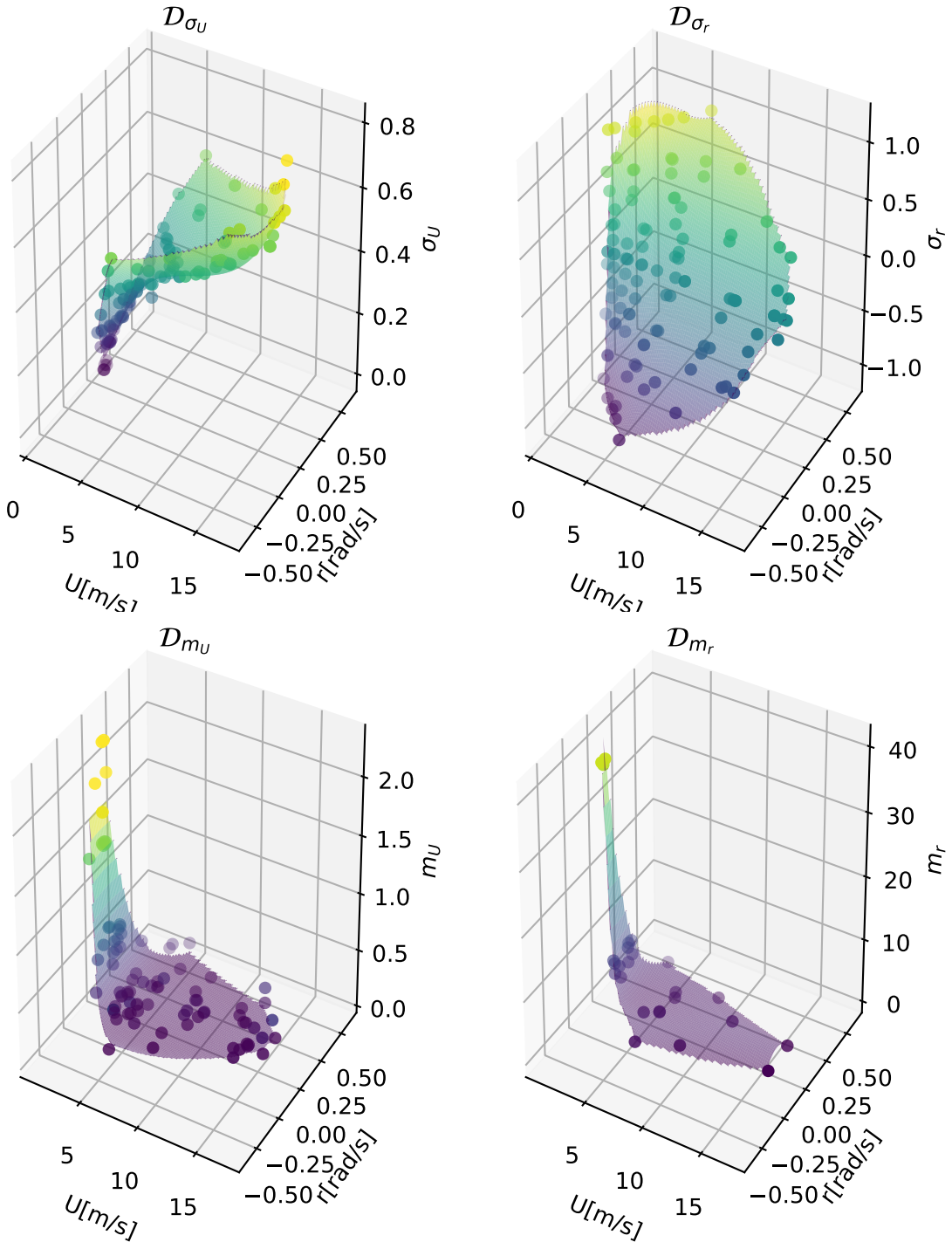


Figure 4.17: Half of damping and inertia data sets extracted using SIMID and respective surfaces from parameter identification results cropped on convex area around points.

