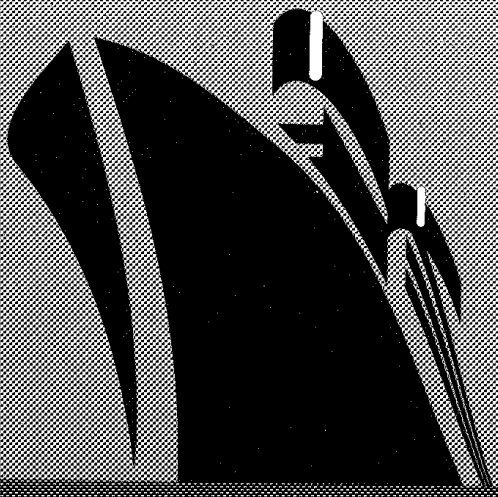
Marine Control Systems

Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles

Thor I. Fossen

Marine Control Systems

Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles



Thor I. Fossen

Marine Control Systems

Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles

Thor I. Fossen
Norwegian University of Science and Technology
Trondheim, Norway

MARINE CYBERNETICS
Trondheim, Norway
http://www.marinecybernetics.com

For ordering see URL: http://www.marinecybernetics.com. The book can also be ordered by sending an e-mail to:

info@marinecybernetics.com

or via fax:

MARINE CYBERNETICS
Trondheim, Norway
fax: [+47] 72 81 00 18

ISBN 82-92356-00-2

Produced from camera-ready copy supplied by the author using *Scientific WorkPlace*.

Printed and bound by Tapir Trykkeri, Trondheim, Norway.

This book is dedicated

to

Professor Jens G. Balchen

who introduced me to the fantastic
world of feedback control.

Contents

Preface

The main motivation for writing this book was to collect new results on nonlinear control of marine craft that have appeared since I published my first book: "*Guidance and Control of Ocean Vehicles*" (John Wiley & Sons Ltd. 1994). Most of these results have been developed in the Department of Engineering Cybernetics at Norwegian University of Science and Technology (NTNU) in close cooperation with my doctoral students; *Ola-Erik Fjellstad, Trygve Lauvdal, Jann Peter Strand, Jan Fredrik Hansen, Bjørnar Vik, Svein Peder Berge, Mehrdad P. Fard, Karl-Petter Lindegaard, Ole Morten Aamo,* and *Roger Skjetne* in the period 1991–2002. We have all been a great team, producing more than one hundred international publications in this period. These have resulted in several patents and industrial implementations.

In particular, I want to express my gratitude to *Mr. Roger Skjetne* and *Dr. Jann Peter Strand* for suggestions, case studies and comments to the manuscript. They have been instrumental in most of the work presented in the book. United European Car Carriers (UECC) and SeaLaunch LLC should also be thanked for contributing with full scale experimental results and case studies to the book. *Dr. Svein Peder Berge* and *Mr. Lars Ove Sæther* at Marintek AS have contributed with experimental results from the Ocean Basin in which model ships have been tested. *Adjunct Professor Svein I. Sagatun* and Norsk Hydro should be thanked for supporting our research projects on inertial navigation systems and marine operations.

Dr. Bjørnar Vik has been invaluable as a research fellow and colleague in the period 1998–2002. His expertise in the rapid prototyping of ship control systems, control theory and navigation systems has been most valuable. This expertise has been one of the keystones in the development of the GNC Lab (Guidance, Navigation and Control Laboratory), MCLAB (Marine Cybernetics Laboratory), and GPS/INS Laboratory at NTNU.

I would like to thank my family, *Heidi, Sindre,* and *Lone* for supporting this book project. Without their support it would have been impossible to accomplish the task of writing 586 pages with equations in less than two years.

Professor Asgeir J. Sørensen and *Professor Tor Arne Johansen* should be thanked for their careful proofreading and comments on the final manuscript. *Professor Olav Egeland* should be thanked for our mutual discussions on modeling and control. I am also grateful to *Mr. Andrew Ross* at the University of Glasgow for his assistance with the English language. The book also greatly benefits from students who took the course in Guidance, Navigation and Control at NTNU in 2002. They have all helped me to keep the number of typographical errors to an acceptable level.

Finally, I would like to thank *Professor Miroslav Krstic* for inviting me on a sabbatical at the University of California, San Diego (UCSD) in 2001, so that I could escape the office in Norway and finish the book in a reasonable time.

Thor I. Fossen
November 2002

List of Tables

Chapter 1

Introduction

The subject of this book is *control of marine vessels.* By marine vessels we mean ships, high-speed craft, semi-submersibles, floating rigs, submarines, remotely operated and autonomous underwater vehicles, torpedoes and other propelled/powered structures e.g. a floating air field.

The words *boat* and *ship* are often used incorrectly about the same vessels. In *Encyclopedia Britannica* a ship and a boat are distinguished by their size through the following definition:

> *Ship:* "any large floating vessel capable of crossing open waters, as opposed to a boat, which is generally a smaller craft. The term formerly was applied to sailing vessels having three or more masts; in modern times it usually denotes a vessel of more than *500* tons of displacement. Submersible ships are generally called boats regardless of their size".

Similar definitions are given for submerged vehicles:

> *Submarine:* "any naval vessel that is capable of propelling itself beneath the water as well as on the water's surface. This is a unique capability among warships, and submarines are quite different in design and appearance from surface ships."
>
> *Underwater Vehicle:* "small vehicle that is capable of propelling itself beneath the water surface as well as on the water's surface. This includes unmanned underwater vehicles (UUV), remotely operated vehicles (ROV), autonomous underwater vehicles (AUV) and underwater robotic vehicles (URV). Underwater vehicles are used both commercially and by the navy."

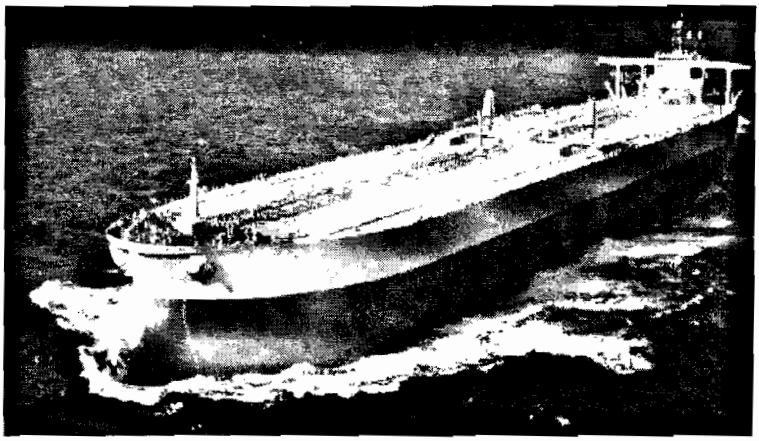


Figure 1.1: Oil tanker.

Degrees of Freedom and Motions of a Marine Vessel

In maneuvering, a marine vessel experiences motion in 6 degrees of freedom (DOF). The motion in the horizontal plane is referred to as *surge* (longitudinal motion, usually superimposed on the steady propulsive motion) and *sway* (sideways motion). Heading, or *yaw* (rotation about the vertical axis) describes the course of the vessel. The remaining three DOFs are *roll* (rotation about the longitudinal axis), *pitch* (rotation about the transverse axis), and *heave* (vertical motion), see Figure 2.1 in Chapter 2 for a more detailed explanation.

Roll is probably the most troublesome DOF, since it produces the highest accelerations and, hence, is the principal villain in seasickness. Similarly, pitching and heaving feel uncomfortable to humans. When designing ship autopilots, yaw is the primary mode for feedback control. Station-keeping of a marine vessel implies stabilization of the surge, sway and yaw modes.

Feedback Control Applied to Marine Vessels

Feedback control systems and their application to *marine vessels* have become more and more popular thanks to the developments in computer science, propulsion systems and modern sensor technology. Examples of commercially available systems are:

- ship and underwater vehicle autopilots for course-keeping and turning control
- way-point tracking, trajectory and path control systems for marine vessels
- depth autopilots for underwater vehicles
- torpedo control systems
- attitude control systems for underwater vehicles
- dynamic positioning (DP) systems for marine vessels
- positioning mooring (PM) systems for floating vessels

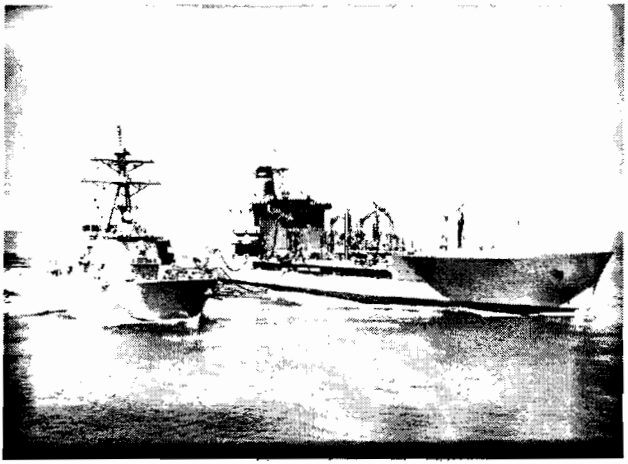


Figure 1.2: Two navy ships moving in open seas. Courtesy US Navy.

- fin and rudder-roll stabilization (RRS) systems
- wave-induced vibration damping systems for high-speed craft
- buoyancy control systems including trim and heel correction systems
- propulsion control systems and forward speed control systems
- propeller and thruster control systems
- energy and power management systems

1.1 From the Invention of the Gyroscope to Model Based Ship Control

The history of model based ship control starts with the invention of the *gyrocompass* in 1908, and it extends further with the development of local positioning systems in the 1970s. Global coverage using satellite navigation systems was first made available in 1994. The gyrocompass was the basic instrument in the first feedback control system for heading control and today these devices are known as autopilots.

Introduction of local area ship positioning systems like hydro acoustic reference systems (*SSBL, SBL, LBL*), hyperbolic radio navigation systems (*Decca, Loran-C, Omega*), local electromagnetic distance measuring (EDM) systems (*Artemis, Autotape, Miniran, Mini-Ranger III, Syledis, Tellurometer, Trident III, Trisponder*), taut wire etc. in conjuncture with new results within feedback control resulted in new applications like dynamic positioning systems for ships and rigs.

In 1994 *Navstar GPS* was declared fully operational (global coverage) even though the first satellite was launched in 1974 (Parkinson and Spilker 1995). Today, GPS receivers are

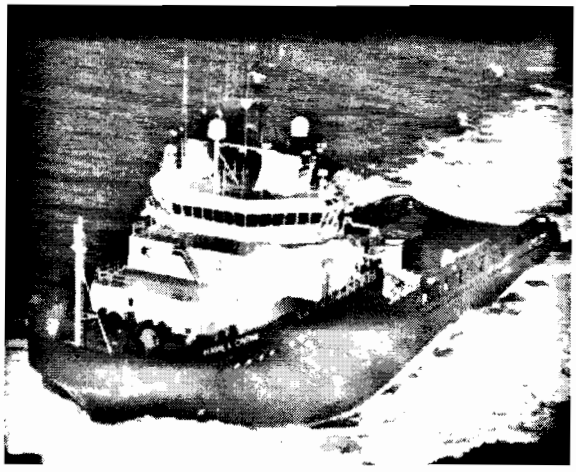


Figure 1.3: Offshore supply vessel.

standard components in way-point tracking control systems and ship positioning systems world wide. They are used commercially and by the Navy.

1.1.1 The Gyroscope and its Contributions to Ship Control

During the 1850s the French scientist *J. B. L. Foucault* conducted experiments with a wheel (rotor) mounted in gimbal rings–i.e., a set of rings that permit it to turn freely in any direction. The name gyroscope was adopted for this device. In the experiments Foucault noticed that the spinning wheel maintained its original orientation in space regardless of the Earth's rotation.

In *Encyclopedia Britannica* the following definition is given for a gyroscope:

> *Gyroscope*: "any device consisting of a rapidly spinning wheel set in a framework that permits it to tilt freely in any direction–i.e., to rotate about any axis. The momentum of such a wheel causes it to retain its attitude when the framework is tilted; from this characteristic derive a number of valuable applications. Gyroscopes are used in such instruments as compasses and automatic pilots onboard ships and aircraft, in the steering mechanisms of torpedoes, in antiroll equipment on large ships, and in inertial guidance systems".

The first recorded construction of the gyroscope is usually credited to *C. A. Bohnenberger* in 1810 while the first electrically driven gyroscope was demonstrated in 1890 by *G. M. Hopkins* (see Allensworth 1999, Bennet 1979).

The development of the electrically driven gyroscope was motivated by the need for more reliable navigation systems in steel ships and underwater warfare. A magnetic compass, as opposed to a gyro compass, is highly sensitive to magnetic disturbances, which are commonly found within steel ships and submarines equipped with electrical devices. In parallel works, *Dr. H. Anschutz* of Germany and *Elmer Sperry* of the USA both worked on a practical application of the gyroscope.

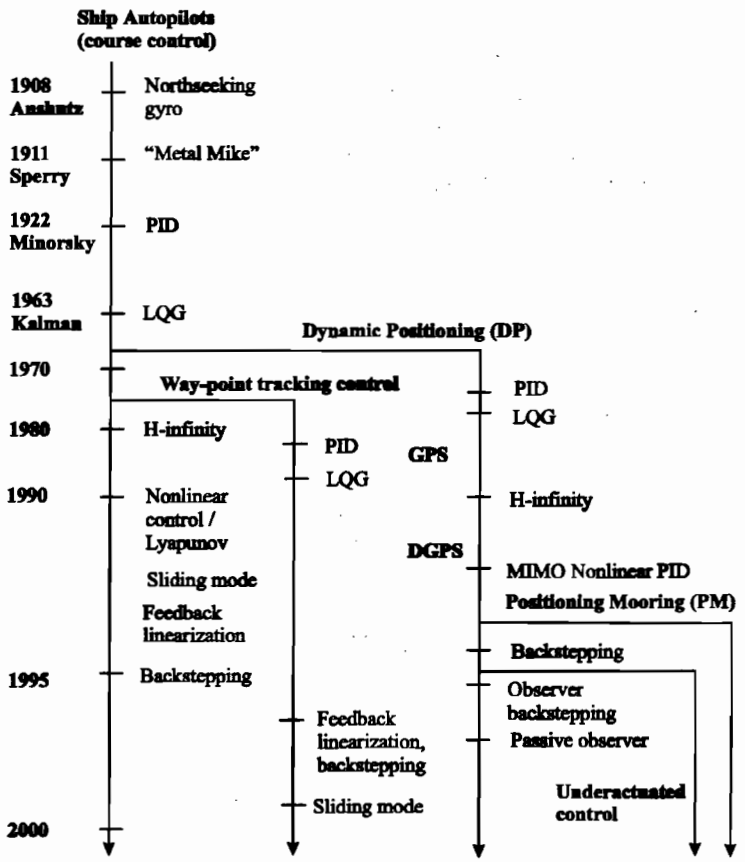


Figure 1.4: Diagram showing the developments from the early autopilot to modern ship control systems (Fossen 2000*a*).

In 1908 Anschutz patented the first North seeking gyrocompass, while Elmer Sperry was granted a patent for his ballistic compass including vertical damping three years later.

The invention of the gyroscope was one of the key breakthroughs in automatic ship control since it lead to the development of the *automatic pilot*; see Figure 1.4 (Fossen 2000*a*). Another useful reference which discusses yacht control systems is Fossen (2000*b*).

1.1.2 Autopilots

The autopilot or *automatic pilot* is a device for controlling an aircraft, ship or other vehicles without constant human intervention. The earliest automatic pilots could do no more than maintain a fixed heading and they are still used to relieve the pilot on smaller boats during routine cruising. For ships, course-keeping capabilities were the first applications. Modern autopilots can, however, execute complex maneuvers like turning, docking operations, or make possible the control of inherently unstable vessels (such as submarines and some large oil tankers). Autopilots are used to steer surface ships, submarines, torpedoes, missiles,

rockets, and spacecraft among others.

As mentioned earlier, the work on the gyrocompass was extended to ship steering and closed-loop control by *Elmer Sperry* (1860-1930), who constructed the first automatic ship steering mechanism in 1911 (see Allensworth 1999, Bennet 1979). This device, referred to as the "*Metal Mike*", was a gyroscope-guided autopilot or a mechanical helmsman. Metal Mike emulated much of the behavior of a skilled pilot or a helmsman, including compensating for varying sea states using feedback control and automatic gain adjustments.

Later, in 1922, *Nicholas Minorsky* (1885-1970) presented a detailed analysis of a position feedback control system where he formulated a three-term control law which today is referred to as *Proportional-Integral-Derivative* (PID) control (see Minorsky 1922). Observing the way in which a helmsman steered a ship motivated these three different behaviors. In Bennet (1979), there is an interesting analysis of the work of Sperry and Minorsky and their contributions to autopilot design.

The autopilot systems of Sperry and Minorsky were both single-input single-output (SISO) control systems, where the heading (yaw angle) of the ship was measured by a gyrocompass. Today, this signal is fed back to a computer, in which a PID control system (autopilot) is implemented in software, see Section 8.4. The autopilot compares the pilot set-point (desired heading) with the measured heading and computes the rudder command, which is then transmitted to the rudder servo for corrective action.

More recently PID-type autopilots have been replaced by autopilots based on LQG and $\mathcal{H}_\infty$-control design techniques. One of the nice features with these design techniques is that they allow for frequency dependent notch filtering of 1st-order wave-induced disturbances; see Chapter 6. Frequency components around the peak frequency of the wave spectrum in yaw must be prevented from entering the feedback loop in order to avoid wear and tear of the thruster and propeller systems. The drawback of the PID-controller in cascade with a dead-band, notch and/or low-pass filter is that additional phase lag and nonlinearities are introduced in the closed-loop system; see Section 6.1.2. A model-based state estimator (Kalman filter) reduces these problems. Linear quadratic and $\mathcal{H}_\infty$ autopilot designs have been reported in the literature by a large number of authors; see Koyama (1967), Norrbin (1972), Van Amerongen and Van Nauta Lemke (1978), Van Amerongen and Van Nauta Lemke (1980), Donha *et al.* (1998), Tzeng (1998), and Fossen (1994) and references therein to mention only some.

In addition to LQG and $\mathcal{H}_\infty$-control, other design techniques have been applied to ship autopilot designs, for instance nonlinear control theory. Autopilot designs for nonlinear systems are treated in detail in Section 8.4.

1.1.3 Dynamic Positioning and Position Mooring Systems

The great successes with PID-based autopilot systems, and the development of local area positioning systems suggest that three decoupled PID-controller could be used to control the horizontal motion of a ship in surge, sway and yaw exclusively by means of thrusters and propellers. The idea was tested in the 1970s, and the invention was referred to as a *dynamic positioning* (DP) system; see Chapter 11.

As for the autopilot systems, a challenging problem was to prevent 1st-order wave-induced disturbances entering the feedback loop. Several techniques like notch and low-pass filtering, and the use of dead band techniques were tested for this purpose, but with varying levels of success.

In 1960–1961 the *Kalman filter* was published by Kalman (1960), and Kalman and Bucy

(1961). Two years later in 1963, the theory for the linear quadratic (LQ) optimal controller was available. This motivated the application of *LQG-controllers* in MIMO ship control like DP since a state observer (Kalman filter) could be used to estimate the wave frequency (WF) and the ship low-frequency (LF) motions, see Figure 11.5 in Section 11.2. Another advantage of a MIMO control strategy was that the interactions between the surge, sway and yaw modes could be properly compensated for. This is not possible with three decoupled PID controllers.

The LQG design technique was first applied to DP by Balchen *et al.* (1976, 1980a, 1980b), and Grimble *et al.* (1979, 1980a). Later Grimble and coauthors suggested to use $\mathcal{H}_\infty$ and μ-methods for filtering and control (Katebi *et al.* 1997*a*). These methods have been further refined by Katebi *et al.* (1997*b*) where nonlinear thruster dynamics is included using describing functions.

After 1995, nonlinear PID-control, passive observer design and observer backstepping designs have been applied to DP by Fossen and coauthor with good results, see Fossen (1994), Grøvlen and Fossen (1996), Fossen and Grøvlen (1998), Strand (1999) and references therein. An overview of DP systems is found in Strand and Sørensen (2000) while extensions to PM mooring systems are found in Strand (1999). DP and PM systems are described more closely in Sections 11.2 and 11.3.

1.1.4 Way-Point Tracking Control Systems

The successful results with LQG controllers in ship autopilots and DP systems, and the availability of global navigation systems like GPS and GLONASS resulted in a growing interest for way-point tracking control systems, see Holzhüter and Schultze (1996), and Holzhüter (1997), and references therein. The transformation of the way-points to a feasible path or trajectory is in general a nonlinear optimization problem, see Section 5.2. The controller can be designed using linear theory or by treating the control problem as a nonlinear control problem. Extensions to nonlinear trajectory tracking and maneuvering control are currently new fields of research. In Chapter 10 the most recent results on nonlinear control for ships are discussed. Guidance systems for trajectory tracking and maneuvering control are discussed in Chapter 5.

1.1.5 The Sea Launch System

In 1999 the disciplines of aeronautics, astronautics and hydrodynamics were truly unified when the first offshore rocket launch platform, known as the *SeaLaunch system,* was constructed, see http://www.sea-launch.com for more details. In this project a semi-submersible was modified to a floating launch pad, so that rockets and their payloads could be launched to all inclinations using the most direct route to orbit, see Figure 1.5. This gives maximum lift capacity and increased payload/mass.

This was a joint project between the *Boeing Commercial Space Company*, which provided the payload fairing, analytical and physical spacecraft integration and mission operations; *RSC Energia*, which provided the Block DM-SL upper stage, launch vehicle integration and mission operations; *SDO Yuzhnoye/PO Yuzhmash*, which provided the first two Zenit-3SL stages, launch vehicle integration support and mission operations; and Anglo-Norwegian *Kværner Group*, providing operational services of the launch platform Odyssey and assembly and the command ship, Sea Launch Commander.

Figure 1.5: The SeaLaunch system. Courtesy Sea Launch LLC.

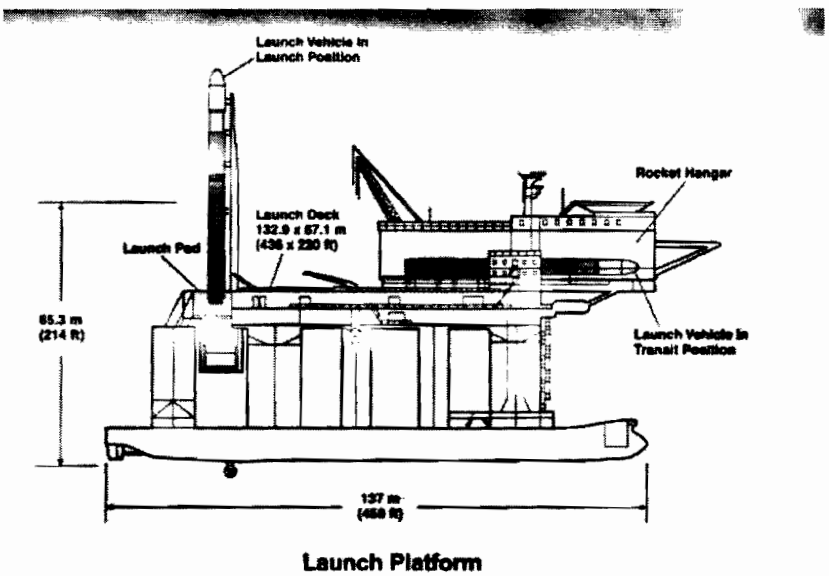


Figure 1.6: Schematic drawing of the SeaLaunch platform. Courtesy Sea Launch LLC.

The SeaLaunch system clearly indicated the need for sophisticated feedback control systems offshore. In this project the position of the vessel is maintained by a dynamic positioning system using feedback from a satellite navigation system and a gyro compass. The roll and pitch angles of the vessels are kept to a constant value, even when the launch vehicle and the rocket is moved from the storage facility to the launch pad. This is achieved by the trim and heel correction systems which are pumping water between the platform legs, see Figure 1.6.

1.2 Model Representations for Marine Vessels

When designing advanced control systems for marine vessels it is desirable to take advantage of physical model properties of the vessel and exploit them in the design of the controller.

1.2.1 The Classical Model in Naval Architecture

Consider the model:

$$\mathbf{M}\dot{\boldsymbol{\nu}} = \sum_{i=1}^{n} \mathbf{F}_i \tag{1.1}$$

where $\mathbf{M}$ is the system inertia matrix, $\dot{\boldsymbol{\nu}}$ is the generalized acceleration vector:

$$\boldsymbol{\nu} = [u, v, w, p, q, r]^\top \tag{1.2}$$

Here (u, v, w) are the linear velocities in surge, sway and heave, (p, q, r) are the angular velocities in roll, pitch and yaw, and:

$$\mathbf{F}_i = [X_i, Y_i, Z_i, K_i, M_i, N_i]^\top, \qquad (i = 1, \ldots, n) \tag{1.3}$$

is a vector of linearly superpositioned forces and moments.

This model is motivated by Newton's law: $F = ma$ where F represent force, m is the mass and a is the acceleration. The model representation (1.1) is used in most textbooks on hydrodynamics where the forces $\mathbf{F}_i$ usually are written in component form using the SNAME (1950) notation.