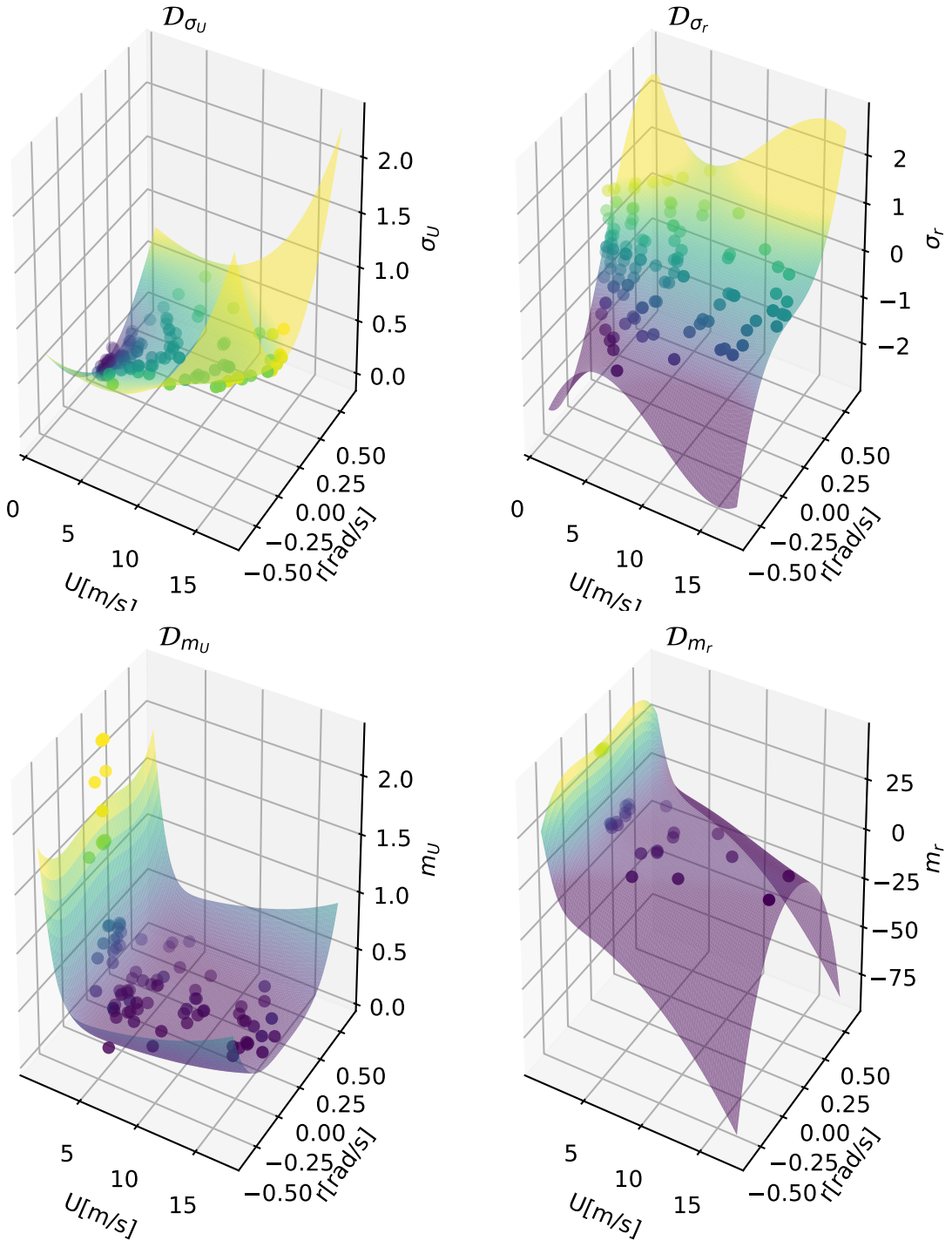


Figure 4.18: Half of damping and inertia data sets extracted using SIMID and respective surfaces from parameter identification results plotted between minimum and maximum surge speed and yaw rate.

The surfaces representing the model with identified parameters showed a good fit to data set points in Figure 4.15 and similar results to the reference articles that used the same acquired data. Even Figure 4.17 only using half of the extracted data also returned a similar result.

On the other hand, it can be seen by comparing Figures 4.16 and 4.18 how the extrapolation of these results cannot be trusted. Their region outside convex area around points is completely different for

the inertia data sets ($\mathcal{D}_{m_U}$ and $\mathcal{D}_{m_r}$) in this example, but could be for any of the data sets depending on the randomly selected points to create the comparison. Moreover, another example of inadequacy of extrapolation is that, in Figure 4.18, the yaw rate inertia m_r becomes negative in the extrapolation, which could never be realistic because it would make the model represent an unreal unstable dynamic.

These results also reinforce how essential it is that the experiment covers the entire operational range of the USV. Otherwise, the model could be distorted inside the operational range, causing problems when using the model as the feedforward component for a controller.

One of the causes for the results from parameter identification only being valid within the convex area around points is the selected model structure. The control-oriented model is a non-first principles model for simplification of identification and the incapacity of use for extrapolation is a disadvantageous consequence of its simplifications. Nevertheless, this problem can be mitigated by acquiring data for all operational limits of the USV as was done. Hence, the model structure did not spoil results, on the contrary it made it possible to fully automate the SI process and no iteration on model structure was necessary.

4.4 Model Validation

The trial for validation is the same as used by Eriksen and Breivik (2017) and its commands for throttle and rudder angle are presented in Figure 4.19. The results achieved by this work are presented in Figure 4.20 and were considered satisfactory enough not to need an iteration in SI procedure. Validation results by Eriksen and Breivik (2017) are shown in Figure 4.21 for comparison on how the results from this thesis were able to achieve the same model validation quality, although fully automated.

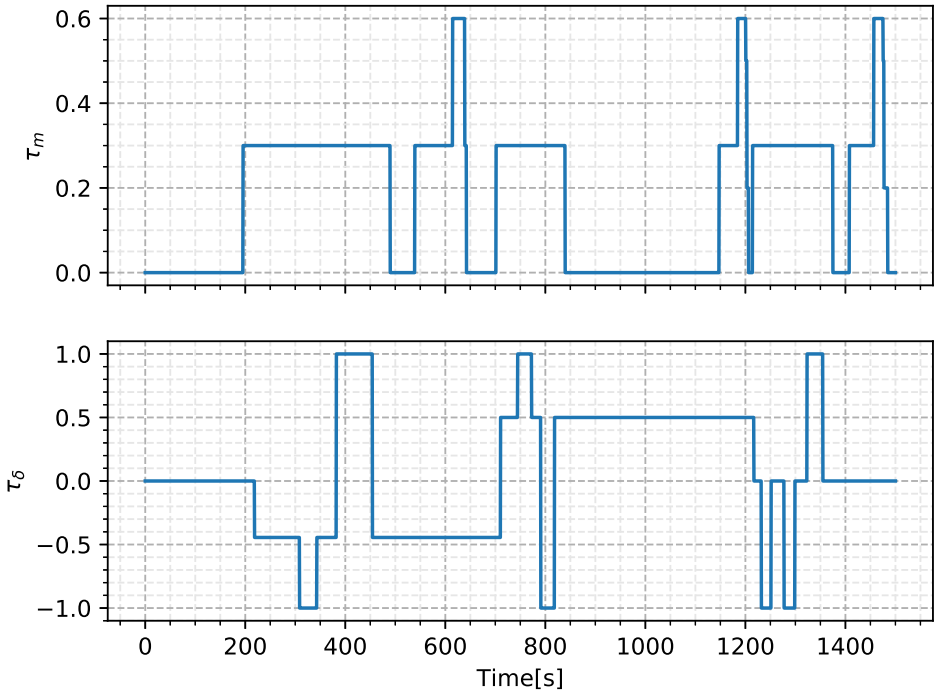


Figure 4.19: Sequence of throttle and rudder angle commands of trial for model validation.

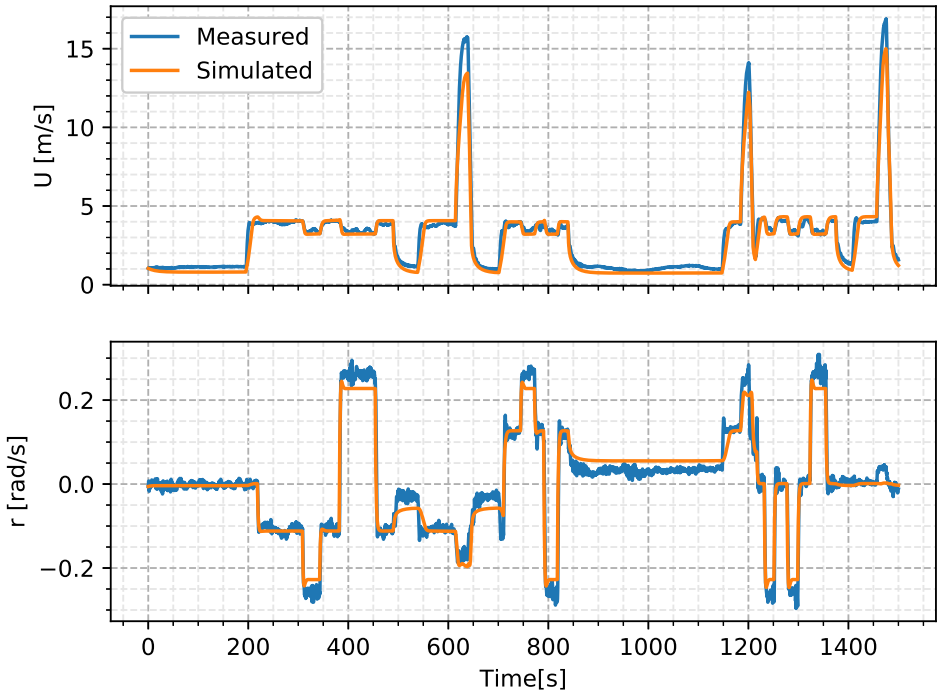


Figure 4.20: Model validation results for this work by comparing simulated and measured data of trial not used for SI.