If it is assumed that $n = 1$ and that $\mathbf{F}_1$ describes only linear damping and added mass, then this results in a total of 72 elements denoted by:

$$\begin{aligned} X_1 &= -X_u u - X_v v - X_w w - X_p p - X_q q - X_r r \\ &\quad -X_{\dot{u}}\dot{u} - X_{\dot{v}}\dot{v} - X_{\dot{w}}\dot{w} - X_{\dot{p}}\dot{p} - X_{\dot{q}}\dot{q} - X_{\dot{r}}\dot{r} \\ &\vdots \\ N_1 &= -N_u u - N_v v - N_w w - N_p p - N_q q - N_r r \\ &\quad -N_{\dot{u}}\dot{u} - N_{\dot{v}}\dot{v} - N_{\dot{w}}\dot{w} - N_{\dot{p}}\dot{p} - N_{\dot{q}}\dot{q} - N_{\dot{r}}\dot{r} \end{aligned}$$

where $X_u, X_v, ..., N_r$ are the linear damping coefficients and $X_{\dot{u}}, X_{\dot{v}}, ..., N_{\dot{r}}$ represent hydrodynamic added mass. Nonlinear theory implies that hundreds of elements must be included in addition to these 36 elements; see Abkowitz (1964), for instance. Hence, model-based control design using (1.1) becomes relatively complicated due to the large number of

hydrodynamic coefficients on the right-hand-side of equation (1.1). These coefficients are difficult to determine with sufficient accuracy. Consequently, it would be beneficial to exploit physical system properties to reduce the number of coefficients needed for control. This was the main motivation for developing a vectorial representation of the model (Fossen 1991).

1.2.2 The Vectorial Model Representation of Fossen (1991)

In Fossen (1991) the robot model (Craig 1989, Sciavicco and Siciliano 1996):

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{q} = \boldsymbol{\tau} \tag{1.4}$$

was used as motivation to derive a compact marine vessel model in 6 DOFs using a vectorial setting. In the robot model $\mathbf{q}$ is a vector of joint angles, $\boldsymbol{\tau}$ is the torque, while $\mathbf{M}$ and $\mathbf{C}$ denote the system inertia and Coriolis matrices, respectively. It turns out that similar quantities can be identified for marine vessels and aircraft. In Fossen (1991) a complete 6 DOF vectorial setting for marine vessels were derived based on these ideas. These results were further refined by Sagatun and Fossen (1991), Fossen (1994), Fossen and Fjellstad (1995), and Berge and Fossen (2000). The 6 DOF model considered in this book is written in a vectorial setting according to:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{1.5}$$

where

$$\boldsymbol{\nu} = [u, v, w, p, q, r]^\top \tag{1.6}$$

$$\boldsymbol{\eta} = [x, y, z, \phi, \theta, \psi]^\top \tag{1.7}$$

are vectors of velocities and position/Euler angles, respectively. The model matrices $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{D}$ denote inertia, Coriolis and damping, respectively, while $\mathbf{g}$ is a vector of gravitational/buoyancy forces and moments.

Component Form versus Vectorial Setting

It turns out that it is highly advantageous to use the model (1.5) instead of (1.1) when designing control systems, since system properties like symmetry, skew-symmetry and positiveness of matrices can be incorporated into the stability analysis. In addition, these properties are related to passivity of the hydrodynamic and rigid-body models (Berge and Fossen 2000). The system properties represent physical properties of the system, which should be exploited when designing controllers and observers for marine vessels. As a consequence, Equation (1.5) is chosen as foundation for this book and the book "*Guidance and Control of Ocean Vehicles*" by Fossen (1994). The model (1.5) has also been adopted by the international community as a standard model for marine control systems design (control modelling) while the "classical model" is the most used in hydrodynamic modelling where isolated effects can be studied.

It should be noted that the "classical" model (1.1) and the model (1.5) are equivalent when written in component form. Therefore it is possible to combine the best of both approaches, that is hydrodynamic and control modelling. However, it is much easier to construct MIMO controllers and observers when using the vectorial representation, since model properties and model reduction follow from basic matrix properties.

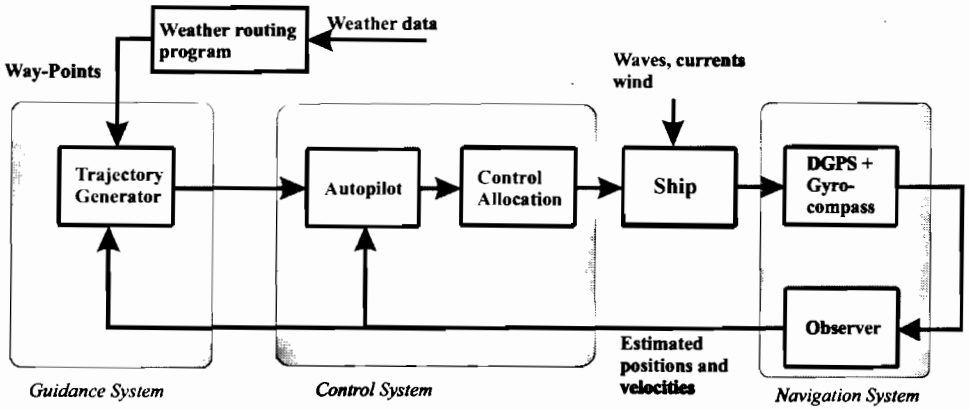


Figure 1.7: Guidance, Navigation and Control.

1.3 The Principle of Guidance, Navigation and Control

A marine vessel control system is usually constructed as three independent blocks denoted as the guidance, navigation and control (GNC) systems. These systems interacts with each other through data and signal transmission as illustrated in Figure 1.7 where a conventional ship autopilot is shown.

GNC, in its most basic form, is a reference model (guidance system), a sensor system (navigation system) and a feedback control system.

1.3.1 Definitions of Guidance, Navigation and Control

In its most advanced form, the GNC blocks represent three interconnected subsystems:

Guidance is the action or the system that continuously computes the reference (desired) position, velocity and acceleration of a vessel to be used by the control system. These data are usually provided to the human operator and the navigation system. The basic components of a guidance system are motion sensors, external data like weather data (wind speed and direction, wave height and slope, current speed and direction, etc.) and a computer. The computer collects and processes the information, and then feeds the results to the vessel's control system. In many cases, advanced optimization techniques are used to compute the optimal trajectory or path for the vessel to follow. This might include sophisticated features like fuel optimization, minimum time navigation, weather routing, collision avoidance, formation control and schedule meetings.

Navigation is the science of directing a craft by determining its position, course, and distance traveled. In some cases velocity and acceleration are determined as well. This is usually done by using a satellite navigation system combined with motion sensors like accelerometers and gyros. The most advanced navigation system for marine applications is the *inertial navigation system* (INS). Navigation is derived from the Latin *navis*, "ship," and *agere*, "to drive." It originally denoted the art of ship driving, including steering and setting the sails. The skill is even more ancient than the word itself,

and it has evolved over the course of many centuries into a technological science that encompasses the planning and execution of safe, timely, and economical operation of ships, underwater vehicles, aircraft, and spacecraft.

Control is the action of determining the necessary control forces and moments to be provided by the vessel in order to satisfy a certain *control objective*. The desired control objective is usually seen in conjunction with the guidance system. Examples of control objectives are minimum energy, set-point regulation, trajectory tracking, path following, maneuvering etc. Constructing the control algorithm involves the design of feedback and feedforward control laws. The outputs from the navigation system, position, velocity and acceleration, are used for feedback control while feedforward control is implemented using signals available in the guidance system and other external sensors.

1.3.2 Set-Point Regulation versus Trajectory Tracking Control

In GNC it is important to distinguish between the following two important control objectives:

Set-Point Regulation: The most basic guidance system is a constant input (set-point) provided by a human operator. The corresponding controller will then be a *regulator*. Examples of set-point regulation are constant depth, trim, heel and speed control, etc. It could also be regulation to zero which is commonly required in roll and pitch for instance.

Trajectory Tracking Control: The objective is for the position and velocity of the vessel to track given desired time-varying position and velocity reference signals. The corresponding feedback controller must then be a *trajectory tracking controller*. Tracking control can be used for course-changing maneuvers, speed changing, attitude control, etc. An advanced guidance system computes optimal time-varying trajectories from a dynamic model and a predefined control objective. If a constant set-point is used as input to a low-pass filter (reference model) the outputs of the filter will be smooth time-varying reference trajectories for position, velocity and acceleration (PVA).

1.4 Organization of Book

The book is organized in five parts:

- I Modeling of Marine Vessels
- II Guidance, Navigation and Control Fundamentals
- III Ship and Rig Applications
- IV Underwater Vehicle Applications
- V Appendices

in addition to the introduction chapter, see Figure 1.8. In Part I we deal with kinematics and dynamics, and develop different physical models of marine vessels. Then, fundamental theory for each of the GNC subsystems is given in Part II. Finally, Parts III and IV deal with

special applications for which tailored models apply and some control design methodologies are suggested. It is recommended that Parts I and II are read before starting with the applications chapters in Parts III and IV. For readers with a good background in kinematics and dynamics, the GNC and application parts can be read directly.

	1. Introduction			
Part I: Modeling of Marine Vessels	2. Kinematics 3. Dynamics of Marine Vessels 4. Models for Wind, Waves and Ocean Currents			
Part II: Guidance, Navigation and Control Fundamentals	5. Maritime Guidance Systems	6. Estimator Based Navigation Systems	7. Control Methods for Marine Vessels	
Part III: Ship and Rig Applications	8. Course Autopilots 9. Autopilots with Roll Damping 10. Trajectory Tracking and Maneuvering Control 11. Positioning Systems		**Part IV: Underwater Vehicle Applications**	12. Propeller Control System Design 13. Decoupled Autopilot Design 14. 6 DOF Position and Attitude Control
Part V: Appendices	A. Nonlinear Stability Theory B. Numerical Methods C. Matlab GNC Toolbox			

Figure 1.8: Organization of the book.

Part I

Modeling of Marine Vessels

Chapter 2

Kinematics

The study of dynamics can be divided into two parts: *kinematics*, which treats only geometrical aspects of motion, and *dynamics*, which is the analysis of the forces causing the motion. In this chapter kinematics with application in terrestrial navigation is discussed, while dynamics is dealt with in Chapter 3.

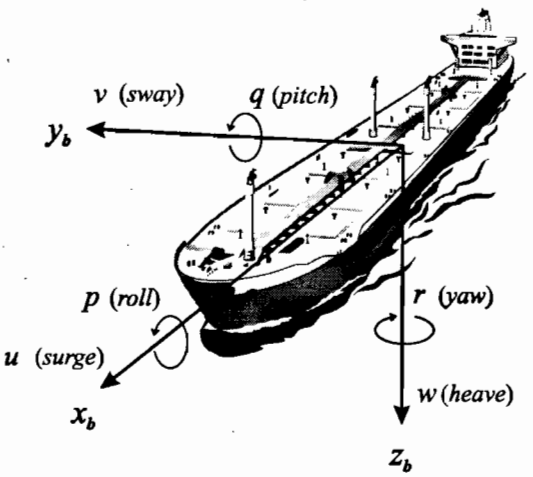


Figure 2.1: Motion variables for a marine vessel (SNAME 1950)

Table 2.1: The notation of SNAME (1950) for marine vessels.

DOF		forces and moments	linear and angular velocities	positions and Euler angles
1	motions in the x-direction (surge)	X	u	x
2	motions in the y-direction (sway)	Y	v	y
3	motions in the z-direction (heave)	Z	w	z
4	rotation about the x-axis (roll, heel)	K	p	ϕ
5	rotation about the y-axis (pitch, trim)	M	q	θ
6	rotation about the z-axis (yaw)	N	r	ψ

The interested reader is advised to consult Britting (1971), Maybeck (1979), Savage (1990), Forssell (1991), Lin (1992), Hofmann-Wellenhof *et al.* (1994), Parkinson and Spilker (1995), Titterton and Weston (1997), Farrell and Barth (1998), and Vik (2000) for a more detailed analyses of the navigation equations. The development of the kinematic equations of motion are found in Kane *et al.* (1983) and Hughes (1986). Both these references use spacecraft systems for illustration. An alternative derivation of the Euler angle representation in the context of ship steering is given by Abkowitz (1964). A more recent discussion of quaternions is found in Chou (1992). An analogy to robot manipulators is given by Craig (1989) or Sciavicco and Siciliano (1996). A detailed discussion of kinematics is found in Goldstein (1980), and Egeland and Gravdahl (2002).

6 DOF Marine Vessel Equations of Motion

The overall goal of Chapters 2 and 3 is to show that the marine vessel equations of motion can be written in a *vectorial setting* according to Fossen (1991):

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{2.1}$$

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

where the different matrices and vectors and their properties will be defined in the forthcoming sections. This model representation is used as a foundation for model-based control design and stability analysis in Parts II, III and IV.

Motion Variables

For marine vessels moving in 6 *degrees of freedom* (DOF), 6 independent coordinates are necessary to determine the position and orientation. The first three coordinates, and their time derivatives, correspond to the position and translational motion along the x-, y-, and z-axes, while the last 3 coordinates and their time derivatives are used to describe orientation and rotational motions. For marine vessels, the 6 different motion components are conveniently defined as, *surge, sway, heave, roll, pitch* and *yaw*; see Table 2.1.

and

$$\begin{aligned}\dot{V}_1 &= \tilde{\mathbf{x}}^\top \mathbf{K}_p \dot{\tilde{\mathbf{x}}} \\ &= \tilde{\mathbf{x}}^\top \mathbf{K}_p(-\boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}) \\ &= -\tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}^\top \mathbf{K}_p \tilde{\mathbf{x}} \end{aligned} \tag{7.342}$$

Step 2:

In the second step we choose a CLF motivated by *"pseudo" kinetic energy*, that is:

$$V_2 = \frac{1}{2}\mathbf{s}^\top \mathbf{M}\mathbf{s} + V_1, \qquad \mathbf{M} = \mathbf{M}^\top > 0 \tag{7.343}$$

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{M}\dot{\mathbf{s}} + \dot{V}_1 \\ &= \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}^\top \mathbf{K}_p \tilde{\mathbf{x}} \\ &= \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s} + \mathbf{K}_p\tilde{\mathbf{x}}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} \end{aligned} \tag{7.344}$$

Hence, we are ready to propose a control law e.g.:

$$\mathbf{B}\mathbf{u} = \mathbf{M}\dot{\mathbf{v}}_r + \mathbf{D}(\mathbf{v})\mathbf{v}_r + \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{K}_p\tilde{\mathbf{x}} - \mathbf{K}_d\mathbf{s}, \qquad \mathbf{K}_d > 0 \tag{7.345}$$

which results in:

$$\dot{V}_2 = -\mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}}$$

Since V_2 is positive definite and $\dot{V}_2$ is negative definite it follows from Theorem A.3 that the equilibrium point $(\tilde{\mathbf{x}}, \mathbf{s}) = (\mathbf{0}, \mathbf{0})$ is GES. Moreover, convergence of $\mathbf{s} \to \mathbf{0}$ and $\tilde{\mathbf{x}} \to \mathbf{0}$ implies that $\tilde{\mathbf{v}} \to \mathbf{0}$. When implementing the control law (7.345) it is assumed that $\mathbf{B}$ has an inverse:

$$\mathbf{B}^\dagger = \mathbf{B}^\top(\mathbf{B}\mathbf{B}^\top)^{-1} \tag{7.346}$$

or simply $\mathbf{B}^{-1}$ for the square case $r = n$.

Nonlinear Mass-Damper-Spring System with Actuator Dynamics

Consider the mass-damper-spring system of the previous section with actuator dynamics:

$$\dot{\mathbf{x}} = \boldsymbol{\nu} \tag{7.347}$$

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{K}(\mathbf{x})\mathbf{x} = \mathbf{B}\mathbf{u} \tag{7.348}$$

$$\mathbf{T}\dot{\mathbf{u}} + \mathbf{u} = \mathbf{u}_c \tag{7.349}$$

where $\mathbf{T} \in \mathbb{R}^{r\times r}$ is a diagonal matrix of actuator time constants and $\mathbf{u}_c \in \mathbb{R}^r$ is a vector of actuator commands. Instead of choosing the controller $\mathbf{u}$ in Step 2, $\mathbf{u}_c$ is treated as the control input to be specified in Step 3. Recall that:

$$\dot{V}_2 = \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s} + \mathbf{K}_p\tilde{\mathbf{x}}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} \tag{7.350}$$

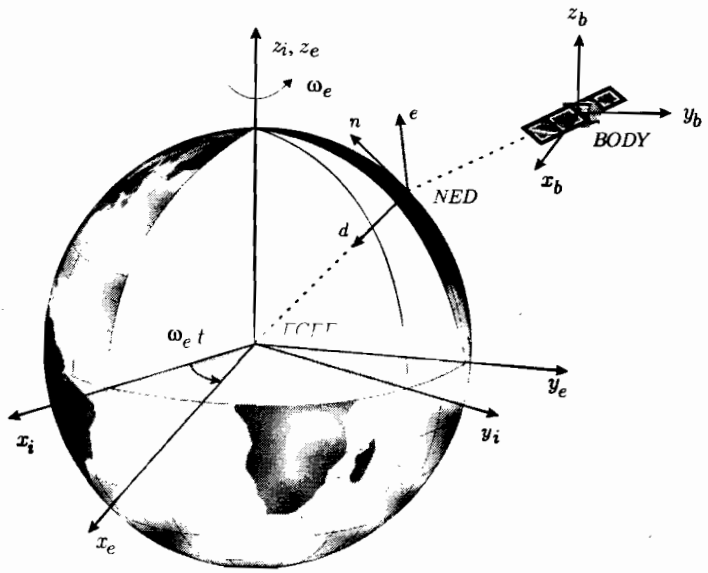


Figure 2.2: The Earth-centered Earth-fixed (ECEF) frame $x_e y_e z_e$ is rotating with angular rate ω_e with respect to an Earth-centered inertial (ECI) frame $x_i y_i z_i$ fixed in space.

2.1 Reference Frames

When analyzing the motion of marine vessels in 6 DOF, it is convenient to define two Earth-centered coordinate frames as indicated in Figure 2.2. In addition several geographic reference frames are needed to describe the motion of a marine vessel.

Earth-Centered Reference Frames

ECI (i-frame) The Earth-centered inertial frame (ECI) is an inertial frame for terrestrial navigation–i.e., a nonaccelerating reference frame in which Newton's laws of motion apply. The origin of the ECI coordinate frame $x_i y_i z_i$ is located at the center of the Earth with axes as shown in Figure 2.2.

ECEF (e-frame) The Earth-centered Earth-fixed reference frame $x_e y_e z_e$ has its origin fixed to the center of the Earth but the axes rotate relative to the inertial frame ECI which is fixed in space. The angular rate of rotation is $\omega_e = 7.2921 \cdot 10^{-5}$ rad/s. For marine vessels moving at relatively low speed, the Earth rotation can be neglected and hence the e-frame can be considered to be inertial. The e-frame is needed for global guidance, navigation and control e.g. to describe the motion and location of ships in transit between different continents.

Geographic Reference Frames

NED (n-frame) The *North-East-Down* coordinate system $x_n y_n z_n$ is defined relative to the Earth's reference ellipsoid (World Geodetic System 1984). This is the coordinate sys-

tem we refer to in our everyday life. It is usually defined as the tangent plane on the surface of the Earth moving with the vessel, but with axes pointing in different directions than the body-fixed axes of the vessel. For this system the x-axis points towards true *North*, the y-axis points towards *East* while the z-axis points *downwards* normal to the Earth's surface. The location of the n-frame relative to the e-frame is determined by using two angles l and μ denoting the *longitude* and *latitude*, respectively.

For marine vessels operating in a local area, approximately constant longitude and latitude, an Earth-fixed tangent plane on the surface is used for navigation. This is usually referred to as flat Earth navigation and it will for simplicity be denoted as the n-frame. For flat Earth navigation we will also assume that the n-frame is inertial such that Newton's laws still apply.

BODY (b-frame) The body-fixed reference frame $x_b y_b z_b$ is a moving coordinate frame which is fixed to the vessel. The position and orientation of the vessel are described relative to the inertial reference frame (approximated by the e- or n-frames for marine vessels) while the linear and angular velocities of the vessel should be expressed in the body-fixed coordinate system. The origin O of the body-fixed frame is usually chosen to coincide with the center of gravity (CG) when CG is in the principal plane of symmetry, or at any other convenient point if this is not the case. For marine vessels, the body axes x_b, y_b and z_b are chosen to coincide with the *principal axes of inertia*, and they are usually defined as (see Figure 2.1):

- x_b - longitudinal axis (directed from aft to fore)
- y_b - transversal axis (directed to starboard)
- z_b - normal axis (directed from top to bottom)

In addition to the body-fixed coordinate system $x_b y_b z_b$, it is convenient to define two other body-fixed coordinate systems when performing hydrodynamic computations. These systems are referred to as *stability and current axes*; see Section 2.4.

Vectorial Definitions

For marine vessels the following notation will be adopted for the linear and angular velocities when decomposed in a reference frame:

$$\begin{aligned} \mathbf{v}_o^n &= \text{linear velocity of point } O \text{ decomposed in frame } \boldsymbol{n} \\ \boldsymbol{\omega}_{eb}^n &= \text{angular velocity of frame } b \text{ with respect to frame } e \\ &\quad \text{decomposed in frame } n \end{aligned}$$

The different quantities in Table 2.1, as defined by SNAME (1950), can now be conveniently expressed in a vectorial setting according to:

ECEF position:	$\mathbf{p}^e = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3$		Longitude and latitude	$\Psi = \begin{bmatrix} l \\ \mu \end{bmatrix} \in \mathcal{S}^2$	
NED position:	$\mathbf{p}^n = \begin{bmatrix} n \\ e \\ d \end{bmatrix} \in \mathbb{R}^3$		Attitude (Euler angles)	$\Theta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \in \mathcal{S}^3$	
Body-fixed linear velocity	$\mathbf{v}_o^b = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \in \mathbb{R}^3$		Body-fixed angular velocity	$\boldsymbol{\omega}_{nb}^b = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \in \mathbb{R}^3$	
Body-fixed force:	$\mathbf{f}_o^b = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \in \mathbb{R}^3$		Body-fixed moment	$\mathbf{m}_o^b = \begin{bmatrix} K \\ M \\ N \end{bmatrix} \in \mathbb{R}^3$	

where $\mathbb{R}^3$ is the *Euclidean space* of dimension 3 and $\mathcal{S}^2$ denotes a *torus* of dimension 2 (shape of a donut) implying that there are two angles defined on the interval $[0, 2\pi]$. In the 3-dimensional case the set is denoted as $\mathcal{S}^3$. Hence, the general motion of a marine vessel in 6 DOF is described by the following vectors (Fossen 1991):

$$\boldsymbol{\eta} = \begin{bmatrix} \mathbf{p}^e \\ \Theta \end{bmatrix}, \quad \boldsymbol{\nu} = \begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix}, \quad \boldsymbol{\tau} = \begin{bmatrix} \mathbf{f}_o^b \\ \mathbf{m}_o^b \end{bmatrix} \tag{2.3}$$

where $\boldsymbol{\eta} \in \mathbb{R}^3 \times \mathcal{S}^3$ denotes the position and orientation vector where the position vector $\mathbf{p}^e \in \mathbb{R}^3$ is decomposed in ECEF, $\Theta \in \mathcal{S}^3$ is a vector of Euler angles, $\boldsymbol{\nu} \in \mathbb{R}^6$ denotes the linear and angular velocity vectors which are decomposed in the body-fixed reference frame and $\boldsymbol{\tau} \in \mathbb{R}^6$ is used to describe the forces and moments acting on the vessel in the body-fixed frame.

In many applications, like flat Earth navigation, the position vector $\mathbf{p}^n \in \mathbb{R}^3$ is decomposed in NED coordinates instead of using $\mathbf{p}^e \in \mathbb{R}^3$. Orientation will be represented by means of the Euler angles Θ or the quaternions $\mathbf{q} \in \mathbb{R}^4$. In the next sections, the kinematic equations relating the body-fixed, NED and ECEF reference frames will be presented.

2.2 Transformations between BODY and NED

The rotation matrix $\mathbf{R}$ between two frames a and b is denoted as $\mathbf{R}_b^a$, and it is an element in $SO(3)$, that is the *special orthogonal group of order 3:*

$$SO(3) = \left\{ \mathbf{R} | \mathbf{R} \in \mathbb{R}^{3\times 3}, \quad \mathbf{R} \text{ is orthogonal and } \det \mathbf{R} = 1 \right\} \tag{2.4}$$

The group $SO(3)$ is a subset of all *orthogonal matrices of order 3*–i.e., $SO(3) \subset O(3)$ where $O(3)$ is defined as:

$$O(3) = \left\{ \mathbf{R} | \mathbf{R} \in \mathbb{R}^{3\times 3}, \quad \mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top \mathbf{R} = \mathbf{I} \right\} \tag{2.5}$$

Rotation matrices are useful when deriving the kinematic equations of motion for a marine vessel. As a consequence of (2.4) and (2.5), the following properties can be stated:

Property 2.1 (Rotation Matrix)
A rotation matrix $\mathbf{R} \in SO(3)$ *satisfies:*

$$\mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top\mathbf{R} = \mathbf{I}, \qquad \det \mathbf{R} = 1$$

which implies that $\mathbf{R}$ *is orthogonal. Hence, the inverse rotation matrix is given by:* $\mathbf{R}^{-1} = \mathbf{R}^\top$.