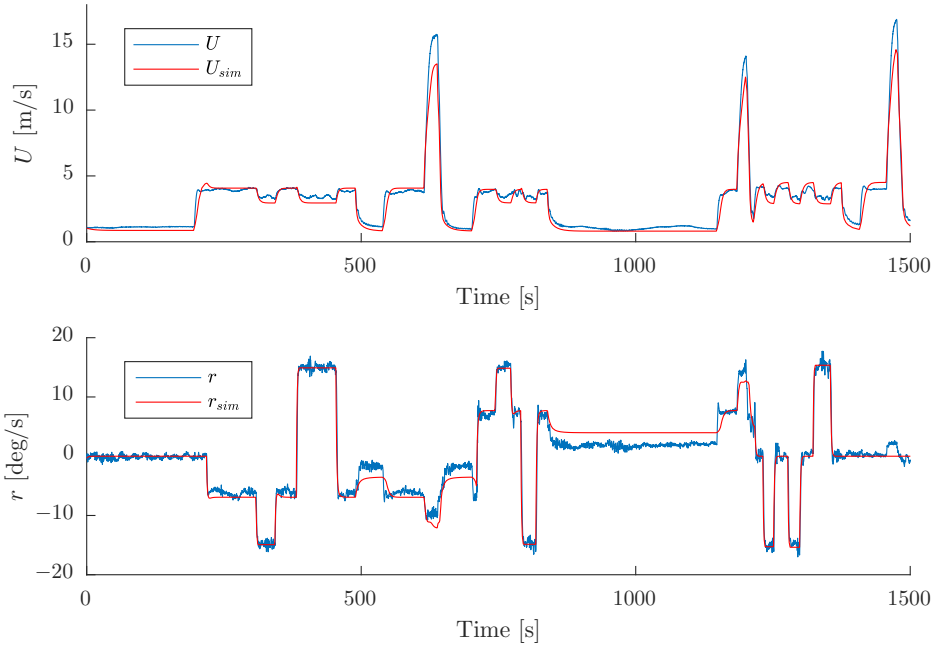


Figure 4.21: Model validation results by comparing simulated and measured data of trial not used for SI (Eriksen and Breivik, 2017).

The results of SI were validated as the simulation using the resulting model was able to reproduce with satisfactory similarity the behaviour of the USV just by receiving the same command inputs. The

discussion for this conclusion of the SI procedure is about what could be the possible causes for the discrepancies between simulation and real states during this model validation.

- **Uncoupled data extraction:** This is discussed in detail in Section 4.2.3.
- **Limitation of polynomial model structure:** Representing the vessel inertia and damping terms with polynomial functions serves its purpose of providing good and simple approximations to complex dynamics, but its downside is the possibility of not capturing the vessel's dynamics completely. It is a known trade-off and the advantages surpass this disadvantage.
- **Vessel asymmetry:** One of the assumptions during SI was that the vessel is port-starboard symmetric. It is rare for the vessel project not to be symmetric, but in practice it is very likely that asymmetries exist. This can clearly be seen during the final part of the validation trial (Figure 4.20), in which the rudder angle remains zero, but a positive yaw rate r appears when throttle is commanded to maximum and it overshoots to negative when the throttle is set back to zero right before the trial is ended. This is a case in which the asymmetry influence is clearly identified, but it might also cause offsets in other parts of the simulation.

In addition, it was identified two causes that could in general create discrepancies between simulation and real states, but were not the case for the model validation of this work.

- **Different weather conditions during trials for identification and validation:** The experiment design is to perform trials in weather conditions as calm as possible, but it is likely that there exist residual wind, waves and currents, which may have an impact on the results. These can have an even bigger impact if they are different during identification and validation trials.
- **No hysteresis in the model:** It is possible for planing vessels to have a hysteresis region of surge speed when entering and leaving the planing mode, as shown in Figure 4.22 by Kragelund et al. (2013), but the control-oriented model does not consider this possibility. Therefore, results may get distorted in the planing region if the USV has this hysteresis behaviour.

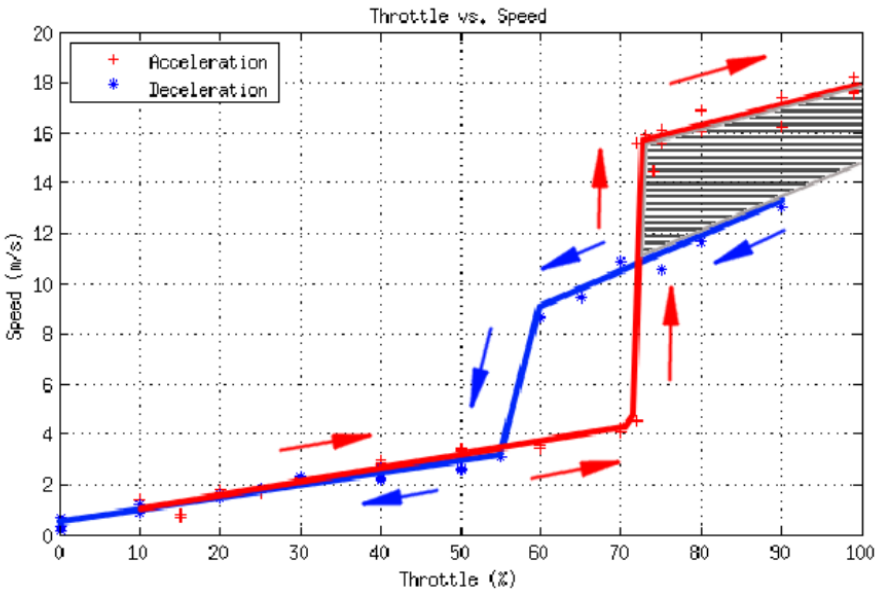


Figure 4.22: Hysteresis in USV surge speed due to entering (acceleration) and leaving (deceleration) planing regime. (Kragelund et al., 2013).

4.5 Reference Filter Saturation Limits

Since the model was validated, the results could be used to find saturation limits for reference filters as discussed in Section 3.5.

For acceleration limits, the results are plotted with dots representing the acceleration estimated for each step as a function of $\Delta\tau_i$ (Equation (3.22)) and dashed lines representing the modeled behaviour (Equation (3.20)) also as a function of $\Delta\tau_i$. The model depends on both states, but a 3D plot made visualization poor, so it was decided to make the plots of maximum acceleration for each state as a function of surge speed U and with yaw rate r equal zero. Figure 4.23 shows the results for maximum acceleration in surge and Figure 4.24 for maximum acceleration in yaw.

For yaw rate limits, the result is shown in Figure 4.25. In this case, the dots in the plot represent the yaw rates measured in steady-state for maximum rudder angle and the line represents the model fitted to these dots as explained in Section 3.5.2.

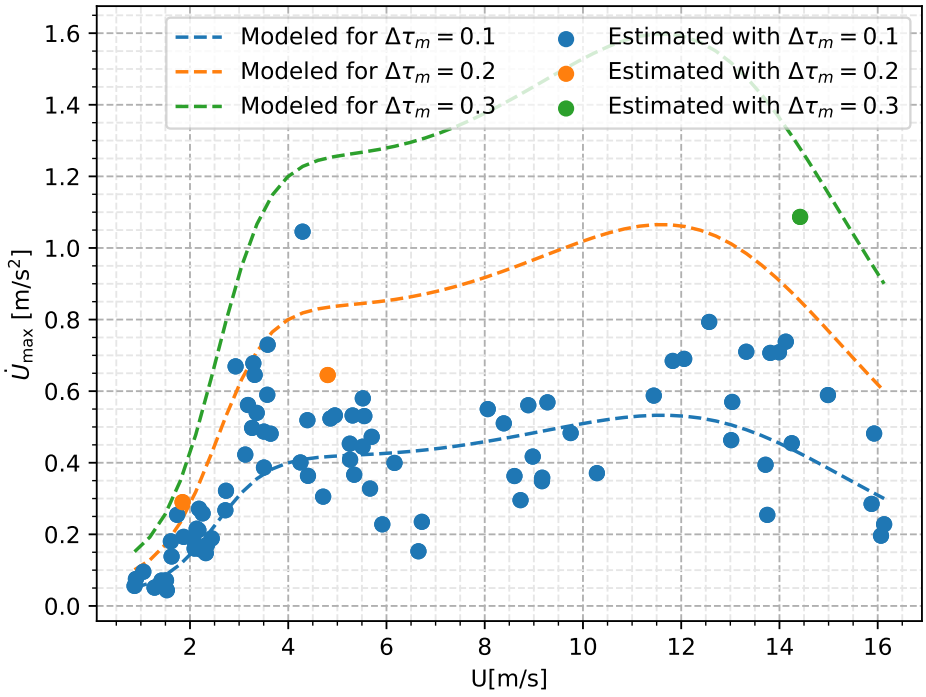


Figure 4.23: Estimated and modeled surge acceleration as a function of surge speed and step in throttle command.

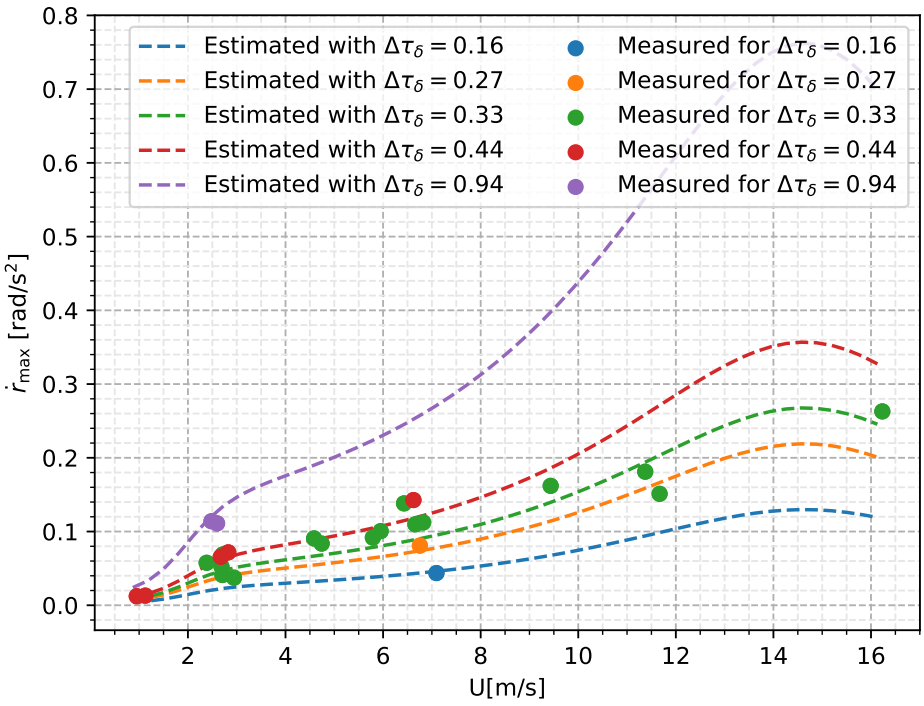


Figure 4.24: Estimated and modeled yaw acceleration as a function of surge speed and step in rudder angle command.