In this book, the following notation is adopted when transforming a vector from one coordinate frame to another:

$$\boldsymbol{\nu}^{\text{to}} = \boldsymbol{R}^{\text{to}}_{\text{from}}\boldsymbol{\nu}^{\text{from}} \tag{2.6}$$

Here $\boldsymbol{\nu}^{\text{from}} \in \mathbb{R}^3$ denotes a velocity vector which can be transformed to a new reference frame by applying the rotation matrix $\mathbf{R}^{\text{to}}_{\text{from}}$. The result is the vector $\boldsymbol{\nu}^{\text{to}} \in \mathbb{R}^3$.

A frequently used rotation matrix in guidance, navigation and control is the rotation matrix $\mathbf{R}^n_b$ between the *n*- and *b*-frames. When deriving the expression for $\mathbf{R}^n_b$ we will make use of the following matrix properties:

Definition 2.1 (Skew-Symmetry of a Matrix)
A matrix $\mathbf{S} \in SS(n)$, *that is the set of skew-symmetric matrices of order n, is said to be skew-symmetrical if:*

$$\mathbf{S} = -\mathbf{S}^\top.$$

This implies that the off-diagonal elements of $\mathbf{S}$ *satisfy* $s_{ij} = -s_{ji}$ *for* $i \neq j$ *while the diagonal elements are zero.*

Definition 2.2 (Cross Product Operator)
The vector cross product $\times$ *is defined by:*

$$\boldsymbol{\lambda} \times \mathbf{a} := \mathbf{S}(\boldsymbol{\lambda})\mathbf{a} \tag{2.7}$$

where $\mathbf{S} \in SS(3)$ *is defined as:*

$$\mathbf{S}(\boldsymbol{\lambda}) = -\mathbf{S}^\top(\boldsymbol{\lambda}) = \begin{bmatrix} 0 & -\lambda_3 & \lambda_2 \\ \lambda_3 & 0 & -\lambda_1 \\ -\lambda_2 & \lambda_1 & 0 \end{bmatrix}, \quad \boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} \tag{2.8}$$

Matlab:
The cross product operator is included in the GNC toolbox as `Smtrix.m`. Hence, the cross product $\mathbf{b} = \mathbf{S}(\boldsymbol{\lambda})\mathbf{a}$ can be computed as:

```
S = Smtrx(lambda)
b = S*a
```

Definition 2.3 (Simple Rotation)
The motion of a rigid body or reference frame $\mathcal{B}$ *relative to a rigid body or reference frame* $\mathcal{A}$ *is called a simple rotation of* $\mathcal{B}$ *in* $\mathcal{A}$ *if there exists a line L, called an axis of rotation, whose orientation relative to both* $\mathcal{A}$ *and* $\mathcal{B}$ *remains unaltered throughout the motion.*

Based on this definition Euler stated the following theorem for rotation of two rigid bodies or reference frames (Euler 1776).

Theorem 2.1 (Euler's Theorem on Rotation)
Every change in the relative orientation of two rigid bodies or reference frames $\mathcal{A}$ and $\mathcal{B}$ can be produced by means of a simple rotation of $\mathcal{B}$ in $\mathcal{A}$.

Let $\mathbf{v}_o^b$ be a vector fixed in BODY and $\mathbf{v}_o^n$ be a vector fixed in NED. Hence, the vector $\mathbf{v}_o^n$ can be expressed in terms of the vector $\mathbf{v}_o^b$, the unit vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^\top, |\boldsymbol{\lambda}| = 1$, parallel to the axis of rotation and β the angle NED is rotated. This rotation is described by (see Hughes 1986, Kane *et al.* 1983):

$$\mathbf{v}_o^n = \mathbf{R}_b^n \mathbf{v}_o^b, \quad \mathbf{R}_b^n := \mathbf{R}_{\boldsymbol{\lambda},\beta} \tag{2.9}$$

Here, $\mathbf{R}_{\boldsymbol{\lambda},\beta}$ is the rotation matrix corresponding to a rotation β about the $\boldsymbol{\lambda}$-axis:

$$\mathbf{R}_{\boldsymbol{\lambda},\beta} = \mathbf{I}_{3\times 3} + \sin\beta\, \mathbf{S}(\boldsymbol{\lambda}) + (1-\cos\beta)\, \mathbf{S}^2(\boldsymbol{\lambda}) \tag{2.10}$$

where $\mathbf{I}_{3\times 3}$ is the identity matrix and $\mathbf{S}(\boldsymbol{\lambda})$ is the skew-symmetric matrix according to Definition 2.2. Consequently, $\mathbf{S}^2(\boldsymbol{\lambda}) = \boldsymbol{\lambda}\boldsymbol{\lambda}^\top - \mathbf{I}_{3\times 3}$ since $\boldsymbol{\lambda}$ is a unit vector.

Expanding (2.10) yields the following expressions for the matrix elements R_{ij}:

$$\begin{aligned}
R_{11} &= (1-\cos\beta)\,\lambda_1^2 + \cos\beta \\
R_{22} &= (1-\cos\beta)\,\lambda_2^2 + \cos\beta \\
R_{33} &= (1-\cos\beta)\,\lambda_3^2 + \cos\beta \\
R_{12} &= (1-\cos\beta)\,\lambda_1\lambda_2 - \lambda_3 \sin\beta \\
R_{21} &= (1-\cos\beta)\,\lambda_2\lambda_1 + \lambda_3 \sin\beta \\
R_{23} &= (1-\cos\beta)\,\lambda_2\lambda_3 - \lambda_1 \sin\beta \\
R_{32} &= (1-\cos\beta)\,\lambda_3\lambda_2 + \lambda_1 \sin\beta \\
R_{31} &= (1-\cos\beta)\,\lambda_3\lambda_1 - \lambda_2 \sin\beta \\
R_{13} &= (1-\cos\beta)\,\lambda_1\lambda_3 + \lambda_2 \sin\beta
\end{aligned} \tag{2.11}$$

2.2.1 Euler Angle Transformation

The Euler angles: roll (ϕ), pitch (θ) and yaw (ψ) can now be used to decompose the body-fixed velocity vector $\mathbf{v}_o^b$ in the NED reference frame. Let $\mathbf{R}_b^n(\boldsymbol{\Theta}) : \mathcal{S}^3 \to SO(3)$ denote the Euler angle rotation matrix with argument $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$. Hence:

$$\mathbf{v}_o^n = \mathbf{R}_b^n(\boldsymbol{\Theta}) \mathbf{v}_o^b \tag{2.12}$$

Principal Rotations

The principal rotation matrices (one axis rotations) can be obtained by setting $\boldsymbol{\lambda} = [1,0,0]^\top$, $\boldsymbol{\lambda} = [0,1,0]^\top$ and $\boldsymbol{\lambda} = [0,0,1]^\top$ corresponding to the x, y and z axes, and $\beta = \phi, \beta = \theta$ and $\beta = \psi$, respectively, in the formula for $\mathbf{R}_{\boldsymbol{\lambda},\beta}$ given by (2.10).

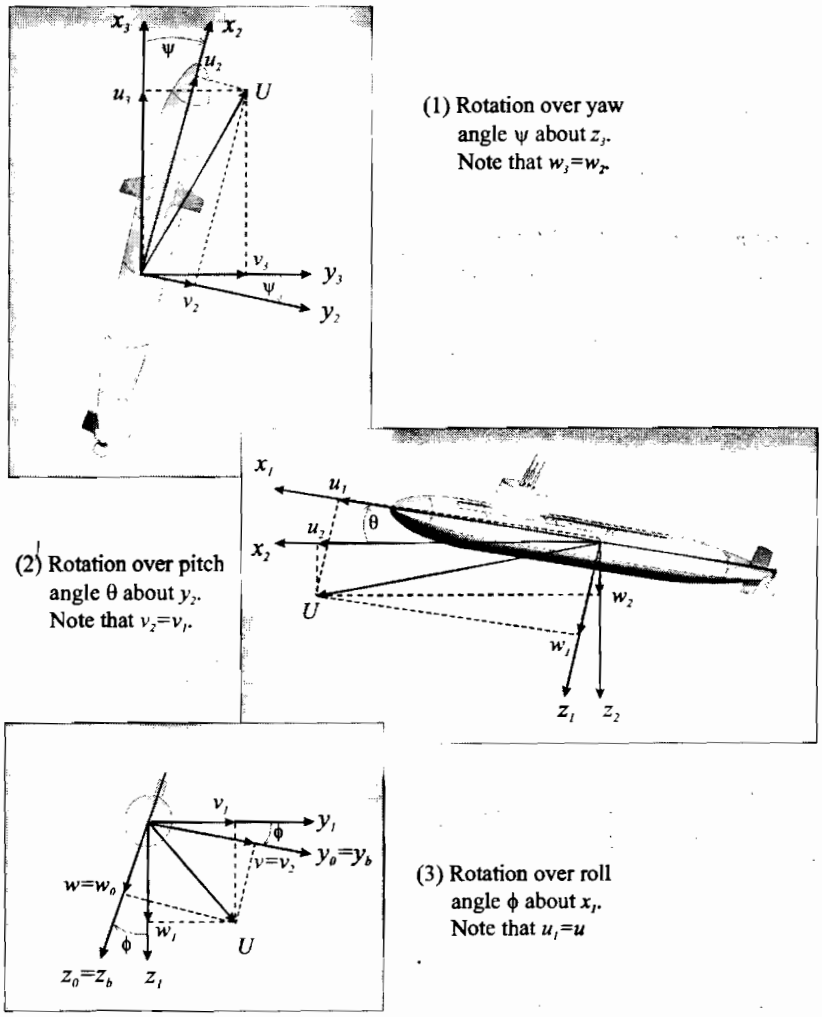


Figure 2.3: Euler angle rotation sequence (*zyx*-convention). The submarine is rotated from the *n*-frame to the *b*-frame by using 3 principal rotations.

This yields:

$$\mathbf{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ 0 & \mathrm{s}\phi & \mathrm{c}\phi \end{bmatrix}, \quad \mathbf{R}_{y,\theta} = \begin{bmatrix} \mathrm{c}\theta & 0 & \mathrm{s}\theta \\ 0 & 1 & 0 \\ -\mathrm{s}\theta & 0 & \mathrm{c}\theta \end{bmatrix}, \quad \mathbf{R}_{z,\psi} = \begin{bmatrix} \mathrm{c}\psi & -\mathrm{s}\psi & 0 \\ \mathrm{s}\psi & \mathrm{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

where $\mathrm{s}\cdot = \sin(\cdot)$ and $\mathrm{c}\cdot = \cos(\cdot)$.

Linear Velocity Transformation

It is customary to describe $\mathbf{R}_b^n(\boldsymbol{\Theta})$ by three *principal* rotations about the z, y and x axes (zyx-convention). Note that the order in which these rotations is carried out is not arbitrary. In guidance, navigation and control applications it is common to use the zyx-convention from the n-frame to the b-frame specified in terms of the Euler angles ϕ, θ and ψ for the rotations. This matrix is denoted $\mathbf{R}_n^b(\boldsymbol{\Theta}) = \mathbf{R}_b^n(\boldsymbol{\Theta})^\top$. The matrix transpose implies that the same result is obtained by transforming a vector from the b-frame to the n-frame–i.e., by reversing the order of the transformation. This rotation sequence is mathematically equivalent to:

$$\mathbf{R}_b^n(\boldsymbol{\Theta}) := \mathbf{R}_{z,\psi}\mathbf{R}_{y,\theta}\mathbf{R}_{x,\phi} \tag{2.14}$$

and the inverse transformation is then written (zyx-convention):

$$\mathbf{R}_b^n(\boldsymbol{\Theta})^{-1} = \mathbf{R}_n^b(\boldsymbol{\Theta}) = \mathbf{R}_{x,\phi}^\top \mathbf{R}_{y,\theta}^\top \mathbf{R}_{z,\psi}^\top \tag{2.15}$$

where we have used the result of Property 2.1. This can also be seen by studying Figure 2.3.

Let $x_3y_3z_3$ be the coordinate system obtained by translating the NED coordinate system $x_ny_nz_n$ parallel to itself until its origin coincides with the origin of the body-fixed coordinate system. The coordinate system $x_3y_3z_3$ is rotated a *yaw* angle ψ about the z_3-axis. This yields the coordinate system $x_2y_2z_2$. The coordinate system $x_2y_2z_2$ is rotated a *pitch* angle θ about the y_2-axis. This yields the coordinate system $x_1y_1z_1$. Finally, the coordinate system $x_1y_1z_1$ is rotated a *roll* angle ϕ about the x_1-axis. This yields the body-fixed coordinate system $x_by_bz_b$.

Expanding (2.14), yields:

$$\mathbf{R}_b^n(\boldsymbol{\Theta}) = \begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & -\mathrm{s}\psi\mathrm{c}\phi + \mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\phi & \mathrm{s}\psi\mathrm{s}\phi + \mathrm{c}\psi\mathrm{c}\phi\mathrm{s}\theta \\ \mathrm{s}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{c}\phi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi & -\mathrm{c}\psi\mathrm{s}\phi + \mathrm{s}\theta\mathrm{s}\psi\mathrm{c}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} \tag{2.16}$$

Matlab:
The rotation matrix is implemented in the GNC toolbox as:

```
R = Rzyx(phi,theta,psi)
```

For small angles $\delta\phi, \delta\theta$ and $\delta\psi$ the expression (2.16) simplifies to:

$$\mathbf{R}_b^n(\delta\boldsymbol{\Theta}) \approx \mathbf{I}_{3\times 3} + \mathbf{S}(\delta\boldsymbol{\Theta}) = \begin{bmatrix} 1 & -\delta\psi & \delta\theta \\ \delta\psi & 1 & -\delta\phi \\ -\delta\theta & \delta\phi & 1 \end{bmatrix} \tag{2.17}$$

which is quite useful when applying linear theory.

The body-fixed velocity vector $\mathbf{v}_o^b$ are now decomposed in the NED reference frame as:

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{v}_o^b \tag{2.18}$$

where $\dot{\mathbf{p}}^n$ is the NED velocity vector. Expanding (2.18), yields:

$$\begin{aligned}
\dot{n} &= u\cos\psi\cos\theta + v(\cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi) \\
&\quad + w(\sin\psi\sin\phi + \cos\psi\cos\phi\sin\theta) \qquad (2.19)\\
\dot{e} &= u\sin\psi\cos\theta + v(\cos\psi\cos\phi + \sin\phi\sin\theta\sin\psi) \\
&\quad + w(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) \qquad (2.20)\\
\dot{d} &= -u\sin\theta + v\cos\theta\sin\phi + w\cos\theta\cos\phi \qquad (2.21)
\end{aligned}$$

The inverse velocity transformation is obtained by Definition 2.1 as:

$$\mathbf{v}_o^b = \mathbf{R}_b^n(\boldsymbol{\Theta})^{-1}\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\boldsymbol{\Theta})^\top\dot{\mathbf{p}}^n \qquad (2.22)$$

Example 2.1 (Numercial Computation of Position Trajectory)
The flight path or position trajectory of the vessel $\mathbf{p}^n$ *relative to the NED coordinate system is found by numerical integration of (2.18), for instance by using* Euler integration *(see Appendix B.2.1):*

$$\mathbf{p}^n(k+1) = \mathbf{p}^n(k) + h\mathbf{R}_b^n(\boldsymbol{\Theta}(k))\mathbf{v}_o^b(k) \qquad (2.23)$$

where $h > 0$ *is the sampling time and* k *is the sample index.*

Angular Velocity Transformation

The body-fixed angular velocity vector $\boldsymbol{\omega}_{nb}^b = [p, q, r]^\top$ and the Euler rate vector $\dot{\boldsymbol{\Theta}} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$ are related through a transformation matrix $\mathbf{T}_\Theta(\boldsymbol{\Theta})$ according to:

$$\dot{\boldsymbol{\Theta}} = \mathbf{T}_\Theta(\boldsymbol{\Theta})\boldsymbol{\omega}_{nb}^b \qquad (2.24)$$

It should be noted that the angular body velocity vector $\boldsymbol{\omega}_{nb}^b = [p, q, r]^\top$ cannot be integrated directly to obtain actual angular coordinates. This is due to the fact that $\int_0^t \boldsymbol{\omega}_{nb}^b(\tau)d\tau$ does not have any immediate physical interpretation; however, the vector $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$ does represent proper generalized coordinates. The transformation matrix $\mathbf{T}_\Theta(\boldsymbol{\Theta})$ can be derived in several ways, e.g.:

$$\boldsymbol{\omega}_{nb}^b = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{R}_{x,\phi}^\top \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{R}_{x,\phi}^\top\mathbf{R}_{y,\theta}^\top \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} := \mathbf{T}_\Theta^{-1}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}} \qquad (2.25)$$

This relationship is verified by inspection of Figure 2.3. Expanding (2.25) yields:

$$\mathbf{T}_\Theta^{-1}(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & 0 & -\mathrm{s}\theta \\ 0 & \mathrm{c}\phi & \mathrm{c}\theta\mathrm{s}\phi \\ 0 & -\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} \quad \Longrightarrow \quad \mathbf{T}_\Theta(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & \mathrm{s}\phi\mathrm{t}\theta & \mathrm{c}\phi\mathrm{t}\theta \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ 0 & \mathrm{s}\phi/\mathrm{c}\theta & \mathrm{c}\phi/\mathrm{c}\theta \end{bmatrix} \qquad (2.26)$$

where $\mathrm{s}\cdot = \sin(\cdot)$, $\mathrm{c}\cdot = \cos(\cdot)$ and $\mathrm{t}\cdot = \tan(\cdot)$. Expanding (2.24) yields the Euler angle attitude equations in component form:

$$\dot{\theta} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{2.27}$$

$$\dot{\phi} = q\cos\phi - r\sin\phi \tag{2.28}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta}, \quad \theta \neq \pm 90^o \tag{2.29}$$

Notice that $\mathbf{T}_\Theta(\boldsymbol{\Theta})$ is undefined for a pitch angle of $\theta = \pm\ 90^o$ and that $\mathbf{T}_\Theta(\boldsymbol{\Theta})$ does not satisfy Property 2.1. Consequently, $\mathbf{T}_\Theta^{-1}(\boldsymbol{\Theta}) \neq \mathbf{T}_\Theta^\top(\boldsymbol{\Theta})$. For surface vessels this is not a problem whereas both underwater vehicles and aircraft may operate close to this singularity. In this case, the kinematic equations can be described by two Euler angle representations with different singularities and the singular point can be avoided by switching between these. Another possibility is to use a quaternion representation; see Section 2.2.2.

For small angles $\delta\phi, \delta\theta$, and $\delta\psi$ the transformation matrix $\mathbf{T}_\Theta(\boldsymbol{\Theta})$ simplifies to:

$$\mathbf{T}_\Theta(\delta\boldsymbol{\Theta}) \approx \begin{bmatrix} 1 & 0 & \delta\theta \\ 0 & 1 & -\delta\phi \\ 0 & \delta\phi & 1 \end{bmatrix} \tag{2.30}$$

The differential equation for the rotation matrix is given by Theorem 2.2.

Theorem 2.2 (Rotation Matrix Differential Equation)
The differential equation for the rotation matrix between the BODY and NED reference frames is:

$$\dot{\mathbf{R}}_b^n = \mathbf{R}_b^n \mathbf{S}(\boldsymbol{\omega}_{nb}^b) \tag{2.31}$$

where

$$\mathbf{S}(\boldsymbol{\omega}_{nb}^b) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \tag{2.32}$$

This can be written in component form as 9 differential equations:

$$\begin{bmatrix} \dot{R}_{11} & \dot{R}_{12} & \dot{R}_{13} \\ \dot{R}_{21} & \dot{R}_{22} & \dot{R}_{23} \\ \dot{R}_{31} & \dot{R}_{23} & \dot{R}_{33} \end{bmatrix} = \begin{bmatrix} R_{12}r - R_{13}q & -R_{11}r + R_{13}p & R_{11}q - R_{12}p \\ R_{22}r - R_{23}q & -R_{21}r + R_{23}p & R_{21}q - R_{22}p \\ R_{23}r - R_{33}q & -R_{31}r + R_{33}p & R_{31}q - R_{23}p \end{bmatrix} \tag{2.33}$$

Proof. For a small time increment Δt the rotation matrix $\mathbf{R}_b^n$ satisfies:

$$\mathbf{R}_b^n(t + \Delta t) \approx \mathbf{R}_b^n(t)\mathbf{R}_b^n(\Delta t) \tag{2.34}$$

since $\sin(\Delta t) \approx \Delta t$ and $\cos(\Delta t) \approx 1$. Assume that after time $t + \Delta t$ there has been an infinitesimal increment $\Delta\beta$ in the rotation angle. From (2.10) we have:

$$\begin{aligned} \mathbf{R}_b^n(\Delta t) &= \mathbf{I}_{3\times 3} + \sin(\Delta\beta)\,\mathbf{S}(\boldsymbol{\lambda}) + (1 - \cos(\Delta\beta))\,\mathbf{S}^2(\boldsymbol{\lambda}) \\ &\approx \mathbf{I}_{3\times 3} + \Delta\beta\,\mathbf{S}(\boldsymbol{\lambda}) \end{aligned} \tag{2.35}$$

since $\sin(\Delta\beta) \approx \Delta\beta$ and $\cos(\Delta\beta) \approx 1$. From (2.34), we get:

$$\mathbf{R}_b^n(t + \Delta t) = \mathbf{R}_b^n(t)\,[\mathbf{I}_{3\times 3} + \Delta\beta\,\mathbf{S}(\boldsymbol{\lambda})] \tag{2.36}$$

Defining the vector $\Delta\boldsymbol{\beta}^b := \Delta\beta\boldsymbol{\lambda}$, the time derivative of $\mathbf{R}_b^n$ is found as:

$$\begin{aligned}
\dot{\mathbf{R}}_b^n(t) &= \lim_{\Delta t\to 0} \frac{\mathbf{R}_b^n(t+\Delta t) - \mathbf{R}_b^n(t)}{\Delta t} \\
&= \lim_{\Delta t\to 0} \frac{\mathbf{R}_b^n(t)\,\Delta\beta\,\mathbf{S}(\boldsymbol{\lambda})}{\Delta t} \\
&= \lim_{\Delta t\to 0} \frac{\mathbf{R}_b^n(t)\,\mathbf{S}(\Delta\boldsymbol{\beta}^b)}{\Delta t} \\
&= \mathbf{R}_b^n(t)\,\mathbf{S}(\boldsymbol{\omega}_{nb}^b)
\end{aligned} \tag{2.37}$$

where $\boldsymbol{\omega}_{nb}^b = \lim_{\Delta t\to 0}(\Delta\boldsymbol{\beta}^b/\Delta t)$. ■

6 DOF Kinematic Equations

Summarizing the results from this section, the 6 DOF kinematic equations can be expressed in vector form as:

$$\begin{gathered}
\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \\
\Updownarrow \\
\begin{bmatrix} \dot{\mathbf{p}}^n \\ \dot{\boldsymbol{\Theta}} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b^n(\boldsymbol{\Theta}) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{T}_\Theta(\boldsymbol{\Theta}) \end{bmatrix} \begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix}
\end{gathered} \tag{2.38}$$

where $\boldsymbol{\eta} \in \mathbb{R}^3 \times \mathcal{S}^3$ and $\boldsymbol{\nu} \in \mathbb{R}^6$.

Matlab:
The transformation matrix $\mathbf{J}$ and its elements $\mathbf{J}_1 = \mathbf{R}_b^n(\boldsymbol{\Theta})$ and $\mathbf{J}_2 = \mathbf{T}_\Theta(\boldsymbol{\Theta})$ can be computed by using the GNC toolbox command:

```
[J,J1,J2] = eulerang(phi,theta,psi)
```

The differential equations are then found by:

```
p_dot     = J1*v_o
theta_dot = J2*w_nb
```

Alternatively, we can write the equations (2.38) in component form as:

$$\begin{aligned}
\dot{n} &= u\cos\psi\cos\theta + v(\cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi) \\
&\quad + w(\sin\psi\sin\phi + \cos\psi\cos\phi\sin\theta) && (2.39)\\
\dot{e} &= u\sin\psi\cos\theta + v(\cos\psi\cos\phi + \sin\phi\sin\theta\sin\psi) \\
&\quad + w(\sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi) && (2.40)\\
\dot{d} &= -u\sin\theta + v\cos\theta\sin\phi + w\cos\theta\cos\phi && (2.41)\\
\dot{\theta} &= p + q\sin\phi\tan\theta + r\cos\phi\tan\theta && (2.42)\\
\dot{\phi} &= q\cos\phi - r\sin\phi && (2.43)\\
\dot{\psi} &= q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta}, \quad \theta \neq \pm 90^o && (2.44)
\end{aligned}$$

3 DOF Model for Surface Vessels

An attractive simplification of (2.38) is the 3 DOF (surge, sway and yaw) representation for surface vessels. This is based on the assumption that ϕ and θ are small which is a good approximation for most conventional ships and rigs. Hence, $\mathbf{R}_b^n(\Theta) = \mathbf{R}_{z,\psi}\mathbf{R}_{y,\theta}\mathbf{R}_{x,\phi} \approx \mathbf{R}_{z,\psi}$ and $\mathbf{T}_\Theta(\Theta) \approx \mathbf{I}_{3\times 3}$. Neglecting the elements corresponding to heave, roll, and pitch finally yields:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{2.45}$$

where $\mathbf{R}(\psi) = \mathbf{R}_{z,\psi}$ while $\boldsymbol{\nu} = [u, v, r]^\top$ and $\boldsymbol{\eta} = [x, y, \psi]^\top$.

2.2.2 Unit Quaternions

An alternative to the Euler angle representation is a four-parameter method based on *unit quaternions* or *Euler parameters*. The main motivation for using four parameters is to avoid the representation singularity of the Euler angles.

A quaternion $\mathbf{q}$ is defined as a complex number (Chou 1992) with one real part η and three imaginary parts given by the vector:

$$\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \varepsilon_3]^\top \tag{2.46}$$

A unit quaternion satisfies $\mathbf{q}^\top\mathbf{q} = 1$. The set Q of unit quaternions is therefore defined as:

$$Q = \left\{\mathbf{q} | \mathbf{q}^\top\mathbf{q} = 1, \mathbf{q} = [\eta, \boldsymbol{\varepsilon}^\top]^\top, \quad \boldsymbol{\varepsilon} \in \mathbb{R}^3 \text{ and } \eta \in \mathbb{R}\right\} \tag{2.47}$$

The motion of the body-fixed reference frame relative to the inertial frame will now be defined in terms of unit quaternions.

Unit Quaternions

From (2.10) it is seen that:

$$\mathbf{R}_{\beta,\boldsymbol{\lambda}} = \mathbf{I}_{3\times 3} + \sin\beta\,\mathbf{S}(\boldsymbol{\lambda}) + (1 - \cos\beta)\,\mathbf{S}^2(\boldsymbol{\lambda}) \tag{2.48}$$

The real and imaginary parts of the unit quaternions are defined as (Chou 1992):

$$\eta = \cos\frac{\beta}{2} \tag{2.49}$$

$$\varepsilon = [\varepsilon_1, \varepsilon_2, \varepsilon_3]^\top = \boldsymbol{\lambda}\sin\frac{\beta}{2} \tag{2.50}$$

where $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^\top$ is a unit vector satisfying:

$$\boldsymbol{\lambda} = \pm\frac{\boldsymbol{\varepsilon}}{\sqrt{\boldsymbol{\varepsilon}^\top\boldsymbol{\varepsilon}}}; \qquad \sqrt{\boldsymbol{\varepsilon}^\top\boldsymbol{\varepsilon}} \neq 0. \tag{2.51}$$

Consequently, the unit quaternions can be expressed in the form:

$$\mathbf{q} = \begin{bmatrix} \eta \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = \begin{bmatrix} \cos\frac{\beta}{2} \\ \boldsymbol{\lambda}\sin\frac{\beta}{2} \end{bmatrix} \in Q, \qquad 0 \leq \beta \leq 2\pi. \tag{2.52}$$

This parameterization implies that the unit quaternions satisfy the constraint $\mathbf{q}^\top\mathbf{q} = 1$, i.e.:

$$\eta^2 + \varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 = 1 \tag{2.53}$$

From (2.48) with (2.49) and (2.50), the following coordinate transformation matrix for the unit quaternions is obtained:

$$\mathbf{R}_b^n(\mathbf{q}) := \mathbf{R}_{\eta,\boldsymbol{\varepsilon}} = \mathbf{I}_{3\times 3} + 2\eta\mathbf{S}(\boldsymbol{\varepsilon}) + 2\mathbf{S}^2(\boldsymbol{\varepsilon}). \tag{2.54}$$

Linear Velocity Transformation

The transformation relating the linear velocity vector in an inertial reference frame to a velocity in the body-fixed reference frame can now be expressed as:

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\mathbf{q})\mathbf{v}_o^b \tag{2.55}$$

where

$$\mathbf{R}_b^n(\mathbf{q}) = \begin{bmatrix} 1-2(\varepsilon_2^2+\varepsilon_3^2) & 2(\varepsilon_1\varepsilon_2-\varepsilon_3\eta) & 2(\varepsilon_1\varepsilon_3+\varepsilon_2\eta) \\ 2(\varepsilon_1\varepsilon_2+\varepsilon_3\eta) & 1-2(\varepsilon_1^2+\varepsilon_3^2) & 2(\varepsilon_2\varepsilon_3-\varepsilon_1\eta) \\ 2(\varepsilon_1\varepsilon_3-\varepsilon_2\eta) & 2(\varepsilon_2\varepsilon_3+\varepsilon_1\eta) & 1-2(\varepsilon_1^2+\varepsilon_2^2) \end{bmatrix} \tag{2.56}$$

Expanding (2.55), yields:

$$\dot{n} = u(1-2\varepsilon_2^2-2\varepsilon_3^2) + 2v(\varepsilon_1\varepsilon_2-\varepsilon_3\eta) + 2w(\varepsilon_1\varepsilon_3+\varepsilon_2\eta) \tag{2.57}$$

$$\dot{e} = 2u(\varepsilon_1\varepsilon_2+\varepsilon_3\eta) + v(1-2\varepsilon_1^2-2\varepsilon_3^2) + 2w(\varepsilon_2\varepsilon_3-\varepsilon_1\eta) \tag{2.58}$$

$$\dot{d} = 2u(\varepsilon_1\varepsilon_3-\varepsilon_2\eta) + 2v(\varepsilon_2\varepsilon_3+\varepsilon_1\eta) + w(1-2\varepsilon_1^2-2\varepsilon_2^2) \tag{2.59}$$

As for the Euler angle representation, Property 2.1 implies that the inverse transformation matrix satisfies $\mathbf{R}_b^n(\mathbf{q})^{-1} = \mathbf{R}_b^n(\mathbf{q})^\top$.

Matlab:
The quaternion rotation matrix is easily computed by using the GNC toolbox commands:

```
q = [eta,eps1,eps2,eps3]
R = Rquat(q)
```

Notice that $\mathbf{q}^\top\mathbf{q} = 1$ must be true for `Rquat.m` to return a solution. One way to ensure this is to use the transformation:

```
q = euler2q(phi,theta,psi)   \
```

transforming the three Euler angles ϕ, θ and ψ to the unit quaternion vector $\mathbf{q}$, see Section 2.2.3 for details.

Angular Velocity Transformation

The angular velocity transformation can be derived by substituting the expressions for R_{ij} from (2.56) into the differential equation $\dot{\mathbf{R}}_b^n = \mathbf{R}_b^n \mathbf{S}(\omega_{nb}^b)$, see Theorem 2.2. Some calculations yield:

$$\dot{\mathbf{q}} = \mathbf{T}_q(\mathbf{q})\omega_{nb}^b \tag{2.60}$$

where:

$$\mathbf{T}_q(\mathbf{q}) = \frac{1}{2}\begin{bmatrix} -\varepsilon_1 & -\varepsilon_2 & -\varepsilon_3 \\ \eta & -\varepsilon_3 & \varepsilon_2 \\ \varepsilon_3 & \eta & -\varepsilon_1 \\ -\varepsilon_2 & \varepsilon_1 & \eta \end{bmatrix}, \qquad \mathbf{T}_q^\top(\mathbf{q})\mathbf{T}_q(\mathbf{q}) = \frac{1}{4}\mathbf{I}_{3\times 3} \tag{2.61}$$

Consequently:

$$\dot{\eta} = -\frac{1}{2}(\varepsilon_1 p + \varepsilon_2 q + \varepsilon_3 r) \tag{2.62}$$

$$\dot{\varepsilon}_1 = \frac{1}{2}(\eta p - \varepsilon_3 q + \varepsilon_2 r) \tag{2.63}$$

$$\dot{\varepsilon}_2 = \frac{1}{2}(\varepsilon_3 p + \eta q - \varepsilon_1 r) \tag{2.64}$$

$$\dot{\varepsilon}_3 = \frac{1}{2}(-\varepsilon_2 p + \varepsilon_1 q + \eta r) \tag{2.65}$$

An alternative formulation is the vector representation (Kane *et al.* 1983):

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\eta} \\ \dot{\varepsilon} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\varepsilon^\top \\ \eta\mathbf{I}_{3\times 3} + \mathbf{S}(\varepsilon) \end{bmatrix}\omega_{nb}^b \tag{2.66}$$

6 DOF Kinematic Equations

Consequently, the 6 DOF kinematic equations of motion can be expressed by 7 differential equations (recall that only 6 differential are needed when using the Euler angle representation):

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}$$
$$\Updownarrow$$
$$\begin{bmatrix} \dot{\mathbf{p}}^n \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b^n(\mathbf{q}) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{4\times 3} & \mathbf{T}_q(\mathbf{q}) \end{bmatrix} \begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix} \tag{2.67}$$

where $\boldsymbol{\eta} \in \mathbb{R}^7$ and $\boldsymbol{\nu} \in \mathbb{R}^6$, and $\mathbf{J}(\boldsymbol{\eta}) \in \mathbb{R}^{7\times 6}$ is a non-quadratic transformation matrix. Equation (2.67) in component form is given by (2.57)–(2.59) and (2.62)–(2.65).

Matlab:
The transformation matrix $\mathbf{J}$ and its elements $\mathbf{J}_1 = \mathbf{R}_b^n(\mathbf{q})$ and $\mathbf{J}_2 = \mathbf{T}_q(\mathbf{q})$ can be computed directly in the GNC toolbox by using the following commands:

```
q = [eta,eps1,eps2,eps3]'
[J,J1,J2] = quatern(q)
```

The corresponding differential equations are:

```
p_dot = J1*v_o
q_dot = J2*w_nb
```

Implementation Considerations: Unit Quaternion Normalization

When integrating (2.60), a normalization procedure is necessary to ensure that the constraint:

$$\mathbf{q}^\top \mathbf{q} = \varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 + \eta^2 = 1 \tag{2.68}$$

is satisfied in the presence of measurement noise and numerical round-off errors. For this purpose, the following simple discrete-time algorithm can be applied:

Algorithm 2.1 (Discrete-Time Normalization of the Unit Quaternions)

1. $k = 0$. *Compute initial values of* $\mathbf{q}(k = 0)$.
2. *For simplicity, Euler Integration implies that (see Appendix B.2):*

$$\mathbf{q}(k+1) = \mathbf{q}(k) + h\mathbf{T}_q(\mathbf{q}(k))\boldsymbol{\omega}_{nb}^b(k) \tag{2.69}$$

   where h is the sampling time.

3. *Normalization:*

$$\mathbf{q}(k+1) = \frac{\mathbf{q}(k+1)}{\|\mathbf{q}(k+1)\|} = \frac{\mathbf{q}(k+1)}{\sqrt{\mathbf{q}^\top(k+1)\mathbf{q}(k+1)}}$$

4. *Let* $k = k + 1$ *and return to Step 2.*

A continuous time algorithm for unit quaternion normalization can be implemented by noting that:

$$\frac{d}{dt}\left(\mathbf{q}^\top \mathbf{q}\right) = 2\mathbf{q}^\top \mathbf{T}_q(\mathbf{q})\boldsymbol{\omega}_{nb}^b = 0 \tag{2.70}$$

This shows that if $\mathbf{q}$ is initialized as a unit vector, then it will remain a unit vector. Since integration of the quaternion vector $\mathbf{q}$ from the differential equation (2.60) will introduce numerical errors that will cause the length of $\mathbf{q}$ to deviate from unity, a nonlinear feedback normalization term is suggested. In Simulink™ this is done by replacing the kinematic differential equation (2.60) with:

$$\dot{\mathbf{q}} = \mathbf{T}_q(\mathbf{q})\boldsymbol{\omega}_{nb}^b + \frac{\gamma}{2}(1 - \mathbf{q}^\top \mathbf{q})\mathbf{q} \tag{2.71}$$

where $\gamma \geq 0$ (typically 100) is a design parameter reflecting the convergence rate of the normalization. This results in:

$$\frac{d}{dt}\left(\mathbf{q}^\top \mathbf{q}\right) = \underbrace{2\mathbf{q}^\top \mathbf{T}_q(\mathbf{q})\boldsymbol{\omega}_{nb}^b}_{0 \text{ since } \mathbf{q}(0) \text{ is a unit vector}} + \gamma(1 - \mathbf{q}^\top \mathbf{q})\mathbf{q}^\top \mathbf{q} = \gamma(1 - \mathbf{q}^\top \mathbf{q})\mathbf{q}^\top \mathbf{q} \tag{2.72}$$

Observe that $\mathbf{q}^\top \mathbf{q}$ will decrease if $\mathbf{q}^\top \mathbf{q} > 1$ while it increases for $\mathbf{q}^\top \mathbf{q} < 1$. When $\mathbf{q}^\top \mathbf{q} = 1$ the usual kinematic differential equations are recovered. A change of coordinates $x = 1 - \mathbf{q}^\top \mathbf{q}$, $\dot{x} = -\frac{d}{dt}\left(\mathbf{q}^\top \mathbf{q}\right)$, yields:

$$\dot{x} = -\gamma x(1 - x) \tag{2.73}$$

Linearization about $x = 0$ gives $\dot{x} = -\gamma x$. Consequently, the normalization algorithm converges with a time constant $T = \gamma^{-1}$.

2.2.3 Quaternions from Euler Angles

If the Euler angles $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$ are known and therefore the expression for the rotation matrix $\mathbf{R}_b^n(\boldsymbol{\Theta}) = \{R_{ij}\}$, a singularity free extraction procedure can be used to compute the corresponding unit quaternions (Shepperd 1978).

Algorithm 2.2 (Quaternions From Euler Angles)

1. *Given the Euler angles* ϕ, θ, *and* ψ. *Let the transformation matrix* $\mathbf{R}_b^n$ *according to (2.16) be written:*

$$\mathbf{R}_b^n(\boldsymbol{\Theta}) := \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

2. *The trace of* $\mathbf{R}_b^e(\boldsymbol{\Theta})$ *is computed as:*

$$R_{44} = \text{tr}(\mathbf{R}_b^n) = R_{11} + R_{22} + R_{33}$$

3. *Let* $1 \leq i \leq 4$ *be the index corresponding to:*

$$R_{ii} = \max(R_{11}, R_{22}, R_{33}, R_{44})$$

4. *Compute* p_i *corresponding to Index* i *of Step 3 according to:*

$$p_i = \left|\sqrt{1 + 2R_{ii} - R_{44}}\right|$$

where the sign ascribed to p_i *can be chosen to be either positive or negative.*

5. *Compute the other three* p_i*-values from:*

$$\begin{aligned} p_4p_1 &= R_{32} - R_{23} & \qquad p_2p_3 &= R_{32} + R_{23} \\ p_4p_2 &= R_{13} - R_{31} & \qquad p_3p_1 &= R_{13} + R_{31} \\ p_4p_3 &= R_{21} - R_{12} & \qquad p_1p_2 &= R_{21} + R_{12} \end{aligned}$$

by simply dividing the three equations containing the component p_i *with the known value of* p_i *(from Step 4) on both sides.*

6. *Compute the Euler parameters* $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\mathsf{T}$ *according to:*

$$\begin{aligned} \varepsilon_j &= p_j/2 \qquad (j = 1, 2, 3) \\ \eta &= p_4/2 \end{aligned}$$

Matlab:
Algorithm 2.2 is implemented in the GNC toolbox as `euler2q.m`. This algorithm can also be used to compute the initial values of the Euler parameters corresponding to Step 1 of Algorithm 2.1.

Example 2.2 (Euler Angles to Unit Quaternions)
Consider a marine vessel with orientation $\phi = 10.0^o, \theta = -20.0^o$, *and* $\psi = 30.0^o$. *The unit quaternions are computed in Matlab*™ *by using the commands:*

```
phi=10*(pi/180), theta=-20*(pi/180), psi=30*(pi/180)
q = euler2q(phi,theta,psi)
q =
     0.9437
     0.1277
    -0.1449
     0.2685
% normalization test
norm(q) =
    1.0000
```

2.2.4 Euler Angles from Quaternions

The relationship between the Euler angles ϕ, θ, and ψ (zyx-convention), and the unit quaternions q_i ($i = 1, \ldots, 4$) can be established by requiring that the rotation matrices of the two kinematic representations are equal:

$$\mathbf{R}_b^n(\Theta) := \mathbf{R}_b^n(\mathbf{q}) \tag{2.74}$$

Let the elements of $\mathbf{R}_b^n(\mathbf{q})$ be denoted by R_{ij} where the superscripts i and j denote the i-th row and j-th column. Writing expression (2.74) in component form yields a system of 9 equations with 3 unknowns (ϕ, θ, and ψ), that is:

$$\begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & -\mathrm{s}\psi\mathrm{c}\phi + \mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\phi & \mathrm{s}\psi\mathrm{s}\phi + \mathrm{c}\psi\mathrm{c}\phi\mathrm{s}\theta \\ \mathrm{s}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{c}\phi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi & -\mathrm{c}\psi\mathrm{s}\phi + \mathrm{s}\theta\mathrm{s}\psi\mathrm{c}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \tag{2.75}$$

Algorithm 2.3 (Euler Angles from Quaternions)
One solution to (2.75) is:

$$\phi = \mathrm{atan2}(R_{32}, R_{33}) \tag{2.76}$$

$$\theta = -\sin^{-1}(R_{31}) = -\tan^{-1}\left(\frac{R_{31}}{\sqrt{1 - R_{31}^2}}\right); \qquad \theta \neq \pm 90^o \tag{2.77}$$

$$\psi = \mathrm{atan2}(R_{21}, R_{11}) \tag{2.78}$$

In Algorithm 2.3 $\mathrm{atan2}(y, x)$ is the four quadrant *arctangent* of the real parts of the elements of x and y, satisfying:

$$-\frac{\pi}{2} \leq \mathrm{atan2}(y, x) \leq \frac{\pi}{2} \tag{2.79}$$

Precautions must be taken against computational errors in the vicinity of $\theta = \pm 90^o$.

Matlab:
The GNC toolbox script:

```
[phi,theta,psi] = q2euler(q)
```

is based on Algorithm 2.3. A singularity test is included in order to avoid that $\theta = \pm 90^o$.

Example 2.3 (Unit Quaternions to Euler Angles)
Consider the marine vessel in Example 2.2 where the Euler angles where converted into unit quaternions. The inverse transformation `q2euler.m` *results in:*

```
q =[0.9437,0.1277,-0.1449,0.2685]'
[phi,theta,psi] = q2euler(q/norm(q))
phi   = 0.1746
theta =-0.3491
psi   = 0.5235
```

corresponding to $\phi = 10.0^o, \theta = -20.0^o$, *and* $\psi = 30.0^o$.

2.2.5 QUEST Algorithm for Position and Attitude Determination

The position and three-axis attitude of a vessel can be determined from two or more vector observations by using the *TRIAD* or *QUEST* algorithms (Schuster and Oh 1981). The *TRIAD* algorithm, provides a deterministic nonoptimal solution for the attitude based on two vector observations. The *QUEST* algorithm on the other hand is an optimal algorithm which determines the attitude that achieves the best weighted overlap of an arbitrary number reference and observation vectors. In Schuster and Oh (1981), analytical expressions are given for the covariance matrices for the two algorithms using a model for the measurement errors, and the mathematical relationship of the two algorithms and their relative merits are discussed.

Matlab:
The QUEST algorithm is included in the GNC Toolbox as:

```
[R,q] = quest(W,V)
```

where `R` is the quaternion rotation matrix between the vectors `W` and `V`, and `q` is the unit quaternion vector. A particular useful application of this algorithm is determination of the position and the attitude of a model ship using a camera. For this purpose the Matlab™ function:

```
[eta,q,R] = quest6DOF(y,mb,rcamera)
```

where `y` is a vector $\mathbf{y}^n = [(\mathbf{y}_1^n)^\top, (\mathbf{y}_2^n)^\top, (\mathbf{y}_3^n)^\top]$ of real-time NED camera measurements $\mathbf{y}_i^n = [n_i, e_i, d_i]^\top$ $(i = 1, 2, 3)$ for the three markers onboard the vessel, `mb` is a vector of marker positions $\mathbf{m}_i^b = [x_{mi}, y_{mi}, z_{mi}]^\top$ $(i = 1, 2, 3)$ in body-fixed coordinates relative to the CG of the ship, and `rcamera` is a vector $\mathbf{r}_{\text{camera}}^n = [x_c, y_c, z_c]^\top$ denoting the location of the camera in NED coordinates.

6 DOF Position and Attitude form Camera Measurements

Application of the QUEST algorithm will now be demonstrated for CyberShip I (model ship scale 1:70 of a supply vessel) in the GNC Laboratory at the Norwegian University of Science and Technology, see http://www.itk.ntnu.no/ansatte/Fossen_Thor/GNC.

The GNC Laboratory uses a real-time camera image processing system developed by the Computer Vision and Robotics Group at the University of Girona, Spain (Batlle *et al.* 2000). The camera system has the feature that several markers located onboard a model ship can be distinguished by their color. The marker positions $\mathbf{y}_i^n$ $(i = 1, 2, 3)$ are measured using one wall mounted camera with fixed coordinates $\mathbf{r}_{\text{camera}}^n$ as shown in Figure 2.4.

Consider the problem of determining the position vector $\mathbf{r}_{\text{cg}}^n = [x, y, z]^\top$–i.e., the NED position of the ship's CG, and the attitude parameterized by unit quaternions $\mathbf{q}$ when the ship is moving. The ship marker positions $\mathbf{m}_i^b$ $(i = 1, 2, 3)$ are in body-fixed coordinates relative to the CG. This gives us three measurement equations:

$$\mathbf{r}_{\text{cg}}^n = \mathbf{r}_{\text{camera}}^n + \mathbf{y}_i^n - \mathbf{R}_b^n(\mathbf{q})\mathbf{m}_i^b, \quad (i = 1, 2, 3) \tag{2.80}$$

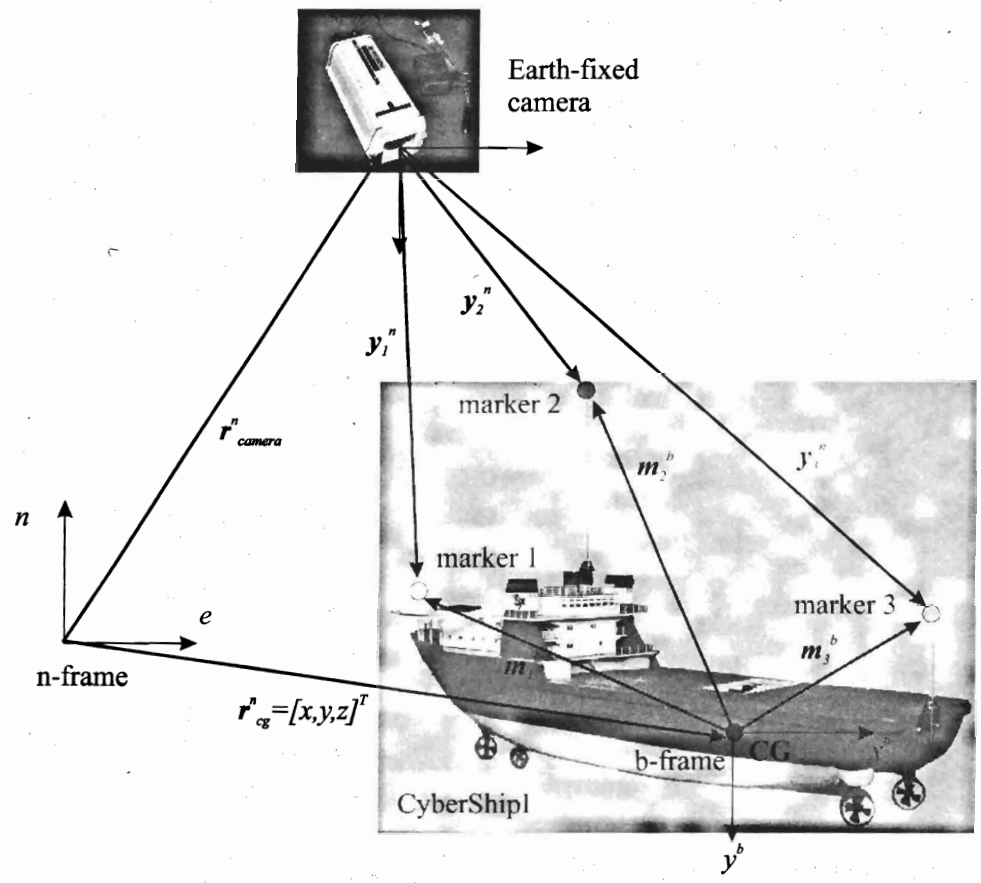


Figure 2.4: GNC-Laboratory camera measurement system for 6 DOF position and attitude determination.

This is the position vector that must be computed from the camera measurements. Notice that one camera produces 9 measurement equations or three $\mathbf{y}_i^n$-vectors–i.e., one for each marker, in which there are 7 unknowns. These are the position vector $\mathbf{r}_{cg}^n = [x, y, z]^\top$ and the attitude vector $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top$. The measurement equations are, however, nonlinear in the unknowns $\mathbf{r}_{cg}^n$ and $\mathbf{q}$, so we cannot solve for these quantities directly. Therefore, the QUEST algorithm is used to solve this problem.

The existence of a solution depends on the configuration of the markers located onboard the ship. These must be located such that the measurement equations are linearly independent. For instance, locating all markers close to each other will not work. It is advantageous from a numerical point of view to maximize the distance between the markers and also to use different heights. However, this is not a hard requirement, since we have 9 equations and only 7 unknowns.