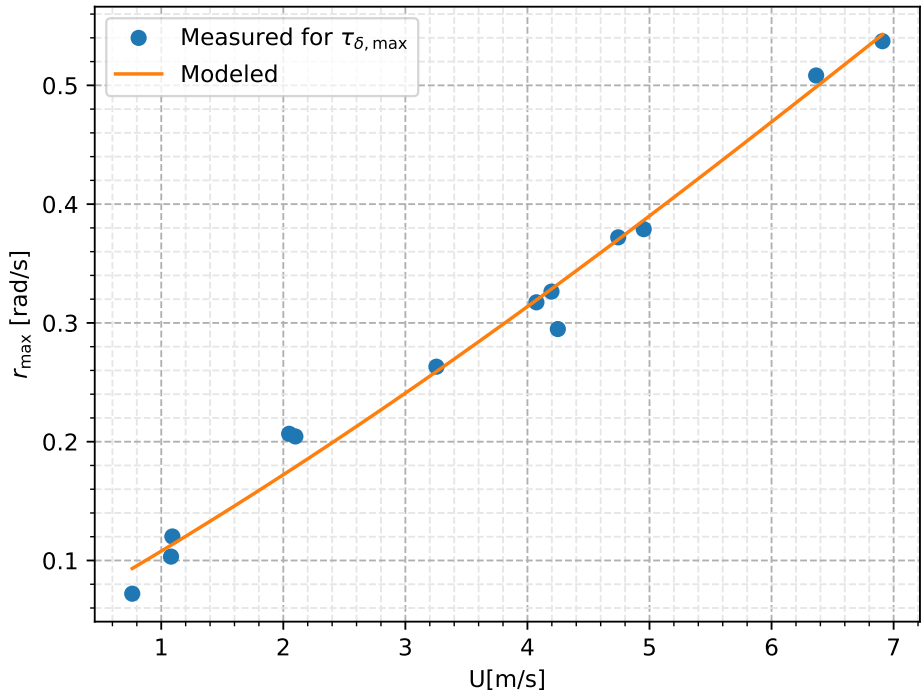


Figure 4.25: Measured and modeled yaw rate as a function of surge speed for maximum rudder angle.

The discrepancies discussed in model validation also affect the results for the reference filter saturation limits considering they are directly calculated from the model. The inverse relation between limits and modelled inertia, as defined in Equation (3.20), makes the errors seem big, because errors that are considered small at inertia modelling are amplified, but the methodology used guarantees that the saturation limits are feasible because it leaves margin for compensating possible imprecision. It is also important to remember that the spread in estimated points compared to the modelled lines in Figures 4.23 and 4.24 is also due to plotting the lines for a yaw rate equal to zero, which is not the case for most of the points, but a reference was necessary to avoid too many lines. On the whole, the results show how the usual tuning with a single constant value is far from the real saturation limits.

Surge and yaw acceleration limits have similar tendencies, but for different reasons. At low speeds, the effectiveness of the rudder is low, so it is not capable of producing much yaw acceleration, but the effectiveness grows as speed grows. A physical explanation for the drop at higher speeds could not be formulated and the hypothesis is that it is due to the smaller number of points in that area, making disturbances more relevant in addition to the previously explained amplification of errors because of inverting inertia results.

On the other hand, the drop for maximum surge acceleration at high speeds may be due to fast growth in drag forces at such speeds, reducing the capacity of the propeller to output higher acceleration when requested. For the same reason, the model reaches a plateau during transition from displacement to planing mode around 5 and 6 meters per second.

Finally, the results for maximum yaw rate in Figure 4.25 present a great fit of the model to the measured data. It is important to remember that for speeds higher than 7 meters per second there should be no extrapolation of the model, but the truncated value of maximum yaw rate for this speed. Problems with model extrapolation are explained in Section 4.3, and although the USV is capable of maintaining higher speeds steadily, it is not when the rudder is at its maximum angle.

4.6 General Remarks

The combination of SIMID for data extraction and methods for automatic removal of unreliable data made possible the pursued automation of SI. This means that, with the entire procedure properly coded, the complete identification of a vessel model can be performed in a single day. The example case in this work gathered a total of less than 3 hours of data, including throttle steps, rudder steps and validation trial. On top of that, there would be no added hours for tuning of parameters for SI, which usually can take a considerable amount of time not only for the tuning itself, but also to get enough experience to perform it.