The marker configuration shown in Figure 2.4, were all markers are on the centerline but at different heights (z-positions) along the x-axis of the ship, gave a unique solution. The

numerical values for the CyberShip I markers were:

$$\begin{aligned} \mathbf{m}_1^b &= [0.69, 0, -0.15]^\top \\ \mathbf{m}_2^b &= [0.42, 0, -0.51]^\top \\ \mathbf{m}_3^b &= [-0.38, 0, -0.18]^\top \end{aligned}$$

while the camera was located on the wall at:

$$\mathbf{r}^n_{\text{camera}} = [2.11, -0.48, 1.89]^\top$$

Assume that the following marker positions were measured with the camera:

$$\begin{aligned} \mathbf{y}_1^n &= [-0.09,\ 0.947, -1.997]^\top \\ \mathbf{y}_2^n &= [0.066, 3.688, -2.327]^\top \\ \mathbf{y}_3^n &= [0.641, 3.175, -1.934]^\top \end{aligned}$$

The position and attitude vector $\boldsymbol{\eta}$, unit quaternion vector $\mathbf{q}$, and rotation matrix $\mathbf{R}_b^n(\mathbf{q})$ corresponding to these measurements were computed to:

$$\begin{aligned} \boldsymbol{\eta} &= [2.5, 3.0, 0.1, 10^o, 5^o, 134^o]^\top \\ \mathbf{q} &= [0.3924, -0.0060, 0.0971, 0.9146]^\top \\ \mathbf{R}_b^n(\mathbf{q}) &= \begin{bmatrix} -0.6920 & -0.7189 & 0.0653 \\ 0.7166 & -0.6732 & 0.1824 \\ -0.0872 & 0.1730 & 0.9811 \end{bmatrix} \end{aligned}$$

Matlab:
The numerical results were generated in Matlab using `quest6DOF.m`, see the example script `ExQuest.m` for details:

```
y = [-0.097,3.947,-1.997,0.066,3.688,-2.327,...
     0.641,3.175,-1.934]'
mb = [0.69,0,-0.15,0.42,0,-0.51,-0.38,0,-0.18]'
rcamera = [2.11,-0.48,1.89]'
[eta,q,R] = quest6DOF(y,mb,rcamera)
```

2.3 Transformation between ECEF and NED

Wide area or terrestrial guidance and navigation implies that the position should be related to the Earth center instead of local frame on the Earth surface. This can be done by using the results from the previous sections.

2.3.1 Longitude and Latitude Transformations

The transformation between the ECEF and NED velocity vectors is:

$$\dot{\mathbf{p}}^e = \mathbf{R}_n^e(\boldsymbol{\Psi})\dot{\mathbf{p}}^n = \mathbf{R}_n^e(\boldsymbol{\Psi})\mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{v}_o^b \tag{2.81}$$

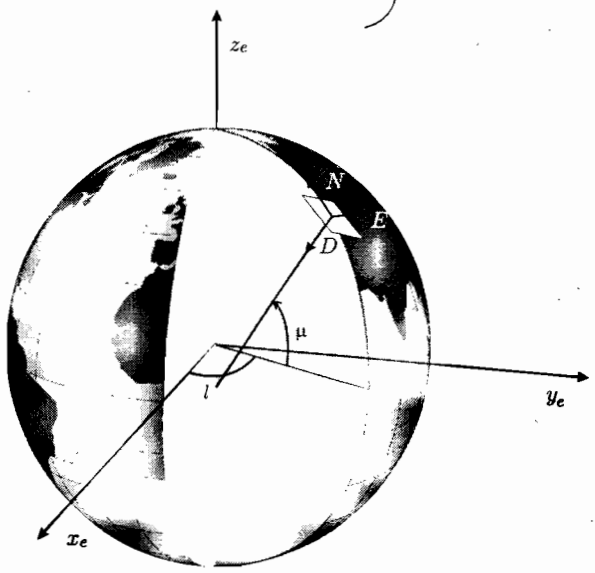


Figure 2.5: Definitions of longitude l and lattitude μ and the NED reference frame on the surface of the Earth. The D-axis points in the normal direction to the Earth's surface.

where $\mathbf{\Psi} = [l, \mu]^\top \in \mathcal{S}^2$ is a vector formed by longitude l and latitude μ, and $\mathbf{R}_n^e(\mathbf{\Psi})$: $\mathcal{S}^2 \rightarrow SO(3)$ is a rotation matrix between ECEF and NED. This is found by performing two principal rotations: (1) a rotation l about the z-axis, and (2) a rotation $(-\mu - \pi/2)$ about the y-axis. This gives:

$$\begin{aligned} \mathbf{R}_n^e(\mathbf{\Psi}) &= \mathbf{R}_{z,l}\mathbf{R}_{y,-\mu-\frac{\pi}{2}} \\ &= \begin{bmatrix} \cos l & -\sin l & 0 \\ \sin l & \cos l & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(-\mu-\frac{\pi}{2}) & 0 & \sin(-\mu-\frac{\pi}{2}) \\ 0 & 1 & 0 \\ -\sin(-\mu-\frac{\pi}{2}) & 0 & \cos(-\mu-\frac{\pi}{2}) \end{bmatrix} \end{aligned} \tag{2.82}$$

Using the trigonometric formulas: $\cos(-\mu - \frac{\pi}{2}) = -\sin\mu$, and $\sin(-\mu - \frac{\pi}{2}) = -\cos\mu$, yields:

$$\mathbf{R}_n^e(\mathbf{\Psi}) = \begin{bmatrix} -\cos l \sin\mu & -\sin l & -\cos l\cos\mu \\ -\sin l \sin\mu & \cos l & -\sin l\cos\mu \\ \cos\mu & 0 & -\sin\mu \end{bmatrix} \tag{2.83}$$

Hence, the ECEF positions $\mathbf{p}^e = [x, y, z]^\top$ can be found by integration of (2.81). This equation can also be used when designing a global way-point tracking control system for ships.

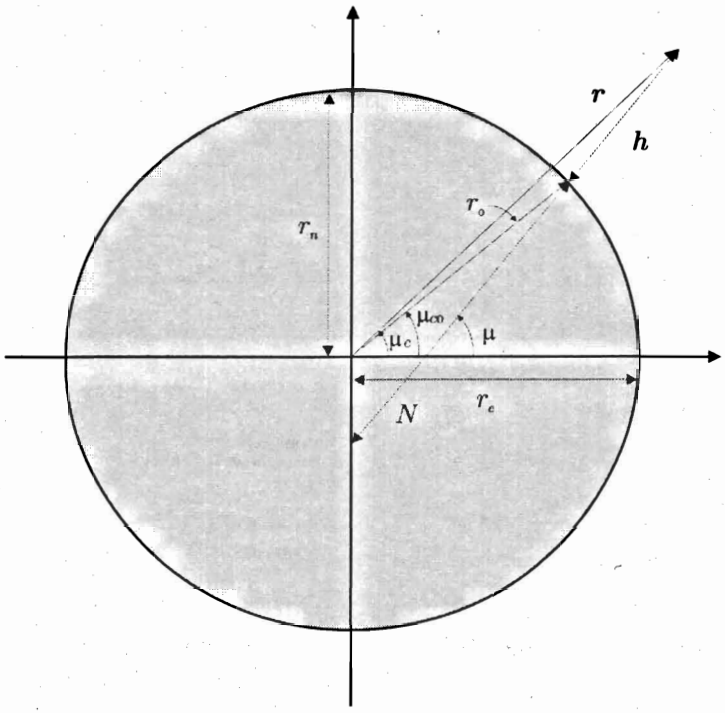


Figure 2.6: Definitions of the ellipsoidal parameters.

Matlab:
The rotation matrix $\mathbf{R}_n^e$ is computed using the GNC toolbox command:

```
R = Rll(l,mu)
```

Flat Earth Navigation

For flat Earth navigation it can be assumed that the NED tangent plane is fixed on the surface of the Earth–i.e., l and and μ are constants, by assuming that the operating radius of the vessel is limited. This suggests that the NED position vector:

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{v}_o^b \tag{2.84}$$

is used for control design. When designing dynamic positioning (DP) systems for offshore vessels this is particular useful. The ECEF coordinates for flat Earth navigation is found by requiring that $\boldsymbol{\Psi} = \boldsymbol{\Psi}_0 = \text{constant}$, such that:

$$\mathbf{R}_n^e(\boldsymbol{\Psi}_0) = \mathbf{R}_o = \text{constant} \tag{2.85}$$

$$\dot{\mathbf{p}}^e = \mathbf{R}_o\mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{v}_o^b \tag{2.86}$$

When designing global way-point tracking control systems for ships, "flat Earth" is not a good approximation since (l, μ) will vary largely for ships in transit between the different continents. Hence, the more general expression (2.81) should be used for global navigation.

2.3.2 Longitude and Latitude from ECEF Coordinates

The measurements of GPS are given in the Cartesian ECEF frame, but this is a measurement that does not make a lot of sense to the user. Presentation of terrestrial position data $\mathbf{p}^e = [x, y, z]^\top$ is therefore made in terms of the ellipsoidal parameters longitude l, latitude μ, and height h.

Figure 2.6 shows the definitions of parameters needed for the transformations. The reference ellipsoid used for GPS, WGS-84, is found by rotating an ellipse around the polar axis. Because of symmetry about the polar axis, it is only necessary to look at the meridian plane (latitude) equations. The ellipsoid's origin coincides with the mass center of the Earth. The most important parameters of the WGS-84 ellipsoid are listed in Table 2.2.

In Figure 2.6, μ_c and μ are the *geocentric* and *geodetic* latitudes, respectively. Furthermore, r is the geocentric radius, r_0 is the geocentric radius of the user position projected onto the surface of the Earth, h is the ellipsoidal height, and N is the radius of curvature in the prime vertical. N is calculated by:

$$N = \frac{r_e^2}{\sqrt{r_e^2 \cos^2 \mu + r_p^2 \sin^2 \mu}} \tag{2.87}$$

where the equatorial and polar earth radii, r_e and r_p, are the semiaxes of the ellipsoid.

Longitude l is easily computed as:

$$l = \operatorname{atan}\left(\frac{y_e}{x_e}\right) \tag{2.88}$$

while latitude μ and height h are implicitly computed by:

$$\tan \mu = \frac{z_e}{p}\left(1 - e^2 \frac{N}{N+h}\right)^{-1} \tag{2.89}$$

$$h = \frac{p}{\cos \mu} - N \tag{2.90}$$

where e, the Earth's eccentricity is:

$$e = \sqrt{1 - \left(\frac{r_p}{r_e}\right)^2} \tag{2.91}$$

Since these two equations are implicit, they can be solved iteratively by using Algorithm 2.4 (Hofmann-Wellenhof *et al.* 1994):

Table 2.2: WGS-84 parameters.

Parameters		Comments
r_e	$= 6378137$ m	Equatorial radius of ellipsoid (semimajor axis)
r_p	$= 6356752$ m	Polar axis radius of ellipsoid (semiminor axis)
ω_e	$= 7.292115 \cdot 10^{-5}$ rad/s	Angular velocity of the Earth.
μ_g	$= 3986005 \cdot 10^8$ m^3/s^2	Gravitational constant of Earth.
e	$= 0.0818$	Eccentricity of ellipsoid.

Algorithm 2.4 (Transformation of (x_e, y_e, z_e) to (l, μ, h))

1. *Compute* $p = \sqrt{x_e^2 + y_e^2}$.
2. *Compute the approximate value* $\mu_{(0)}$ *from:*

$$\tan \mu_{(0)} = \frac{z_e}{p}(1 - e^2)^{-1}$$

3. *Compute an approximate value N from:*

$$N = \frac{r_e^2}{\sqrt{r_e^2 \cos^2 \mu_{(0)} + r_p \sin^2 \mu_{(0)}}}$$

4. *Compute the ellipsoidal height by:*

$$h = \frac{p}{\cos \mu_{(0)}} - N_{(0)}.$$

5. *Compute an improved value for the latitude by:*

$$\tan \mu = \frac{z_e}{p}\left(1 - e^2 \frac{N_{(0)}}{N_{(0)} + h}\right)^{-1}$$

6. *Check for another iteration step: if* $|\mu - \mu_{(0)}| < tol$ *where tol is a small number, then the iteration is complete. Otherwise set* $\mu_{(0)} = \mu$ *and continue with Step 3.*

Matlab:
Algorithm 2.4 is programmed in the GNC toolbox script:

```
[l,mu,h] = ecef2llh(x,y,z)
```

Several other algorithms can be used for this purpose; see Farrell and Barth (1998) and references therein. An approximate solution can also be found in Hofmann-Wellenhof *et al.* (1994), and an exact explicit solution is given by Zhu (1993).

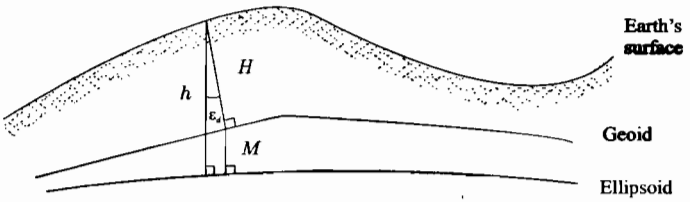


Figure 2.7: Illustration of ellipsoidal and orthonometric heights h and H where ε_d is the deflection of gravity and M is the geoidal height (undulation).

Height transformation

The WGS-84 ellipsoid is a global ellipsoid, which is only an approximation of the mean sea level of the Earth. It can deviate from the real mean sea level by as much as 100 meters at certain locations. The Earth's geoid, on the other hand, is defined physically and its center is coincident with the center of the Earth. It is an equipotential surface so that it has the same gravitational magnitude all over the surface, and the gravity vector is always perpendicular to the geoid.

The geoid is the surface chosen as a zero level reference. The ellipsoidal height h in Figure 2.7 must therefore be transformed to *orthometric* height H in Figure 2.7 through the relation:

$$h \approx H + M$$

where M is called the *geoidal* height. The angle ε_d is small enough for the above approximation to be sufficiently accurate for all practical purposes. The angle ε_d is known as the deflection of the vertical, and does not exceed 30 arcseconds in most of the world. In fact the largest deflection encountered over the entire earth is in the order of 1 arcminutes (Britting 1971). The geoidal height M is found through a *datum* transformation (Hofmann-Wellenhof *et al.* 1994).

2.3.3 ECEF Coordinates from Longitude and Latitude

The transformation from $\Psi = [l, \mu]^\top$ for given heights h to $\mathbf{p}^e = [x, y, z]^\top$ is given by e.g. Heiskanen and Moritz (1967):

$$\begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} (N+h)\cos\mu\cos l \\ (N+h)\cos\mu\sin l \\ \left(\frac{r_p^2}{r_e^2}N + h\right)\sin\mu \end{bmatrix} \tag{2.92}$$

For a ship h is the vertical distance from the sea level to the b-frame coordinate origin. The b-frame is usually chosen to coincide with the center of gravity.

Matlab:
The transformation from $\Psi = [l, \mu]^\top$ to $\mathbf{p}^e = [x, y, z]^\top$, Equation (2.92), is programmed in the GNC toolbox script:

```
[x,y,z]=llh2ecef(l,mu,h)
```

Example 2.4 (ECEF Coordinates from l and μ)
Assume that $l = 10.3^o$, $\mu = 63.0^o$, and $h = 0$ (m). Hence, the ECEF coordinates are computed to be:

$$\begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} 2856552\ (m) \\ 519123\ (m) \\ 5659978\ (m) \end{bmatrix}$$

using the Matlab command:

```
[x,y,z]=llh2ecef(10.3*(pi/180),63.0*(pi/180),0)
```

2.4 Transformations for Stability and Current Axes

When deriving the hydrodynamic derivatives for a marine vessel it is common to rotate the body fixed axes (b-frame) such that the direction of the speed:

$$U = \sqrt{u^2 + v^2 + w^2} \tag{2.93}$$

points in the opposite direction of the new x-axis. Hence, the drag force will be along the body-fixed x-axis, while the lift force will be along the z-axis. Lift and drag forces can then be computed as a function of forward speed and transformed back to the b-frame coordinates.

The transformation of the vessel's b-frame to this system is defined by two principal rotations. First the b-frame coordinate system is rotated a *positive* angle α about the y-axis. This angle is referred to as *angle of attack* and the new coordinate system is called the *stability axes*. The stability axes are then rotated a *negative* sideslip angle $-\beta$ about the z-axis. The resulting coordinate system represents the *current axes*.

The names *stability* and *wind axes* are commonly used in aerodynamics to model lift and drag forces which both are nonlinear functions of α, β and U. This convention has been adopted by the marine community and SNAME to describe lift and drag forces on submerged vessels (SNAME 1950). Hence, *wind axes* are modified to *current axes*.

The transformation between the b-frame and the stability/current axes can be mathematically expressed as:

$$\mathbf{v}^{\text{stab}} = \mathbf{R}_{y,\alpha}\mathbf{v}^b \tag{2.94}$$

$$\mathbf{v}^{\text{current}} = \mathbf{R}_{z,-\beta}\mathbf{v}^{\text{stab}} \tag{2.95}$$

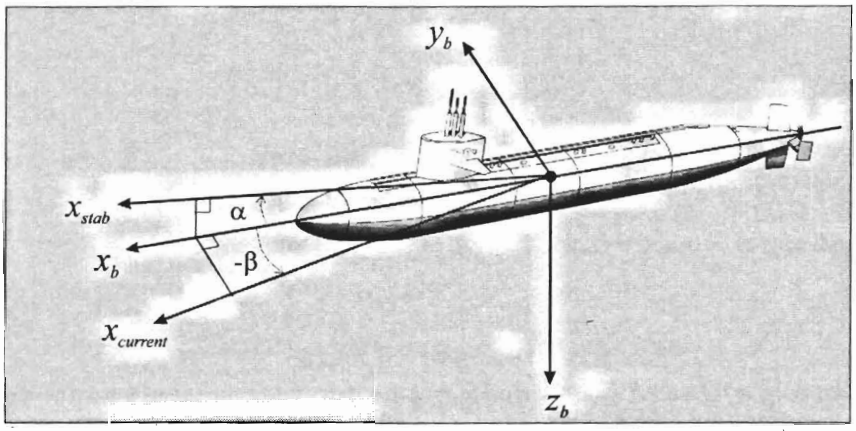


Figure 2.8: Definiton of stability and current axes in terms of the angle of attack α and the sideslip angle β.

where

$$\mathbf{R}_{y,\alpha} = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix}, \quad \mathbf{R}_{z,-\beta} = \mathbf{R}_{z,\beta}^{\top} = \begin{bmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.96}$$

The transformation matrix from the *b*-frame to *current axes* becomes:

$$\begin{aligned} \mathbf{R}_b^{\text{current}} &= \mathbf{R}_{z,-\beta}\mathbf{R}_{y,\alpha} \\ &= \begin{bmatrix} \cos\beta\cos\alpha & \sin\beta & \cos\beta\sin\alpha \\ -\sin\beta\cos\alpha & \cos\beta & -\sin\beta\sin\alpha \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \end{aligned} \tag{2.97}$$

The velocity transformation:

$$\mathbf{v}^{\text{current}} = \mathbf{R}_b^{\text{current}}\mathbf{v}^b \tag{2.98}$$

can now be rewritten as:

$$\mathbf{v}^b = (\mathbf{R}_b^{\text{current}})^{\top}\mathbf{v}^{\text{current}} \tag{2.99}$$

$$\Updownarrow$$

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathbf{R}_{y,\alpha}^{\top}\mathbf{R}_{z,-\beta}^{\top}\begin{bmatrix} U \\ 0 \\ 0 \end{bmatrix} \tag{2.100}$$

Writing this expression in component form, yields:

$$u = U\cos\alpha\cos\beta \tag{2.101}$$
$$v = U\sin\beta \tag{2.102}$$
$$w = U\sin\alpha\cos\beta \tag{2.103}$$

For small angles of α and β (linear theory), this reduces to:

$$u = U, \quad v = \beta U, \quad w = \alpha U \tag{2.104}$$

such that $\alpha = w/U$ and $\beta = v/U$. Notice that the state-space model of a marine vessel can be represented by using one of the following vectors:

$$\boldsymbol{\nu} = [u, v, w, p, q, r]^\top \tag{2.105}$$
$$\boldsymbol{\nu}^{\text{current}} = [U, \beta, \alpha, p, q, r]^\top \tag{2.106}$$

The latter representation is often more intuitive to use from a hydrodynamic point of view, while control engineers prefer the former. Both representations are, however, equivalent since there exists a nonlinear transformation between (u, v, w) and (U, β, α), given by (2.101)–(2.103). For small angles the approximation:

$$\boldsymbol{\nu}^{\text{current}} = \mathbf{V}(U)\boldsymbol{\nu} \tag{2.107}$$

with:

$$\mathbf{V}(U) = \text{diag}\{1, 1/U, 1/U, 1, 1, 1\} \tag{2.108}$$

can be used. Notice that $\dot{\mathbf{V}}(U) \neq \mathbf{0}$ since the forward speed U is time-varying.

2.5 Exercises

Exercise 2.1 *Show that* $\dot{\mathbf{R}}_b^n(\Theta) = \mathbf{R}_n^b(\Theta)\mathbf{S}(\boldsymbol{\omega}_{nb}^b)$ *is equivalent with differentiating each component in the rotation matrix* $\frac{d}{dt}\{\mathbf{R}_n^b(\Theta)\}_{ij}$ $(i, j = 1, 2, 3)$. *It is sufficient to show this for the two elements* R_{11} *and* R_{12}.

Exercise 2.2 *What are the ECEF positions corresponding to* $(l, \mu, h) = (0^o, 0^o, 0)$ *and* $(l, \mu, h) = (180^o, 0^o, 0)$? *Compute the distance (Earth diameter) between the two points. Repeat the computations for* $(l, \mu, h) = (0^o, 90^o, 0)$ *and* $(l, \mu, h) = (0^o, -90^o, 0)$. *Are the two diameters equal?*

Exercise 2.3 *Given the ECEF positions* $(x_e, y_e, z_e) = 10^6 \cdot (2.7688, 1.5986, 5.5005)$. *Compute* l *and* μ *using Algorithm 2.4 and find the city corresponding to these coordinates.*

Exercise 2.4 *Consider a marine craft moving at forward velocity* $u = 2$ *m/s and transverse velocity* $v = 0$ *m/s. The yawing rate during autopilot control is* $r = 1$ *deg/s.*

a) Write down the differential equations for the North and East positions, and the yaw (heading) angle. Assume that the heave, roll *and* pitch *modes can be neglected.*

b) Plot the positions and heading as a function of time. Present the results in two plots: one with North-East axes and one where the yaw angle is plotted against time.

Exercise 2.5 *Consider an underwater vehicle moving at velocity* $\mathbf{v}^b = [u, v, w]^\top = [4, 1, 2]^\top$ *in the b-frame. The Euler angles are* $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top = [10^o, 20^o, -50^o]^\top$. *Solve the following problems by using the Matlab*$^{\text{TM}}$ *GNC toolbox:*

a) Compute the NED (n-frame) velocity vector $\mathbf{v}^n$.

b) Compute the unit quaternion vector $\mathbf{q}$ *corresponding to the Euler angles* $\boldsymbol{\Theta}$.

c) The longitude and latitude of the underwater vehicle is $l = 70^o$ *and* $\mu = 10^o$. *Compute the ECEF (e-frame) velocity vector* $\mathbf{v}^e$.

Exercise 2.6 *Consider the kinematic equations:*

$$\dot{\mathbf{p}}^n = \mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{v}^b$$

$$\dot{\boldsymbol{\Theta}} = \mathbf{T}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\omega}_{nb}^b$$

Show that the NED (n-frame) accelerations can be written:

$$\begin{bmatrix} \ddot{\mathbf{p}}^n \\ \ddot{\boldsymbol{\Theta}} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b^n(\boldsymbol{\Theta}) & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}^b \\ \dot{\boldsymbol{\omega}}_{nb}^b \end{bmatrix} + \begin{bmatrix} \mathbf{0} & -\mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{S}(\mathbf{v}^b) \\ \mathbf{0} & \dot{\mathbf{T}}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}) \end{bmatrix} \begin{bmatrix} \mathbf{v}^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix}$$

Hint: $\mathbf{S}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a} = -\mathbf{S}(\mathbf{b})\mathbf{a}$.

Exercise 2.7 *Show that the* quaternion rotation matrix *can be written:*

$$\mathbf{R}_b^n(\mathbf{q}) = \mathbf{I}_{3\times 3} + 2\eta\mathbf{S}(\boldsymbol{\varepsilon}) + 2\mathbf{S}^2(\boldsymbol{\varepsilon})$$

by substituting the definitions $\boldsymbol{\varepsilon} = \boldsymbol{\lambda} \sin\frac{\beta}{2}$ *and* $\eta = \cos\frac{\beta}{2}$ *into the rotation matrix:*

$$\mathbf{R}_{\eta,\boldsymbol{\varepsilon}} = \mathbf{I}_{3\times 3} + \sin\beta\mathbf{S}(\boldsymbol{\lambda}) + (1 - \cos\beta)\mathbf{S}^2(\boldsymbol{\lambda})$$

Chapter 3

Dynamics of Marine Vessels

The marine vessel equations of motion involve the study of *statics* and *dynamics*. Statics is concerned with the equilibrium of bodies at rest or moving with constant velocity, whereas dynamics is concerned with bodies having accelerated motion. Statics is the oldest of the engineering sciences. In fact, important contributions were made over 2000 years ago by Archimedes (287–212 BC), who derived the basic law of hydrostatic buoyancy. This result is the foundation for static stability analysis of marine vessels.

The study of dynamics started much later, since accurate measurements of time are necessary to perform dynamic experiments. The scientific basis of dynamics was provided by Newton's laws, published in 1687.

In the following sections, it is shown that the 6 DOF nonlinear dynamic equations of motion can be conveniently expressed as (Fossen 1991, Fossen 1994):

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{3.1}$$

where

$\mathbf{M}$	- system inertia matrix (including added mass)
$\mathbf{C}(\boldsymbol{\nu})$	- Coriolis-centripetal matrix (including added mass)
$\mathbf{D}(\boldsymbol{\nu})$	- damping matrix
$\mathbf{g}(\boldsymbol{\eta})$	- vector of gravitational/buoyancy forces and moments
$\boldsymbol{\tau}$	- vector of control inputs
$\mathbf{g}_o$	- vector used for pretrimming (ballast control)
$\mathbf{w}$	- vector of environmental disturbances (wind, waves and currents)

The expressions for $\mathbf{M}, \mathbf{C}(\nu), \mathbf{D}(\nu), \mathbf{g}(\eta)$, and $\mathbf{g}_o$ are derived in Sections 3.1–3.2 while environmental disturbances $\mathbf{w}$ are treated separately in Chapter 4. Sections 3.3 and 3.5 discuss nonlinear model properties and system transformations applicable to 6 DOF vessel models. Matlab examples will demonstrate how the model matrices and vectors can be computed.

The nonlinear model presented in this chapter is mainly intended for control systems design in combination with system identification and parameter estimation. Hence, the extensive literature on hydrodynamics should be consulted to obtain numerical values for the hydrodynamic derivatives which are necessary for accurate prediction and computer simulations. Some standard references in hydrodynamics are Faltinsen (1990), Newman (1977), Sarpkaya (1981), and Triantafyllou and Hover (2002).

Before the 6 DOF dynamic equations of motion are derived, some principles from *Newtonian* mechanics are reviewed, while *Lagrangian* mechanics will be used to derive the expressions for hydrodynamic added mass in Section 3.2.1. Detailed discussions of Newtonian and Lagrangian mechanics is found in Goldstein (1980), Hughes (1986), Kane *et al.* (1983), Meirovitch (1990), and Egeland and Gravdahl (2002).

3.1 Rigid-Body Dynamics

In this section the Newton-Euler formulation for rigid bodies and the foundation of vectorial mechanics are presented. In this context it is convenient to define the vectors without reference to a coordinate frame (*coordinate free vector*). A vector $\vec{v}$ is defined by its magnitude and the direction. The vector $\vec{v}$ decomposed in the inertial reference frame is denoted as $\mathbf{v}^i$, which is also referred to as a *coordinate vector*.

Newton–Euler Formulation

The *Newton–Euler formulation* is based on *Newton's Second Law* which relates mass m, acceleration $\dot{\vec{v}}_c$ and force $\vec{f}_c$ according to:

$$m\dot{\vec{v}}_c = \vec{f}_c \tag{3.2}$$

where the subscript c denotes the center of gravity (CG). This law must be formulated in an *inertial frame*.

If no force is acting ($\vec{f}_c = \vec{0}$), then the rigid body is moving with constant speed ($\vec{v}_c =$ constant), or the body is at rest ($\vec{v}_c = \vec{0}$); a result known as *Newton's First Law*. Newton's laws were published in 1687 by Isaac Newton (1643–1727) in "Philosophia Naturalis Principia Mathematica."

Euler's First and Second Axioms

Leonhard Euler (1707–1783) suggested in his "Novi Commentarii Academiae Scientarium Imperialis Petropolitane" to express Newton's Second Law in terms of conservation of both linear momentum $\vec{p}_c$ and angular momentum $\vec{h}_c$. These results are known as *Euler's First and Second Axioms*, respectively.

$$\dot{\vec{p}}_c = \vec{f}_c \qquad \vec{p}_c = m\vec{v}_c \tag{3.3}$$

$$\dot{\vec{h}}_c = \vec{m}_c \qquad \vec{h}_c = I_c\vec{\omega}_{ib} \tag{3.4}$$

where $\vec{f}_c$ and $\vec{m}_c$ are the forces and moments acting on the body's center of gravity (denoted by subscript *c)*, $\vec{\omega}_{ib}$ is the angular velocity of frame *b* relative to frame *i*, and I_c is the inertia dyadic about the body's center of gravity (to be defined later). The application of these equations is often referred to as *vectorial mechanics* since both conservation laws are expressed in terms of vectors.

When deriving the equations of motion it will be assumed: (1) that the vessel is rigid, and (2) that the NED frame is inertial. The first assumption eliminates the consideration of forces acting between individual elements of mass while the second eliminates forces due to the Earth's motion relative to a star-fixed inertial reference system. For guidance and navigation applications in space it is usual to use a star-fixed reference frame or a reference frame rotating with the Earth. Marine vessels are, on the other hand, usually related to the NED reference frame. This is a good assumption since forces on marine craft due to the Earth's rotation:

$$\omega_{ie} = 7.2921 \cdot 10^{-5} \text{ (rad/s)} \tag{3.5}$$

are quite small compared to the hydrodynamic forces.

3.1.1 Translational Motion

The mass of a rigid body is defined by the volume integral:

$$m := \int_V \rho_m dV \tag{3.6}$$

where ρ_m is the density of the body. For simplicity it will be assumed constant density such that $\dot{m} = 0$. The vector from the inertial frame to the vessel's *CG* is defined as:

$$\vec{r}_c := \frac{1}{m} \int_V \vec{r}\,' \rho_m dV \tag{3.7}$$

The position of the volume element dV is (see Figure 3.1):

$$\vec{r}\,' = \vec{r}_c + \vec{r} \tag{3.8}$$

From the definition of the *CG* it is seen that:

$$\begin{aligned} \int_V \vec{r} \rho_m dV &= \int_V \vec{r}\,' \rho_m dV - \int_V \vec{r}_c \rho_m dV \\ &= m\vec{r}_c - \vec{r}_c \int_V \rho_m dV = \vec{0} \end{aligned} \tag{3.9}$$

since $\vec{r}_c$ is constant over the entire volume.

For marine vessels it is desirable to derive the equations of motion for an arbitrary origin *O* in the *b*-frame to take advantage of the vessel's geometric properties. Since the hydrodynamic and kinematic forces and moments are given in the *b*-frame, Newton's laws will be formulated in the *b*-frame as well. The *b*-frame coordinate system is rotating with respect to the *i*-frame (inertial system). This implies that the velocities of *CG* and *O* must satisfy:

$$\vec{v}_c = \vec{v}_o + \vec{\omega}_{ib} \times \vec{r}_g \tag{3.10}$$

It is common to assume that the NED frame is an approximate inertial frame by neglecting

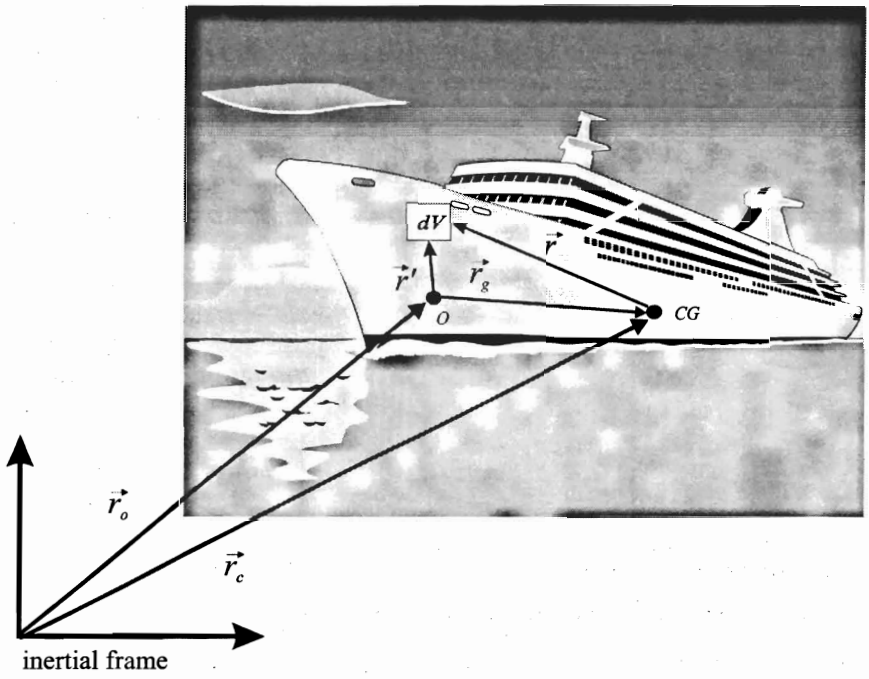


Figure 3.1: Definition of the volume element dV and the coordinate origins O and CG.

the Earth rotation $\vec{\omega}_{ie}$ and the angular velocity $\vec{\omega}_{en}$ due to slow variations in longitude and latitude (flat Earth assumption). This implies that:

$$\begin{aligned} \vec{\omega}_{ib} &= \vec{\omega}_{ie} + \vec{\omega}_{en} + \vec{\omega}_{nb} \\ &\approx \vec{\omega}_{nb} \end{aligned} \tag{3.11}$$

is a good approximation. Decomposing (3.10) into the b frame under the assumption that $\vec{\omega}_{ib} = \vec{\omega}_{nb}$, yields:

$$\mathbf{v}_c^b = \mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b \tag{3.12}$$

Hence:

$$\begin{aligned} \mathbf{v}_c^n &= \mathbf{R}_b^n \mathbf{v}_c^b \\ &= \mathbf{R}_b^n (\mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b) \end{aligned} \tag{3.13}$$

Time differentiation of this expression, yields the acceleration of the CG in NED coordinates:

$$\begin{aligned} \dot{\mathbf{v}}_c^n &= \mathbf{R}_b^n (\dot{\mathbf{v}}_o^b + \dot{\boldsymbol{\omega}}_{nb}^b \times \mathbf{r}_g^b + \boldsymbol{\omega}_{nb}^b \times \dot{\mathbf{r}}_g^b) + \dot{\mathbf{R}}_b^n (\mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b) \\ &= \mathbf{R}_b^n (\dot{\mathbf{v}}_o^b + \dot{\boldsymbol{\omega}}_{nb}^b \times \mathbf{r}_g^b) + \mathbf{R}_b^n \mathbf{S}(\boldsymbol{\omega}_{nb}^b)(\mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b) \\ &= \mathbf{R}_b^n [\dot{\mathbf{v}}_o^b + \mathbf{S}(\dot{\boldsymbol{\omega}}_{nb}^b)\mathbf{r}_g^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{v}_o^b + \mathbf{S}^2(\boldsymbol{\omega}_{nb}^b)\mathbf{r}_g^b] \end{aligned} \tag{3.14}$$

Here we have used the facts that $\dot{\mathbf{R}}_b^n = \mathbf{R}_b^n \mathbf{S}(\boldsymbol{\omega}_{nb}^b)$, see Theorem 2.2 in Section 2.2.1, and $\dot{\mathbf{r}}_g^b = \mathbf{0}$ (the distance between O and CG is constant in a rigid body).

Euler's first axiom (3.3) decomposed in the inertial frame is written:

$$m\dot{\mathbf{v}}_c^i = \mathbf{f}_c^i \tag{3.15}$$

Since NED is assumed to be an inertial frame, (3.15) can be approximated by:

$$\begin{aligned} m\dot{\mathbf{v}}_c^n &= \mathbf{f}_c^n \\ &= \mathbf{R}_b^n \mathbf{f}_c^b \end{aligned} \tag{3.16}$$

Substituting (5.57) into (3.16) yields the translational motion of the coordinate origin O in body-fixed coordinates:

$$m[\dot{\mathbf{v}}_o^b + \mathbf{S}(\dot{\boldsymbol{\omega}}_{nb}^b)\mathbf{r}_g^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{v}_o^b + \mathbf{S}^2(\boldsymbol{\omega}_{nb}^b)\mathbf{r}_g^b] = \mathbf{f}_o^b \tag{3.17}$$

where $\mathbf{f}_c^b = \mathbf{f}_o^b$ is used–i.e., the translational motion is independent of the attack point of the external force. An alternative representation of (3.17) using vector cross products is:

$$m[\dot{\mathbf{v}}_o^b + \dot{\boldsymbol{\omega}}_{nb}^b \times \mathbf{r}_g^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times (\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b)] = \mathbf{f}_o^b \tag{3.18}$$

If the origin of the body-fixed coordinate system (x_o, y_o, z_o) is chosen to coincide with the *CG*, then we have $\mathbf{r}_g^b = [0, 0, 0]^\top$, $\mathbf{f}_o^b = \mathbf{f}_c^b$, and $\mathbf{v}_o^b = \mathbf{v}_c^b$. In this case the translational motion of the *CG* in body-fixed coordinates is:

$$m(\dot{\mathbf{v}}_c^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{v}_c^b) = \mathbf{f}_c^b \tag{3.19}$$

3.1.2 Rotational Motion (Attitude Dynamics)

The angular momenta about *CG* and *O* are defined as:

$$\vec{h}_c = \int_V (\vec{r} \times \vec{v}_p)\rho_m dV \tag{3.20}$$

$$\vec{h}_o = \int_V (\vec{r}\,' \times \vec{v}_p)\rho_m dV \tag{3.21}$$

where $\vec{v}_p$ is the velocity of the volume element dV. Using $\vec{v}_p = \vec{v}_o + \vec{\omega}_{ib} \times \vec{r}\,'$ gives:

$$\begin{aligned} \vec{h}_o &= \int_V (\vec{r}\,' \times \vec{v}_o)\rho_m dV + \int_V \vec{r}\,' \times (\vec{\omega}_{ib} \times \vec{r}\,')\rho_m dV \\ &= \int_V (\vec{r} + \vec{r}_g)\rho_m dV \times \vec{v}_o + \int_V \vec{r}\,' \times (\vec{\omega}_{ib} \times \vec{r}\,')\rho_m dV \\ &= m\vec{r}_g \times \vec{v}_o + \int_V \vec{r}\,' \times (\vec{\omega}_{ib} \times \vec{r}\,')\rho_m dV \end{aligned} \tag{3.22}$$

since $\int_V \vec{r}\rho_m dV = \vec{0}$ and $\vec{r}_g$ is constant over the volume. If we define the *inertia dyadic* about O as:

$$I_o = \int_V -S^2(\vec{r}\,')\rho_m dV \tag{3.23}$$

we get the relationship:

$$\begin{aligned} I_o\vec{\omega}_{ib} &= \int_V -S(\vec{r}\,')S(\vec{r}\,')\vec{\omega}_{ib}\rho_m dV \\ &= \int_V -\vec{r}\,' \times (\vec{r}\,' \times \vec{\omega}_{ib})\rho_m dV \\ &= \int_V \vec{r}\,' \times (\vec{\omega}_{ib} \times \vec{r}\,')\rho_m dV \end{aligned} \tag{3.24}$$

which shows that the angular momentum $\vec{h}_o$ about O can be written:

$$\vec{h}_o = I_o\vec{\omega}_{ib} + m\vec{r}_g \times \vec{v}_o \tag{3.25}$$

Next, from (3.20) and Figure 3.1, we also get:

$$\begin{aligned} \vec{h}_c &= \int_V ((\vec{r}\,' - \vec{r}_g) \times \vec{v}_p)\rho_m dV \\ &= \vec{h}_o - \vec{r}_g \times \int_V \vec{v}_p\rho_m dV \end{aligned} \tag{3.26}$$

Since the velocity of the *CG* is defined as:

$$\vec{v}_c := \frac{1}{m}\int_V \vec{v}_p\rho_m dV \tag{3.27}$$

the angular momentum about the *CG* can be written:

$$\vec{h}_c = \vec{h}_o - m\vec{r}_g \times \vec{v}_c \tag{3.28}$$

The attitude dynamics is derived by decomposing the angular momentums $\vec{h}_o$ and $\vec{h}_c$ in the b-frame. From (3.25) it is seen that:

$$\mathbf{h}_o^b = \mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{r}_g^b \times \mathbf{v}_o^b \tag{3.29}$$

$$\dot{\mathbf{h}}_o^b = \mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + m\mathbf{r}_g^b \times \dot{\mathbf{v}}_o^b \tag{3.30}$$

since $\dot{\mathbf{I}}_o = \mathbf{0}$ (constant inertia) and $\dot{\mathbf{r}}_g^b = \mathbf{0}$. Similarly, (3.28) implies that:

$$\begin{aligned} \mathbf{h}_c^b &= \mathbf{h}_o^b - m\mathbf{r}_g^b \times \mathbf{v}_c^b \\ &= \mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{r}_g^b \times \mathbf{v}_o^b - m\mathbf{r}_g^b \times \mathbf{v}_c^b \qquad (3.31) \\ \dot{\mathbf{h}}_c^b &= \mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + m\mathbf{r}_g^b \times \dot{\mathbf{v}}_o^b - m\mathbf{r}_g^b \times \dot{\mathbf{v}}_c^b \qquad (3.32) \end{aligned}$$

The b-frame equations can be transformed to the *NED* frame by using:

$$\mathbf{h}_c^n = \mathbf{R}_b^n\mathbf{h}_c^b \tag{3.33}$$

such that:

$$\begin{aligned} \dot{\mathbf{h}}_c^n &= \mathbf{R}_b^n\dot{\mathbf{h}}_c^b + \dot{\mathbf{R}}_b^n\mathbf{h}_c^b \\ &= \mathbf{R}_b^n(\dot{\mathbf{h}}_c^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{h}_c^b) \\ &= \mathbf{R}_b^n[\mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + m\mathbf{r}_g^b \times (\dot{\mathbf{v}}_o^b - \dot{\mathbf{v}}_c^b) \\ &\quad + \boldsymbol{\omega}_{nb}^b \times (\mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{r}_g^b \times (\mathbf{v}_o^b - \mathbf{v}_c^b))] \end{aligned} \tag{3.34}$$

Euler's second axiom (3.4) states that:

$$\dot{\mathbf{h}}_c^i = \mathbf{m}_c^i \tag{3.35}$$

which is approximated as:

$$\dot{\mathbf{h}}_c^n = \mathbf{m}_c^n \tag{3.36}$$

under the assumption that the *NED* coordinate system is an inertial frame. The moment about *CG* is:

$$\begin{aligned} \mathbf{m}_c^n &= \mathbf{R}_b^n(\mathbf{m}_o^b - \mathbf{r}_g^b \times \mathbf{f}_c^b) \\ &= \mathbf{R}_b^n[\mathbf{m}_o^b - \mathbf{r}_g^b \times m(\dot{\mathbf{v}}_c^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{v}_c^b)] \end{aligned} \tag{3.37}$$

where (3.19) has been applied. Euler's second axiom with (3.34) and (3.37) yields:

$$\begin{aligned} &\mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + m\mathbf{r}_g^b \times \dot{\mathbf{v}}_o^b + \boldsymbol{\omega}_{nb}^b \times (\mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{r}_g^b \times (\mathbf{v}_o^b - \mathbf{v}_c^b)) \\ &= \mathbf{m}_o^b - \mathbf{r}_g^b \times m(\boldsymbol{\omega}_{nb}^b \times \mathbf{v}_c^b) \\ &= \mathbf{m}_o^b - \mathbf{r}_g^b \times m(\boldsymbol{\omega}_{nb}^b \times (\mathbf{v}_o^b + (\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b))) \end{aligned} \tag{3.38}$$

This can be rearranged to:

$$\mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{r}_g^b \times (\dot{\mathbf{v}}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{v}_o^b) = \mathbf{m}_o^b \tag{3.39}$$

by noticing the identities:

$$\begin{aligned} \boldsymbol{\omega}_{nb}^b \times (\mathbf{r}_g^b \times (\mathbf{v}_o^b - \mathbf{v}_c^b)) &= \mathbf{r}_g^b \times (\boldsymbol{\omega}_{nb}^b \times (\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b)) \\ &\Updownarrow \\ \boldsymbol{\omega}_{nb}^b \times (\mathbf{r}_g^b \times (\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b)) &= -\mathbf{r}_g^b \times ((\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b) \times \boldsymbol{\omega}_{nb}^b) \end{aligned} \tag{3.40}$$

This follows directly from the *Jacobi identity*:

$$\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) + \mathbf{b} \times (\mathbf{c} \times \mathbf{a}) + \mathbf{c} \times (\mathbf{a} \times \mathbf{b}) = \mathbf{0} \tag{3.41}$$

which for $\mathbf{c} = \mathbf{a} \times \mathbf{b}$ reduces to:

$$\mathbf{a} \times (\mathbf{b} \times (\mathbf{a} \times \mathbf{b})) + \mathbf{b} \times ((\mathbf{a} \times \mathbf{b}) \times \mathbf{a}) = \mathbf{0} \tag{3.42}$$

since $(\mathbf{a} \times \mathbf{b}) \times (\mathbf{a} \times \mathbf{b}) = \mathbf{0}$. Hence, setting $\mathbf{a} = \boldsymbol{\omega}_{nb}^b$ and $\mathbf{b} = \mathbf{r}_g^b$ gives (3.40). The result (3.39) describes the attitude dynamics of the vessel. It can also be written in terms of the skew-symmetric matrix $\mathbf{S}$ as:

$$\mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{I}_o\boldsymbol{\omega}_{nb}^b + m\mathbf{S}(\mathbf{r}_g^b)\dot{\mathbf{v}}_o^b + m\mathbf{S}(\mathbf{r}_g^b)\mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{v}_o^b = \mathbf{m}_o^b \tag{3.43}$$

If $\mathbf{r}_g^b = [0, 0, 0]^\top$ this expression reduces to:

$$\mathbf{I}_c\dot{\boldsymbol{\omega}}_{nb}^b + \mathbf{S}(\boldsymbol{\omega}_{nb}^b)\mathbf{I}_c\boldsymbol{\omega}_{nb}^b = \mathbf{m}_c^b \tag{3.44}$$

which is referred to as *Euler's equations*.

Definition 3.1 (Inertia Matrix)
The inertia matrix $\mathbf{I}_o \in \mathbb{R}^{3\times 3}$ *about* O *is defined as:*

$$\mathbf{I}_o := \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix}, \qquad \mathbf{I}_o = \mathbf{I}_o^\top > 0 \tag{3.45}$$

where I_x, I_y, *and* I_z *are the* moments of inertia *about the* x_b, y_b, *and* z_b*-axes, and* $I_{xy} = I_{yx}, I_{xz} = I_{zx}$ *and* $I_{yz} = I_{zy}$ *are the* products of inertia *defined as:*

$$\begin{aligned} I_x &= \textstyle\int_V (y^2 + z^2)\, \rho_m dV; & I_{xy} &= \textstyle\int_V xy\, \rho_m dV = \int_V yx\, \rho_m dV = I_{yx} \\ I_y &= \textstyle\int_V (x^2 + z^2)\, \rho_m dV; & I_{xz} &= \textstyle\int_V xz\, \rho_m dV = \int_V zx\, \rho_m dV = I_{zx} \\ I_z &= \textstyle\int_V (x^2 + y^2)\, \rho_m dV; & I_{yz} &= \textstyle\int_V yz\, \rho_m dV = \int_V zy\, \rho_m dV = I_{zy} \end{aligned}$$

In practise it is convenient to compute the inertia matrix $\mathbf{I}_c \in \mathbb{R}^{3\times 3}$ about the *CG* and transform this matrix to the *b*-frame by using the *parallel axes theorem:*

Theorem 3.1 (Parallel Axes Theorem)
The inertia matrix $\mathbf{I}_o = \mathbf{I}_o^\top \in \mathbb{R}^{3\times 3}$ *about an arbitrary origin* O *is given by:*

$$\mathbf{I}_o = \mathbf{I}_c - m\mathbf{S}^2(\mathbf{r}_g^b) = \mathbf{I}_c - m(\mathbf{r}\mathbf{r}^\top - \mathbf{r}^\top\mathbf{r}\, \mathbf{I}_{3\times 3}) \tag{3.46}$$

where $\mathbf{I}_c = \mathbf{I}_c^\top \in \mathbb{R}^{3\times 3}$ *is the inertia matrix about the body's center of gravity.*

Proof.
Substituting $\vec{h}_o$ given by (3.25) into the expression for $\vec{h}_c$ in (3.28), yields:

$$\vec{h}_c = I_o\vec{\omega}_{ib} + m\vec{r}_g \times (\vec{v}_o - \vec{v}_c) \tag{3.47}$$

Using the fact that

$$\vec{v}_c = \vec{v}_o + \vec{\omega}_{ib} \times \vec{r}_g = \vec{v}_o - \vec{r}_g \times \vec{\omega}_{ib} \tag{3.48}$$

yields

$$\begin{aligned} \mathbf{h}_c^b &= \mathbf{I}_o\boldsymbol{\omega}_{ib}^b + m\mathbf{r}_g^b \times (\mathbf{r}_g^b \times \boldsymbol{\omega}_{ib}^b) \\ &= [\mathbf{I}_o + m\mathbf{S}^2(\mathbf{r}_g^b)]\boldsymbol{\omega}_{ib}^b \qquad (3.49) \\ &\Updownarrow \quad \forall\boldsymbol{\omega}_{ib}^b \\ \mathbf{h}_c^b &= \mathbf{I}_c\boldsymbol{\omega}_{ib}^b \qquad (3.50) \end{aligned}$$

Hence, it follows that:

$$\mathbf{I}_c = \mathbf{I}_o + m\mathbf{S}^2(\mathbf{r}_g^b) \tag{3.51}$$

The alternative representation in (3.46) is found by using:

$$\mathbf{S}^2(\mathbf{r}) = \mathbf{S}(\mathbf{r})\mathbf{S}(\mathbf{r}) = \mathbf{r}\mathbf{r}^\top - \mathbf{r}^\top\mathbf{r}\, \mathbf{I}_{3\times 3} \tag{3.52}$$

■

3.1.3 Rigid-Body Equations of Motion

In the previous sections it was shown how the rigid-body dynamics can be derived by applying *Newtonian* mechanics. In this section, useful properties of the nonlinear equations of motion are discussed and it is also shown how these properties considerably simplify the representation of the nonlinear model.