Since this work used the same converted USV as the reference literature, it was not possible to measure the impact of different configurations of actuators, but there are some expected consequences. The critical stages when considering a different configuration of actuators are experiment design, model structure selection and data extraction. The experiment design must guarantee that the entire operational range of the vessel is covered while not defining a sequence of steps that takes too long to be performed. For an overactuated vessel with multiple redundant actuators, there is a number of possible simplifications that can be made to minimize the extension of the trials while still collecting enough data. The different configurations might impact the model structure due to the necessity of combining commands for different actuators into the commands vector $\boldsymbol{\tau}$. Finally, the data extraction stage might also suffer changes depending on consequences of experiment design and model structure selection, because if step commands have a clear impact on both surge and yaw, the assumption of decoupling becomes impracticable.

Chapter 5

Conclusion

This work successfully implemented improvements to control-oriented modelling such that conversions from SVs to USVs can be better performed. A new data extraction method for control-oriented models, SIMID, simultaneously identifies inertia and damping parameters, skipping the necessity of identifying transitions from transient to steady-state such as performed in previous works. SIMID required no parameter tuning, while returning similarly satisfactory identification results when compared to the reference literature. When combined with methods for removal of unreliable data, it makes the SI procedure automatic. Additionally, the results from SI are successfully used to model saturation limits for reference filters of the system.

Another new data extraction method, RW-SSID, was tested using rolling windows for SSID. It was developed with the intention to require only two tuning parameters, but its utilization was halted when proved that the two parameters had to be independently tuned for every step command. This means it required two times as many tuning parameters as the number of step commands during trials, making its use impracticable.

Regarding recommendations for future work, it was intended to try for this work a experiment design with rudder steps always starting from or ending at zero, reflecting what is usually seen during operation when the boat transitions from turning to straight ahead, instead of changing between rotating states. However, no new experiments could be run for this work, and this alternative experiment design is left for future work. Another suggestion for future work is to explore the coupling of inertia parameters, identifying the coupled parameters during data extraction since results show relevant coupling in some cases. The final recommendation is to test this methodology on a vessel with a configuration of actuators other than a single outboard engine, as used in this work and reference literature.

Bibliography

Abkowitz, M. A. (1980). Measurement of hydrodynamic characteristics from ship maneuvring trials by system identification. *SNAME Transactions*, 88:283–318.

Bertram, V. (2008). Unmanned Surface Vehicles - A Survey. In *Skibsteknisk Selskab, Copenhagen, Denmark*, pages 1–14.

Blanke, M. and Christensen, A. C. (1993). Rudder-Roll Damping Autopilot Robustness to Sway-Yaw-Roll Couplings. In *10th Ship Control Systems Symposium*.

Breivik, M., Hovstein, V. E., and Fossen, T. I. (2008). Straight-line target tracking for unmanned surface vehicles. *Modeling, Identification and Control*, 29(4):131–149.

Cao, S. and Rhinehart, R. R. (1995). An efficient method for on-line identification of steady state. *Journal of Process control*, 5(6):363–374.

Dalheim, Ø. Ø. and Steen, S. (2020). A computationally efficient method for identification of steady state in time series data from ship monitoring. *Journal of Ocean Engineering and Science*, pages 1–13.

Daniels, W. J. and Tucker, H. B. (1952). *Model Sailing Craft*. Chapman & Hall, 3rd edition.

de Freitas, C. S. (2019). *Control system solutions for conversion of surface vessels to unmanned vessels*. Project Thesis (Unpublished), NTNU.

Eriksen, B.-O. H. and Breivik, M. (2017). Modeling, Identification and Control of High-Speed ASVs: Theory and Experiments. In Fossen, T., Pettersen, K., and Nijmeijer, H., editors, *Sensing and Control for Autonomous Vehicles. Lecture Notes in Control and Information Sciences*, volume 474. Springer, Cham.

Eriksen, B.-O. H. and Breivik, M. (2018). A Model-Based Speed and Course Controller for High-Speed ASVs. *IFAC-PapersOnLine*, 51(29):317–322.

Førthmann, P. C. (1998). *Self-Steering under Sail*. McGraw-Hill, 1st edition.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. John Wiley & Sons.

Fossen, T. I. (2005). A nonlinear unified state-space model for ship maneuvering and control in a seaway. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, 15(9):2717–2746.

Fossen, T. I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.

Fossen, T. I. and Perez, T. (2004). Marine Systems Simulator (MSS). https://github.com/cybergalactic/MSS. Accessed: 27-01-2020.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Kragelund, S., Dobrokhodov, V., Monarrez, A., Hurban, M., and Khol, C. (2013). Adaptive speed control for autonomous surface vessels. *2013 OCEANS - San Diego*, (March):1–10.

Kvalvaag, T. M. (2018). *Automatic model identification of high-speed autonomous surface vehicles*. MSc Thesis, NTNU.

Liu, Z., Zhang, Y., Yu, X., and Yuan, C. (2016). Unmanned surface vehicles: An overview of developments and challenges. *Annual Reviews in Control*, 41:71–93.

Ljung, L. (1998). System identification. In Procházka, A., Uhlíř, J., Rayner, P. W. J., and Kingsbury, N. G., editors, *Signal Analysis and Prediction*, pages 163–173. Birkhäuser Boston, Boston, MA.

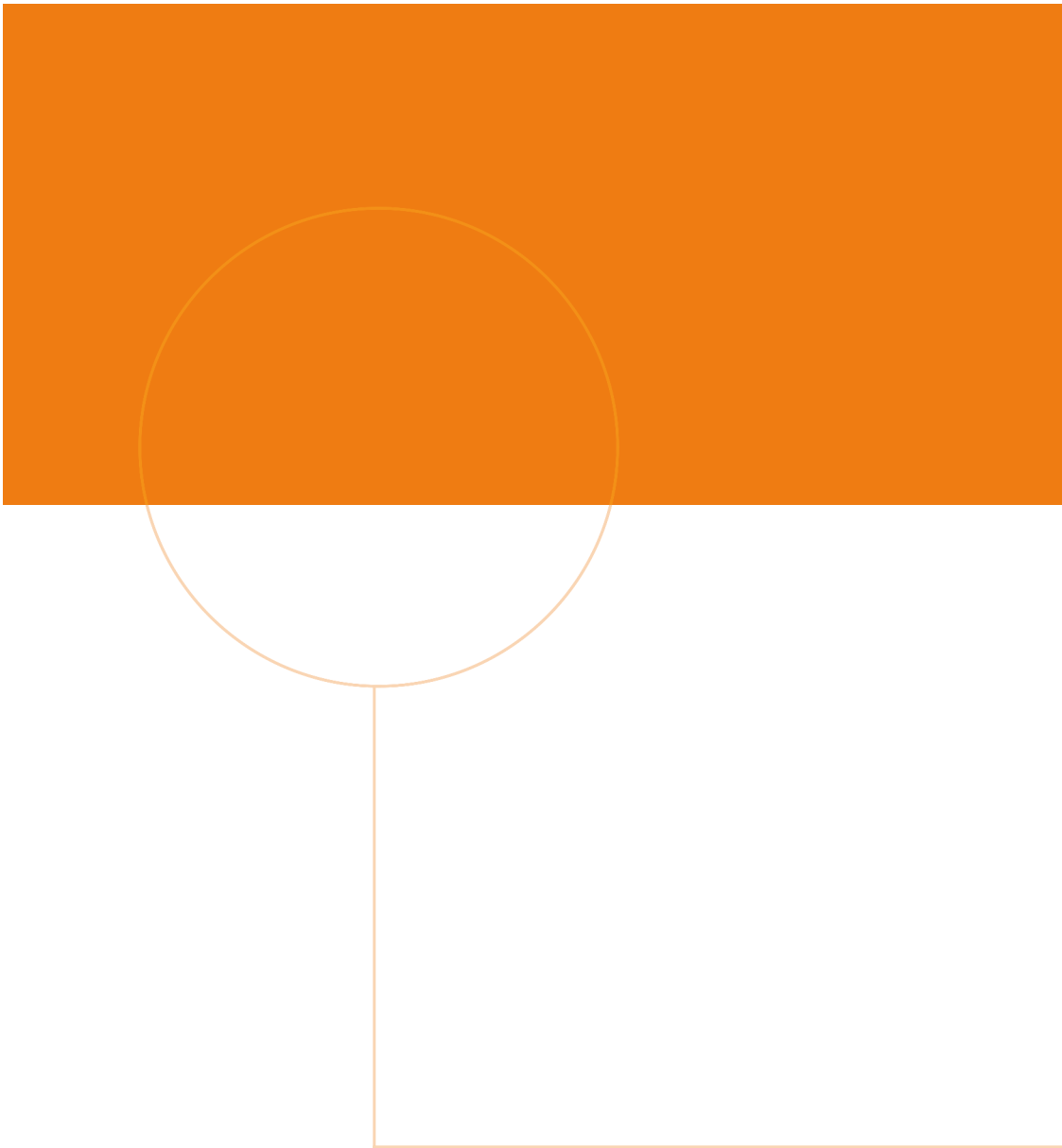
Maritime Safety Commitee (2018). MSC 100/5 - Regulatory Scoping Exercise for the Use of Maritime Autonomous Surface Ships (MASS). Technical report, International Maritime Organization.

Minorsky, N. (1922). Directional stability of automatically steered bodies. *Journal of the American Society for Naval Engineers*, 34(2):280–309.

Nomoto, K., Taguchi, T., Honda, K., and Hirano, S. (1957). On the steering qualities of ships. *International Shipbuilding Progress*, 4(35):354–370.

Perez, T. and Fossen, T. I. (2007). Kinematic models for manoeuvring and seakeeping of marine vessels. *Modeling, Identification and Control*, 28(1):19–30.

Sørensen, A. J. (2011). A survey of dynamic positioning control systems. *Annual Reviews in Control*, 35(1):123–136.



NTNU
Kunnskap for en bedre verden

MARITIME ROBOTICS