6 DOF Rigid-Body Equations of Motion

Equations (3.18) and (3.39) are usually written in component form according to the SNAME (1950) notation by defining:

$$\begin{array}{lcll} \mathbf{f}_o^b & = & [X,Y,Z]^\top & \text{- force decomposed in the } b\text{-frame} \\ \mathbf{m}_o^b & = & [K,M,N]^\top & \text{- moment decomposed in the } b\text{-frame} \\ \mathbf{v}_o^b & = & [u,v,w]^\top & \text{- linear velocity decomposed in the } b\text{-frame} \\ \boldsymbol{\omega}_{nb}^b & = & [p,q,r]^\top & \text{- angular velocity of the } b\text{-frame relative to the } n\text{-frame} \\ \mathbf{r}_g^b & = & [x_g,y_g,z_g]^\top & \text{- vector from O to CG decomposed in the } b\text{-frame} \end{array}$$

Applying this notation, (3.18) and (3.39) becomes:

$$\begin{aligned} m\left[\dot{u}-vr+wq-x_g(q^2+r^2)+y_g(pq-\dot{r})+z_g(pr+\dot{q})\right] &= X \\ m\left[\dot{v}-wp+ur-y_g(r^2+p^2)+z_g(qr-\dot{p})+x_g(qp+\dot{r})\right] &= Y \\ m\left[\dot{w}-uq+vp-z_g(p^2+q^2)+x_g(rp-\dot{q})+y_g(rq+\dot{p})\right] &= Z \\ I_x\dot{p}+(I_z-I_y)qr-(\dot{r}+pq)I_{xz}+(r^2-q^2)I_{yz}+(pr-\dot{q})I_{xy} & \\ \quad +m\left[y_g(\dot{w}-uq+vp)-z_g(\dot{v}-wp+ur)\right] &= K \\ I_y\dot{q}+(I_x-I_z)rp-(\dot{p}+qr)I_{xy}+(p^2-r^2)I_{zx}+(qp-\dot{r})I_{yz} & \\ \quad +m\left[z_g(\dot{u}-vr+wq)-x_g(\dot{w}-uq+vp)\right] &= M \\ I_z\dot{r}+(I_y-I_x)pq-(\dot{q}+rp)I_{yz}+(q^2-p^2)I_{xy}+(rq-\dot{p})I_{zx} & \\ \quad +m\left[x_g(\dot{v}-wp+ur)-y_g(\dot{u}-vr+wq)\right] &= N \end{aligned} \tag{3.53}$$

The first three equations represent the translational motion, while the last three equations represent the rotational motion.

Vectorial Representation

The rigid-body dynamics can be expressed in a vectorial setting as (Fossen 1991):

$$\mathbf{M}_{RB}\dot{\boldsymbol{\nu}}+\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu}=\boldsymbol{\tau}_{RB} \tag{3.54}$$

where $\boldsymbol{\nu}=[u,v,w,p,q,r]^\top$ is the generalized velocity vector decomposed in the b-frame and $\boldsymbol{\tau}_{RB}=[X,Y,Z,K,M,N]^\top$ is a generalized vector of external forces and moments.

Property 3.1 (Rigid-Body System Inertia Matrix $\mathbf{M}_{RB}$)
The representation of the rigid-body system inertia matrix $\mathbf{M}_{RB}$ is unique and it satisfies:

$$\mathbf{M}_{RB}=\mathbf{M}_{RB}^\top>0, \qquad \dot{\mathbf{M}}_{RB}=\mathbf{0}_{6\times 6}$$

where

$$\mathbf{M}_{RB} = \begin{bmatrix} m\mathbf{I}_{3\times 3} & -m\mathbf{S}(\mathbf{r}_g^b) \\ m\mathbf{S}(\mathbf{r}_g^b) & \mathbf{I}_o \end{bmatrix} = \begin{bmatrix} m & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_x & -I_{xy} & -I_{xz} \\ mz_g & 0 & -mx_g & -I_{yx} & I_y & -I_{yz} \\ -my_g & mx_g & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix} \tag{3.55}$$

Here, $\mathbf{I}_{3\times 3}$ is the identity matrix, $\mathbf{I}_o = \mathbf{I}_o^\top > 0$ is the inertia matrix according to Definition 3.1, and $\mathbf{S}(\mathbf{r}_g^b)$ is a skew-symmetric matrix according to Definition 2.2.

Matlab: Computation of $\mathbf{M}_{RB}$
The rigid-body system inertia matrix can be computed in Matlab™ as:

```
r_g = [10 0 1]'         % location of the CG with respect to O
nu  = [10 0 1 0 0 1]' % velocity vector
I_c = 10000*eye(3)      % inertia tensor
m   = 1000;             % mass

% rigid-body system inertia matrix
MRB = [ m*eye(3)       -m*Smtrx(r_g)
        m*Smtrx(r_g)    I_c          ]
```

which produces the numerical result:

$$\mathbf{M}_{RB} = \begin{bmatrix} 1000 & 0 & 0 & 0 & 1000 & 0 \\ 0 & 1000 & 0 & -1000 & 0 & 10000 \\ 0 & 0 & 1000 & 0 & -10000 & 0 \\ 0 & -1000 & 0 & 10000 & 0 & 0 \\ 1000 & 0 & -10000 & 0 & 10000 & 0 \\ 0 & 10000 & 0 & 0 & 0 & 10000 \end{bmatrix}$$

The matrix $\mathbf{C}_{RB}$ in (3.54) represents the Coriolis vector term $\boldsymbol{\omega}_{nb}^b \times \mathbf{v}_o^b$ and the centripetal vector term $\boldsymbol{\omega}_{nb}^b \times (\boldsymbol{\omega}_{nb}^b \times \mathbf{r}_g^b)$. Contrary to the representation of $\mathbf{M}_{RB}$, it is possible to find a large number of representations for the matrix $\mathbf{C}_{RB}$. We use Kirchhoff's equations to derive a *skew-symmetric* representation of $\mathbf{C}_{RB}$.

Theorem 3.2 (Coriolis-Centripetal Matrix from System Inertia Matrix)
Let $\mathbf{M}$ *be a* 6×6 *system inertia matrix defined as:*

$$\mathbf{M} = \mathbf{M}^\top = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} > 0 \tag{3.56}$$

where $\mathbf{M}_{21} = \mathbf{M}_{12}^\top$. *Then the Coriolis-centripetal matrix can always be parameterized such that* $\mathbf{C}(\boldsymbol{\nu}) = -\mathbf{C}^\top(\boldsymbol{\nu})$ *by choosing:*

$$\mathbf{C}(\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{0}_{3\times 3} & -\mathbf{S}(\mathbf{M}_{11}\boldsymbol{\nu}_1 + \mathbf{M}_{12}\boldsymbol{\nu}_2) \\ -\mathbf{S}(\mathbf{M}_{11}\boldsymbol{\nu}_1 + \mathbf{M}_{12}\boldsymbol{\nu}_2) & -\mathbf{S}(\mathbf{M}_{21}\boldsymbol{\nu}_1 + \mathbf{M}_{22}\boldsymbol{\nu}_2) \end{bmatrix} \tag{3.57}$$

where $\boldsymbol{\nu}_1 = [u, v, w]^\top$, $\boldsymbol{\nu}_2 = [p, q, r]^\top$, *and* $\mathbf{S}$ *is the cross product operator according to Definition 2.2.*

Proof. The kinetic energy T is written in the quadratic form:

$$T = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu}, \quad \mathbf{M} = \mathbf{M}^\top > 0 \tag{3.58}$$

Expanding this expression yields:

$$T = \frac{1}{2}(\boldsymbol{\nu}_1^\top \mathbf{M}_{11}\boldsymbol{\nu}_1 + \boldsymbol{\nu}_1^\top \mathbf{M}_{12}\boldsymbol{\nu}_2 + \boldsymbol{\nu}_2^\top \mathbf{M}_{21}\boldsymbol{\nu}_1 + \boldsymbol{\nu}_2^\top \mathbf{M}_{22}\boldsymbol{\nu}_2) \tag{3.59}$$

where $\mathbf{M}_{12} = \mathbf{M}_{21}^\top$ and $\mathbf{M}_{21} = \mathbf{M}_{12}^\top$. This gives:

$$\frac{\partial T}{\partial \boldsymbol{\nu}_1} = \mathbf{M}_{11}\boldsymbol{\nu}_1 + \mathbf{M}_{12}\boldsymbol{\nu}_2 \tag{3.60}$$

$$\frac{\partial T}{\partial \boldsymbol{\nu}_2} = \mathbf{M}_{21}\boldsymbol{\nu}_1 + \mathbf{M}_{22}\boldsymbol{\nu}_2 \tag{3.61}$$

Using Kirchhoff's equations, see (3.89)–(3.90) in Section 3.2.1, it is seen that there are some terms dependant on $\dot{\boldsymbol{\nu}}$ with the remaining terms due to Coriolis-centripetal forces. Hence:

$$\mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} := \begin{bmatrix} \mathbf{S}(\boldsymbol{\nu}_2)\frac{\partial T}{\partial \boldsymbol{\nu}_1} \\ \mathbf{S}(\boldsymbol{\nu}_2)\frac{\partial T}{\partial \boldsymbol{\nu}_2} + \mathbf{S}(\boldsymbol{\nu}_1)\frac{\partial T}{\partial \boldsymbol{\nu}_1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3\times 3} & -\mathbf{S}(\frac{\partial T}{\partial \boldsymbol{\nu}_1}) \\ -\mathbf{S}(\frac{\partial T}{\partial \boldsymbol{\nu}_1}) & -\mathbf{S}(\frac{\partial T}{\partial \boldsymbol{\nu}_2}) \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_1 \\ \boldsymbol{\nu}_2 \end{bmatrix}$$

which after substitution of (3.60) and (3.61) gives (3.57), see Sagatun and Fossen (1991) for the original proof of this theorem. ■

We next state some useful properties of the Coriolis and centripetal matrix $\mathbf{C}_{RB}(\boldsymbol{\nu})$:

Property 3.2 (Rigid-Body Coriolis and Centripetal Matrix $\mathbf{C}_{RB}$)
According to Theorem 3.2 the rigid-body Coriolis and centripetal matrix $\mathbf{C}_{RB}(\boldsymbol{\nu})$ *can always be represented such that* $\mathbf{C}_{RB}(\boldsymbol{\nu})$ *is skew-symmetric–i.e.:*

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = -\mathbf{C}_{RB}^\top(\boldsymbol{\nu}), \qquad \forall \boldsymbol{\nu} \in \mathbb{R}^6$$

Application of Theorem 3.2 with $\mathbf{M} = \mathbf{M}_{RB}$ *yields the following expression for* $\mathbf{C}_{RB}(\boldsymbol{\nu})$*:*

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{0}_{3\times 3} & -m\mathbf{S}(\boldsymbol{\nu}_1) - m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{r}_g^b) \\ -m\mathbf{S}(\boldsymbol{\nu}_1) - m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{r}_g^b) & m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_1)\mathbf{r}_g^b) - \mathbf{S}(\mathbf{I}_o\boldsymbol{\nu}_2) \end{bmatrix} \tag{3.62}$$

for which it is noticed that $\mathbf{S}(\boldsymbol{\nu}_1)\boldsymbol{\nu}_1 = \mathbf{0}$.

Three other useful skew-symmetric representations were derived by Fossen and Fjellstad (1995):

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{0}_{3\times 3} & -m\mathbf{S}(\boldsymbol{\nu}_1) - m\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{S}(\mathbf{r}_g^b) \\ -m\mathbf{S}(\boldsymbol{\nu}_1) + m\mathbf{S}(\mathbf{r}_g^b)\mathbf{S}(\boldsymbol{\nu}_2) & -\mathbf{S}(\mathbf{I}_o\boldsymbol{\nu}_2) \end{bmatrix} \tag{3.63}$$

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} m\mathbf{S}(\boldsymbol{\nu}_2) & -m\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{S}(\mathbf{r}_g^b) \\ m\mathbf{S}(\mathbf{r}_g^b)\mathbf{S}(\boldsymbol{\nu}_2) & -\mathbf{S}(\mathbf{I}_o\boldsymbol{\nu}_2) \end{bmatrix} \tag{3.64}$$

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} m\mathbf{S}(\boldsymbol{\nu}_2) & -m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{r}_g^b) \\ -m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_2)\mathbf{r}_g^b) & m\mathbf{S}(\mathbf{S}(\boldsymbol{\nu}_1)\mathbf{r}_g^b) - \mathbf{S}(\mathbf{I}_o\boldsymbol{\nu}_2) \end{bmatrix} \tag{3.65}$$

The first of these three expressions is written in component form according to Fossen (1991) as:

$$\mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -m(y_g q + z_g r) & m(y_g p + w) & m(z_g p - v) \\ m(x_g q - w) & -m(z_g r + x_g p) & m(z_g q + u) \\ m(x_g r + v) & m(y_g r - u) & -m(x_g p + y_g q) \end{bmatrix} \tag{3.66}$$

$$\begin{bmatrix} m(y_g q + z_g r) & -m(x_g q - w) & -m(x_g r + v) \\ -m(y_g p + w) & m(z_g r + x_g p) & -m(y_g r - u) \\ -m(z_g p - v) & -m(z_g q + u) & m(x_g p + y_g q) \\ 0 & -I_{yz}q - I_{xz}p + I_z r & I_{yz}r + I_{xy}p - I_y q \\ I_{yz}q + I_{xz}p - I_z r & 0 & -I_{xz}r - I_{xy}q + I_x p \\ -I_{yz}r - I_{xy}p + I_y q & I_{xz}r + I_{xy}q - I_x p & 0 \end{bmatrix}$$

Matlab: Computation of $\mathbf{C}_{RB}(\boldsymbol{\nu})$

Theorem 3.2 is implemented in the Matlab GNC toolbox in the script `m2c.m`. The following example demonstrates how $\mathbf{C}_{RB}(\boldsymbol{\nu})$ can be computed numerically using this script:

```
% rigid-body system inertia matrix
MRB = [1000*eye(3)  zeros(3,3)
       zeros(3,3)   10000*eye(3)];

% rigid-body Coriolis and centripetal matrix
nu  = [10 1 1 1 2 3]'
CRB = m2c(MRB,nu)
```

which produces the numerical result:

$$\mathbf{C}_{RB} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1000 & -1000 \\ 0 & 0 & 0 & -1000 & 0 & 10000 \\ 0 & 0 & 0 & 1000 & -10000 & 0 \\ 0 & 1000 & -1000 & 0 & 30000 & -20000 \\ -1000 & 0 & 10000 & -30000 & 0 & 10000 \\ 1000 & -10000 & 0 & 20000 & -10000 & 0 \end{bmatrix}$$

Simplified 6 DOF Rigid-Body Equations of Motion

The rigid-body equations of motion can be simplified by choosing the origin of the body-fixed coordinate system according to the following criteria:

(1) Origin O coincides with the CG: This implies that $\mathbf{r}_g^b = [0,0,0]^\top$, $\mathbf{I}_c = \mathbf{I}_o$ (see Theorem 3.1), and:

$$\mathbf{M}_{RB} = \begin{bmatrix} m\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{I}_c \end{bmatrix} \tag{3.67}$$

A further simplification is obtained when the body axes (x_b, y_b, z_b) coincide with the principal axes of inertia. This implies that $\mathbf{I}_c = \mathbf{I}_o = \text{diag}\{I_x, I_y, I_z\}$.

(2) Rotation of the body axes such that $\mathbf{I}_o$ becomes diagonal: The body-fixed coordinate system (x_b, y_b, z_b) can be rotated about its axes to obtain a diagonal inertia matrix by simply performing a *principal axis transformation*. The eigenvalues λ_i $(i = 1, 2, 3)$ of the inertia matrix $\mathbf{I}_o$ are found from the characteristic equation:

$$\det(\lambda \mathbf{I}_{3\times 3} - \mathbf{I}_o) = \lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0 \tag{3.68}$$

where $\mathbf{I}_{3\times 3}$ is the identity matrix. The modal matrix $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3]$ is obtained from the right eigenvectors $\mathbf{h}_i$ such that:

$$(\lambda_i \mathbf{I}_{3\times 3} - \mathbf{I}_o)\mathbf{h}_i = \mathbf{0}; \qquad (i = 1, 2, 3) \tag{3.69}$$

Consequently, the coordinate system (x_b, y_b, z_b) should be rotated about its axes to form a new coordinate system (x_b', y_b', z_b') with unit vectors:

$$\mathbf{e}_x' = \mathbf{H}\mathbf{e}_x; \qquad \mathbf{e}_y' = \mathbf{H}\mathbf{e}_y; \qquad \mathbf{e}_z' = \mathbf{H}\mathbf{e}_z \tag{3.70}$$

Here, $\mathbf{e}_x$, $\mathbf{e}_y$ and $\mathbf{e}_z$ are the unit vectors corresponding to (x_b, y_b, z_b). The result is that the new inertia matrix $\mathbf{I}_o'$ will be diagonal, that is:

$$\mathbf{I}_o' = \text{diag}\{I_x', I_y, I_z'\} = \text{diag}\{\lambda_1, \lambda_2, \lambda_3\} \tag{3.71}$$

The disadvantage with this approach is that the new coordinate system will differ from the longitudinal, lateral, and normal symmetry axes of the vessel. This can be compensated for in the control design by transforming the desired state trajectory to the (x_b', y_b', z_b') system. Applying these results to (3.53) yields the following simple representation:

$$\begin{aligned} m(\dot{u} - vr + wq) &= X; & I_x\dot{p} + (I_z - I_y)qr &= K \\ m(\dot{v} - wp + ur) &= Y; & I_y\dot{q} + (I_x - I_z)rp &= M \\ m(\dot{w} - uq + vp) &= Z; & I_z\dot{r} + (I_y - I_x)pq &= N \end{aligned} \tag{3.72}$$

(3) Translation of the origin O such that $\mathbf{I}_o$ becomes diagonal: It is often convenient to let the body axes coincide with the principal axes of inertia or the longitudinal, lateral, and normal symmetry axes of the vessel. The origin of the body-fixed coordinate system can then be chosen such that the inertia matrix of the body-fixed coordinate system will be diagonal, that is $\mathbf{I}_o = \text{diag}\{I_x, I_y, I_z\}$, by applying the parallel axes theorem;

see Theorem 3.1. Expanding (3.46) with $\mathbf{I}_o = \text{diag}\{I_x, I_y, I_z\}$ and $\mathbf{I}_c$ as a full matrix, yields the following set of equations:

$$\begin{aligned} I_x &= I_x^{cg} + m(y_g^2 + z_g^2) \\ I_y &= I_y^{cg} + m(x_g^2 + z_g^2) \\ I_z &= I_z^{cg} + m(x_g^2 + y_g^2) \end{aligned} \tag{3.73}$$

where x_g, y_g and z_g must be chosen such that:

$$\begin{aligned} mI_{yz}^{cg} x_g^2 &= -I_{xy}^{cg} I_{xz}^{cg} \\ mI_{xz}^{cg} y_g^2 &= -I_{xy}^{cg} I_{yz}^{cg} \\ mI_{xy}^{cg} z_g^2 &= -I_{xz}^{cg} I_{yz}^{cg} \end{aligned} \tag{3.74}$$

are satisfied. Hence, the rigid-body equations of motion can be expressed as:

$$\begin{aligned} m\left[\dot{u} - vr + wq - x_g(q^2 + r^2) + y_g(pq - \dot{r}) + z_g(pr + \dot{q})\right] &= X \\ m\left[\dot{v} - wp + ur - y_g(r^2 + p^2) + z_g(qr - \dot{p}) + x_g(qp + \dot{r})\right] &= Y \\ m\left[\dot{w} - uq + vp - z_g(p^2 + q^2) + x_g(rp - \dot{q}) + y_g(rq + \dot{p})\right] &= Z \\ I_x\dot{p} + (I_z - I_y)qr + m\left[y_g(\dot{w} - uq + vp) - z_g(\dot{v} - wp + ur)\right] &= K \\ I_y\dot{q} + (I_x - I_z)rp + m\left[z_g(\dot{u} - vr + wq) - x_g(\dot{w} - uq + vp)\right] &= M \\ I_z\dot{r} + (I_y - I_x)pq + m\left[x_g(\dot{v} - wp + ur) - y_g(\dot{u} - vr + wq)\right] &= N \end{aligned} \tag{3.75}$$

3.2 Hydrodynamic Forces and Moments

In hydrodynamics it is common to assume that the hydrodynamic forces and moments on a rigid body can be linearly superimposed (see Faltinsen 1990).

Radiation-Induced Forces

An important contribution to the hydrodynamic forces and moments are (Faltinsen 1990):

> *"Forces on the body when the body is forced to oscillate with the wave excitation frequency and there are no incident waves"*

The radiation-induced forces and moments can be identified as the sum of three new components:

(1) *Added mass* due to the inertia of the surrounding fluid.

(2) *Radiation-induced potential damping* due to the energy carried away by generated surface waves.

(3) *Restoring forces* due to Archimedes (weight and buoyancy).

The contribution from these three components can be expressed mathematically as:

$$\boldsymbol{\tau}_R = -\underbrace{\mathbf{M}_A\dot{\boldsymbol{\nu}} - \mathbf{C}_A(\boldsymbol{\nu})\boldsymbol{\nu}}_{\text{added mass}} - \underbrace{\mathbf{D}_P(\boldsymbol{\nu})\boldsymbol{\nu}}_{\text{potential damping}} - \underbrace{\mathbf{g}(\boldsymbol{\eta}) + \mathbf{g}_o}_{\text{restoring forces}} \tag{3.76}$$

In addition to potential damping we have to include other damping effects like skin friction, wave drift damping, and damping due to vortex shedding, that is:

$$\boldsymbol{\tau}_D = -\underbrace{\mathbf{D}_S(\boldsymbol{\nu})\boldsymbol{\nu}}_{\substack{\text{skin}\\ \text{friction}}} - \underbrace{\mathbf{D}_W(\boldsymbol{\nu})\boldsymbol{\nu}}_{\substack{\text{wave drift}\\ \text{damping}}} - \underbrace{\mathbf{D}_M(\boldsymbol{\nu})\boldsymbol{\nu}}_{\substack{\text{damping due to}\\ \text{vortex shedding}}} \tag{3.77}$$

Defining the total hydrodynamic damping matrix $\mathbf{D}(\boldsymbol{\nu})$ as:

$$\mathbf{D}(\boldsymbol{\nu}) := \mathbf{D}_P(\boldsymbol{\nu}) + \mathbf{D}_S(\boldsymbol{\nu}) + \mathbf{D}_W(\boldsymbol{\nu}) + \mathbf{D}_M(\boldsymbol{\nu}) \tag{3.78}$$

implies that the hydrodynamic forces and moments $\boldsymbol{\tau}_H$ can be written as the sum of $\boldsymbol{\tau}_R$ and $\boldsymbol{\tau}_D$, that is:

$$\boldsymbol{\tau}_H = -\mathbf{M}_A\dot{\boldsymbol{\nu}} - \mathbf{C}_A(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{g}(\boldsymbol{\eta}) + \mathbf{g}_o \tag{3.79}$$

Environmental Disturbances

In addition to the hydrodynamic forces and moments $\boldsymbol{\tau}_H$, the vessel will be exposed to environmental forces. These are:

- wind
- waves
- currents

The resulting environmental force and moment vector is denoted as $\mathbf{w}$. Simple models for wind, waves, and current disturbances applicable to control system design are presented in Chapter 4. A more general discussion on marine hydrodynamics is found in Faltinsen (1990), Newman (1977), and Sarpkaya (1981).

Resulting Model

The resulting rigid-body dynamics is then expressed as (see Section 3.1.3):

$$\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_{RB} \tag{3.80}$$

where:

$$\boldsymbol{\tau}_{RB} = \boldsymbol{\tau}_H + \mathbf{w} + \boldsymbol{\tau} \tag{3.81}$$

The vector $\boldsymbol{\tau}$ represents the *propulsion* forces and moments. The resulting model is then given by:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \mathbf{g}_o + \mathbf{w} + \boldsymbol{\tau} \tag{3.82}$$

where:

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_{RB} + \mathbf{M}_A \\ \mathbf{C}(\boldsymbol{\nu}) &= \mathbf{C}_{RB}(\boldsymbol{\nu}) + \mathbf{C}_A(\boldsymbol{\nu}) \end{aligned}$$

The terms in (3.82) will now be discussed in more detail.

3.2.1 Added Mass and Inertia

In the previous section, it was shown that the rigid body dynamics of a marine vessel can be derived by applying the *Newtonian* formulation. As for the rigid-body dynamics, it is advantageous to separate the added mass forces and moments in terms which belong to the *added mass system inertia matrix* $\mathbf{M}_A$ and a matrix of hydrodynamic Coriolis and centripetal terms denoted $\mathbf{C}_A(\nu)$. To derive the expressions for these two matrices, an *energy approach* based on Kirchhoff's equations is applied.

Lagrangian Mechanics

An alternative approach to the *Newton–Euler formulation* is to apply *Lagrangian mechanics*. The Lagrangian approach involves three basic steps. First, suitable expressions for the vessel's kinetic and potential energies, denoted T and V respectively, must be fomulated. Then the Lagrangian L is given by:

$$L = T - V \tag{3.83}$$

Finally, the *Euler-Lagrange equation* is:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\boldsymbol{\eta}}}\right) - \frac{\partial L}{\partial \boldsymbol{\eta}} = \mathbf{J}^{-\top}(\boldsymbol{\eta})\boldsymbol{\tau} \tag{3.84}$$

which in component form corresponds to a set of 6 second-order differential equations. From the above formula it is seen that the Lagrangian mechanics describes the system dynamics in terms of energy. Formula (3.84) is valid in any reference frame, inertial and body-fixed, as long as *generalized coordinates* are used.

For a vessel not subject to any motion constraints, the number of independent (*generalized*) coordinates is equal to the number of DOF. For a marine vessel moving in 6 DOF the generalized coordinates can be chosen as (NED reference frame):

$$\boldsymbol{\eta} = [n, e, d, \phi, \theta, \psi]^\top \tag{3.85}$$

It should be noted that the alternative representation:

$$\boldsymbol{\eta} = [n, e, d, \eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top \tag{3.86}$$

using unit quaternions cannot be used in a Lagrangian approach since this representation is defined by 7 parameters. Hence, these parameters are not *generalized coordinates*. Often it is advantageous to formulate the equations of motion in a body-fixed reference frame. Unfortunately, the body-fixed velocity vector:

$$\boldsymbol{\nu} = [u, v, w, p, q, r]^\top \tag{3.87}$$

cannot be integrated to yield a set of generalized coordinates in terms of position and orientation since $\int_0^t \boldsymbol{\nu} d\tau$ has no immediate physical interpretation. Consequently, the Lagrange equation cannot be directly used to formulate the equations of motion in the body-fixed coordinate system. However, this problem can be circumvented by applying Kirchhoff's equations of motion, or the so-called *Quasi-Lagrangian* approach; see Meirovitch and Kwak (1989) for details.

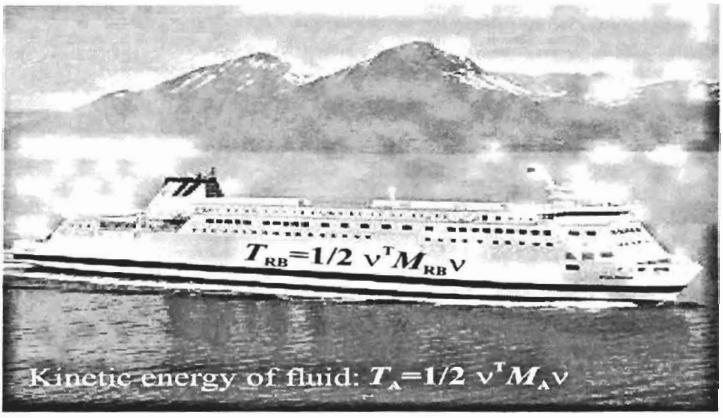


Figure 3.2: Rigid-body and fluid kinetic energy (ocean surrounding the vehicle).

Kirchhoff's Equations in Vector Form (Kirchhoff 1869)

Consider a vessel with body-fixed linear velocity $\boldsymbol{\nu}_1 = [u, v, w]^\top$ and angular velocity $\boldsymbol{\nu}_2 = [p, q, r]^\top$. Hence, the force $\boldsymbol{\tau}_1 = [X, Y, Z]^\top$ and moment $\boldsymbol{\tau}_2 = [K, M, N]^\top$ are related to the kinetic energy:

$$T = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu} \tag{3.88}$$

by the vector equations:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \boldsymbol{\nu}_1}\right) + \mathbf{S}(\boldsymbol{\nu}_2)\frac{\partial T}{\partial \boldsymbol{\nu}_1} = \boldsymbol{\tau}_1 \tag{3.89}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \boldsymbol{\nu}_2}\right) + \mathbf{S}(\boldsymbol{\nu}_2)\frac{\partial T}{\partial \boldsymbol{\nu}_2} + \mathbf{S}(\boldsymbol{\nu}_1)\frac{\partial T}{\partial \boldsymbol{\nu}_1} = \boldsymbol{\tau}_2 \tag{3.90}$$

where $\mathbf{S}$ is the skew-symmetric cross-product operator in Definition 2.2. *Kirchhoff's equations* will prove to be very useful in the derivation of the expression for added inertia. Notice that Kirchhoff's equations do not include the gravitational forces.

Added Mass

The concept of added mass is usually misunderstood as a finite amount of water connected to the vessel, such that the vessel body and the fluid represent a new system with a mass larger than the original system. This is *not* true, since the vessel motion would force the whole fluid to oscillate with different fluid particle amplitudes, in phase with the forced harmonic motion of the vessel. However, the amplitudes will decay far away from the body and may therefore be negligible.

Added (virtual) mass can also be understood as pressure-induced forces and moments due to a forced harmonic motion of the vessel body proportional to its acceleration. Consequently, the added mass forces and the acceleration will be 180 degrees out of phase with the forced harmonic motion.

Fluid Kinetic Energy

For completely submerged vessels it will be assumed that the added mass coefficients are constant and thus independent of the wave circular frequency. Together with this assumption, the concept of fluid kinetic energy is used to derive the added mass terms. Moreover, any motion of the vessel will induce a motion in the otherwise stationary fluid. In order to allow the vessel to pass through the fluid, it must move aside and then close behind the vessel. As a consequence, the fluid motion possesses kinetic energy that it would lack otherwise.

The expression for the fluid kinetic energy T_A, see Lamb (1932), is written as a quadratic form in the body axis velocity vector:

$$T_A = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}_A \boldsymbol{\nu} \tag{3.91}$$

where $\mathbf{M}_A$ is the 6×6 *system inertia matrix* of added mass terms:

$$\mathbf{M}_A = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \qquad (\mathbf{A}_{ij} \in \mathbb{R}^{3\times 3}) \tag{3.92}$$

$$= -\begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \tag{3.93}$$

The notation of SNAME (1950) is used in this expression; for instance the hydrodynamic added mass force Y along the y-axis due to an acceleration $\dot{u}$ in the x-direction is written as:

$$Y = -Y_{\dot{u}}\dot{u} \tag{3.94}$$

where

$$Y_{\dot{u}} := \frac{\partial Y}{\partial \dot{u}} \tag{3.95}$$

In some textbooks the notation $A_{ij} = -\{M_A\}_{ij}$ is used instead. This implies that $A_{21} = -Y_{\dot{u}}$ in the example above. It should be noted that the hydrodynamic derivatives $A_{11} = -X_{\dot{u}}, A_{22} = -Y_{\dot{v}}, A_{33} = -Z_{\dot{w}}, A_{44} = -K_{\dot{p}}, A_{55} = -M_{\dot{q}}$ and $A_{66} = -N_{\dot{r}}$, corresponding to the diagonal, will all be positive for most applications. However at certain frequencies, negative added mass values have been documented for catamarans, bulb sections, and submerged body sections close to the free surface. For completely submerged vessels $\mathbf{M}_A$ will always be strictly positive, that is $\mathbf{M}_A > 0$.

Expanding (3.91) under the assumption that $\mathbf{M}_A = \mathbf{M}_A^\top$, yields the component form expression:

$$\begin{aligned} 2T_A = \; & -X_{\dot{u}}u^2 - Y_{\dot{v}}v^2 - Z_{\dot{w}}w^2 - 2Y_{\dot{w}}vw - 2X_{\dot{w}}wu - 2X_{\dot{v}}uv \\ & -K_{\dot{p}}p^2 - M_{\dot{q}}q^2 - N_{\dot{r}}r^2 - 2M_{\dot{r}}qr - 2K_{\dot{r}}rp - 2K_{\dot{q}}pq \\ & -2p(X_{\dot{p}}u + Y_{\dot{p}}v + Z_{\dot{p}}w) \\ & -2q(X_{\dot{q}}u + Y_{\dot{q}}v + Z_{\dot{q}}w) \\ & -2r(X_{\dot{r}}u + Y_{\dot{r}}v + Z_{\dot{r}}w) \end{aligned} \tag{3.96}$$

where the kinetic energy T_A is just the right-hand-side divided by 2.

Property 3.3 (Hydrodynamic System Inertia Matrix $\mathbf{M}_A$)
For a rigid-body at rest ($U \approx 0$), and under the assumption of an ideal fluid, no incident waves, no sea currents, and zero frequency, the hydrodynamic system inertia matrix is positive definite:

$$\mathbf{M}_A = \mathbf{M}_A^\top > 0$$

Proof. Newman (1977). ■

Remark 1: In a real fluid the 36 elements of $\mathbf{M}_A$ may all be distinct but still $\mathbf{M}_A > 0$. Experience has shown that the numerical values of the added mass derivatives in a real fluid are usually in good agreement with those obtained from ideal theory (see Wendel 1956). Hence, $\mathbf{M}_A = \mathbf{M}_A^\top > 0$ is a good approximation in marine control systems.

Remark 2: It should be noted that for surface ships moving with a speed $U \gg 0$ in waves, Salvesen *et al.* (1970) have shown by applying strip theory that:

$$\mathbf{M}_A(U) \neq \mathbf{M}_A^\top(U) \tag{3.97}$$

However, for underwater vehicles (ROVs) and foilborne catamarans operating outside the wave-affected zone, symmetry and frequency independence have shown to be reasonable assumptions. This is also a good approximation for positioned ships since $U \approx 0$.

Consider a symmetrical added inertia matrix having 21 distinct hydrodynamic derivatives. The added mass forces and moments can be derived by applying potential theory. This method is based on the assumptions of an inviscid fluid, no circulation and a completely submerged body in an unbounded fluid. The last assumption is violated at the seabed, near underwater installations, and at the surface. However, this is not a practical problem since *double-body* theory can be applied (Faltinsen 1990).

Added Mass Forces and Moments

Based on the kinetic energy T_A of the fluid, it is straightforward to derive the added mass forces and moments. This is usually done by application of *Kirchhoff's equations* (Kirchhoff 1869), which simply relates the fluid energy to the forces and moments acting on the vessel. Consider *Kirchhoff's equations* in component form (see Milne-Thomson 1968):

$$\begin{aligned}
\frac{d}{dt}\frac{\partial T_A}{\partial u} &= r\frac{\partial T_A}{\partial v} - q\frac{\partial T_A}{\partial w} - X_A \\
\frac{d}{dt}\frac{\partial T_A}{\partial v} &= p\frac{\partial T_A}{\partial w} - r\frac{\partial T_A}{\partial u} - Y_A \\
\frac{d}{dt}\frac{\partial T_A}{\partial w} &= q\frac{\partial T_A}{\partial u} - p\frac{\partial T_A}{\partial v} - Z_A \\
\frac{d}{dt}\frac{\partial T_A}{\partial p} &= w\frac{\partial T_A}{\partial v} - v\frac{\partial T_A}{\partial w} + r\frac{\partial T_A}{\partial q} - q\frac{\partial T_A}{\partial r} e - K_A \\
\frac{d}{dt}\frac{\partial T_A}{\partial q} &= u\frac{\partial T_A}{\partial w} - w\frac{\partial T_A}{\partial u} + p\frac{\partial T_A}{\partial r} - r\frac{\partial T_A}{\partial p} - M_A \\
\frac{d}{dt}\frac{\partial T_A}{\partial r} &= v\frac{\partial T_A}{\partial u} - u\frac{\partial T_A}{\partial v} + q\frac{\partial T_A}{\partial p} - p\frac{\partial T_A}{\partial q} - N_A
\end{aligned} \tag{3.98}$$

Substituting (3.96) into (3.98) gives the following expressions for the added mass terms (Imlay 1961):

$$
\begin{aligned}
X_A &= X_{\dot{u}}\dot{u} + X_{\dot{w}}(\dot{w} + uq) + X_{\dot{q}}\dot{q} + Z_{\dot{w}}wq + Z_{\dot{q}}q^2 \\
&\quad + X_{\dot{v}}\dot{v} + X_{\dot{p}}\dot{p} + X_{\dot{r}}\dot{r} - Y_{\dot{v}}vr - Y_{\dot{p}}rp - Y_{\dot{r}}r^2 \\
&\quad - X_{\dot{v}}ur - Y_{\dot{w}}wr \\
&\quad + Y_{\dot{w}}vq + Z_{\dot{p}}pq - (Y_{\dot{q}} - Z_{\dot{r}})qr \\
Y_A &= X_{\dot{v}}\dot{u} + Y_{\dot{w}}\dot{w} + Y_{\dot{q}}\dot{q} \\
&\quad + Y_{\dot{v}}\dot{v} + Y_{\dot{p}}\dot{p} + Y_{\dot{r}}\dot{r} + X_{\dot{v}}vr - Y_{\dot{w}}vp + X_{\dot{r}}r^2 + (X_{\dot{p}} - Z_{\dot{r}})rp - Z_{\dot{p}}p^2 \\
&\quad - X_{\dot{w}}(up - wr) + X_{\dot{u}}ur - Z_{\dot{w}}wp \\
&\quad - Z_{\dot{q}}pq + X_{\dot{q}}qr \\
Z_A &= X_{\dot{w}}(\dot{u} - wq) + Z_{\dot{w}}\dot{w} + Z_{\dot{q}}\dot{q} - X_{\dot{u}}uq - X_{\dot{q}}q^2 \\
&\quad + Y_{\dot{w}}\dot{v} + Z_{\dot{p}}\dot{p} + Z_{\dot{r}}\dot{r} + Y_{\dot{v}}vp + Y_{\dot{r}}rp + Y_{\dot{p}}p^2 \\
&\quad + X_{\dot{v}}up + Y_{\dot{w}}wp \\
&\quad - X_{\dot{v}}vq - (X_{\dot{p}} - Y_{\dot{q}})pq - X_{\dot{r}}qr \\
K_A &= X_{\dot{p}}\dot{u} + Z_{\dot{p}}\dot{w} + K_{\dot{q}}\dot{q} - X_{\dot{v}}wu + X_{\dot{r}}uq - Y_{\dot{w}}w^2 - (Y_{\dot{q}} - Z_{\dot{r}})wq + M_{\dot{r}}q^2 \\
&\quad + Y_{\dot{p}}\dot{v} + K_{\dot{p}}\dot{p} + K_{\dot{r}}\dot{r} + Y_{\dot{w}}v^2 - (Y_{\dot{q}} - Z_{\dot{r}})vr + Z_{\dot{p}}vp - M_{\dot{r}}r^2 - K_{\dot{q}}rp \\
&\quad + X_{\dot{w}}uv - (Y_{\dot{v}} - Z_{\dot{w}})vw - (Y_{\dot{r}} + Z_{\dot{q}})wr - Y_{\dot{p}}wp - X_{\dot{q}}ur \\
&\quad + (Y_{\dot{r}} + Z_{\dot{q}})vq + K_{\dot{r}}pq - (M_{\dot{q}} - N_{\dot{r}})qr \\
M_A &= X_{\dot{q}}(\dot{u} + wq) + Z_{\dot{q}}(\dot{w} - uq) + M_{\dot{q}}\dot{q} - X_{\dot{w}}(u^2 - w^2) - (Z_{\dot{w}} - X_{\dot{u}})wu \\
&\quad + Y_{\dot{q}}\dot{v} + K_{\dot{q}}\dot{p} + M_{\dot{r}}\dot{r} + Y_{\dot{p}}vr - Y_{\dot{r}}vp - K_{\dot{r}}(p^2 - r^2) + (K_{\dot{p}} - N_{\dot{r}})rp \\
&\quad - Y_{\dot{w}}uv + X_{\dot{v}}vw - (X_{\dot{r}} + Z_{\dot{p}})(up - wr) + (X_{\dot{p}} - Z_{\dot{r}})(wp + ur) \\
&\quad - M_{\dot{r}}pq + K_{\dot{q}}qr \\
N_A &= X_{\dot{r}}\dot{u} + Z_{\dot{r}}\dot{w} + M_{\dot{r}}\dot{q} + X_{\dot{v}}u^2 + Y_{\dot{w}}wu - (X_{\dot{p}} - Y_{\dot{q}})uq - Z_{\dot{p}}wq - K_{\dot{q}}q^2 \\
&\quad + Y_{\dot{r}}\dot{v} + K_{\dot{r}}\dot{p} + N_{\dot{r}}\dot{r} - X_{\dot{v}}v^2 - X_{\dot{r}}vr - (X_{\dot{p}} - Y_{\dot{q}})vp + M_{\dot{r}}rp + K_{\dot{q}}p^2 \\
&\quad - (X_{\dot{u}} - Y_{\dot{v}})uv - X_{\dot{w}}vw + (X_{\dot{q}} + Y_{\dot{p}})up + Y_{\dot{r}}ur + Z_{\dot{q}}wp \\
&\quad - (X_{\dot{q}} + Y_{\dot{p}})vq - (K_{\dot{p}} - M_{\dot{q}})pq - K_{\dot{r}}qr
\end{aligned}
\tag{3.99}
$$

Imlay (1961) arranged the equations in four lines with longitudinal components on the first line and lateral components on the second. The third line consists of mixed terms involving u or w as one factor. If one or both of these velocities are large enough to be treated as constants, the third line may be treated as an additional term to the lateral equations of motion. The fourth line contains mixed terms that usually can be neglected as second order terms.

It should be noted that the off-diagonal elements of $\mathbf{M}_A$ will be small compared to the diagonal elements for most practical applications. A more detailed discussion on the different added mass derivatives can be found in Humphreys and Watkinson (1978).

Property 3.4 (Hydrodynamic Coriolis and centripetal matrix $\mathbf{C}_A$)
For a rigid-body moving through an ideal fluid the hydrodynamic Coriolis and centripetal matrix $\mathbf{C}_A(\boldsymbol{\nu})$ can always be parameterized such that it is skew-symmetric:

$$\mathbf{C}_A(\boldsymbol{\nu}) = -\mathbf{C}_A^\top(\boldsymbol{\nu}), \qquad \forall \boldsymbol{\nu} \in \mathbb{R}^6$$

by defining:

$$\mathbf{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{0}_{3\times 3} & -\mathbf{S}(\mathbf{A}_{11}\boldsymbol{\nu}_1 + \mathbf{A}_{12}\boldsymbol{\nu}_2) \\ -\mathbf{S}(\mathbf{A}_{11}\boldsymbol{\nu}_1 + \mathbf{A}_{12}\boldsymbol{\nu}_2) & -\mathbf{S}(\mathbf{A}_{21}\boldsymbol{\nu}_1 + \mathbf{A}_{22}\boldsymbol{\nu}_2 \end{bmatrix} \tag{3.100}$$

where $\mathbf{A}_{ij}$ $(i,j = 1,2)$ *are defined in (3.92).*

Proof. Substituting:

$$\mathbf{M} = \mathbf{M}_A = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \tag{3.101}$$

into (3.57) in Theorem 3.2 directly proves (3.100). ■

Formula (3.100) can be written in component form according to:

$$\mathbf{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix} \tag{3.102}$$

where

$$\begin{aligned} a_1 &= X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r \\ a_2 &= Y_{\dot{u}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r \\ a_3 &= Z_{\dot{u}}u + Z_{\dot{v}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r \\ b_1 &= K_{\dot{u}}u + K_{\dot{v}}v + K_{\dot{w}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r \\ b_2 &= M_{\dot{u}}u + M_{\dot{v}}v + M_{\dot{w}}w + M_{\dot{p}}p + M_{\dot{q}}q + M_{\dot{r}}r \\ b_3 &= N_{\dot{u}}u + N_{\dot{v}}v + N_{\dot{w}}w + N_{\dot{p}}p + N_{\dot{q}}q + N_{\dot{r}}r \end{aligned} \tag{3.103}$$

Example 3.1 (Added Mass for Surface Vessels)
For surface ships like tankers, cargo ships, cruise-liners, etc., it is common to decouple the surge mode from the steering dynamics due to xz-plane symmetry. Similarly, the heave, pitch, and roll modes are neglected under the assumption that these motion variables are small. Hence, we define $\boldsymbol{\nu} = [u, v, r]^\top$. This implies that the contribution from the added mass derivatives on a surface ship moving with forward speed $U \gg 0$ is:

$$\mathbf{M}_A \neq \mathbf{M}_A^\top = -\begin{bmatrix} X_{\dot{u}} & 0 & 0 \\ 0 & Y_{\dot{v}} & Y_{\dot{r}} \\ 0 & N_{\dot{v}} & N_{\dot{r}} \end{bmatrix}$$

$$\mathbf{C}_A(\boldsymbol{\nu}) = -\mathbf{C}_A^\top(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -Y_{\dot{v}}v - \frac{Y_{\dot{r}}+N_{\dot{v}}}{2}r \\ 0 & 0 & X_{\dot{u}}u \\ Y_{\dot{v}}v + \frac{Y_{\dot{r}}+N_{\dot{v}}}{2}r & -X_{\dot{u}}u & 0 \end{bmatrix}$$

For ship positioning, $U \approx 0$ and therefore $\mathbf{M}_A = \mathbf{M}_A^\top$. Hence, we can replace $N_{\dot{v}}$ with $Y_{\dot{r}}$

in the above expression which yields:

$$\mathbf{M}_A = \mathbf{M}_A^\top = -\begin{bmatrix} X_{\dot{u}} & 0 & 0 \\ 0 & Y_{\dot{v}} & Y_{\dot{r}} \\ 0 & Y_{\dot{r}} & N_{\dot{r}} \end{bmatrix}$$

$$\mathbf{C}_A(\boldsymbol{\nu}) = -\mathbf{C}_A^\top(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -(Y_{\dot{v}}v + Y_{\dot{r}}r) \\ 0 & 0 & X_{\dot{u}}u \\ Y_{\dot{v}}v + Y_{\dot{r}}r & -X_{\dot{u}}u & 0 \end{bmatrix}$$

Example 3.2 (Added Mass for Underwater Vehicles)
In general, the motion of an underwater vehicle moving in 6 DOF at high speed will be highly nonlinear and coupled. However, in many ROV applications the vehicle will only be allowed to move at low speed. If the vehicle also has three planes of symmetry, this suggests that the contribution from the off-diagonal elements in the matrix $\mathbf{M}_A$ *can be neglected. Hence, the following simple expressions for the matrices* $\mathbf{M}_A$ *and* $\mathbf{C}_A$ *are obtained:*

$$\mathbf{M}_A = \mathbf{M}_A^\top = -\text{diag}\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\} \tag{3.104}$$

$$\mathbf{C}_A(\boldsymbol{\nu}) = -\mathbf{C}_A^\top(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \tag{3.105}$$

The diagonal structure is highly attractive, since off-diagonal elements are difficult to determine from experiments as well as theory. In practice, the diagonal approximation is found to be quite good for many applications. This is due to the fact that the off-diagonal elements of a positive inertia matrix will be much smaller than their diagonal counterparts.

Strip Theory

For slender bodies an estimate of the hydrodynamic derivatives can be obtained by applying *strip theory* (Newman 1977). The principle of strip theory involves dividing the submerged part of the vessel into a finite number of strips. Hence, two-dimensional hydrodynamic coefficients for added mass can be computed for each strip and then summated over the length of the body to yield the three-dimensional coefficients.

A more general discussion on added mass derivatives for bodies of various geometries is found in Imlay (1961). Other useful references discussing methods for computation of the added mass derivatives are Humphreys and Watkinson (1978), and Triantafyllou and Amzallag (1984).

Hydrodynamic Computation Programs

WAMIT has become the de facto industry standard among oil and engineering companies for numerical computation of the transfer function and phase between the vessel and the waves for given wave directions and frequencies. WAMIT is a computer program based on a three dimensional panel method for analyzing hydrodynamic interactions with floating or

submerged bodies in the presence of ocean waves using potential theory. Besides the wave-induced transfer functions, the 6 DOF hydrodynamic added system inertia matrix $\mathbf{M}_A$, potential damping matrix $\mathbf{D}_P$, and hydrostatic matrix $\mathbf{G}$ are available from WAMIT. For more detailed information regarding the capabilities of WAMIT, please consult the user manual on http://www.wamit.com or the reference available at the same site.

Marintek AS (http://www.marintek.sintef.no) offers several software programs for numerical hydrodynamics. The most useful programs for station-keeping, maneuvering, sealoads, and control designs are:

- *RESPONSE* - a PC-based program for analysis of wave-induced motion of a vessel. Motion parameters or seafastening forces at specified locations are calculated based on 6 DOF motion transfer functions.
- *SIMAN* - Maneuvering prediction program that computes the rigid-body and hydrodynamic forces on a marine craft in transit.
- *MIMOSA* - Analysis program for mooring systems and moored vessels. Efficient frequency domain techniques are used to calculate low and wave frequency vessel motions and mooring tensions.
- *SIMO* - (Simulation of complex marine operations) Time domain simulation program for multibody systems, allowing nonlinear effects to be included in the wave frequency range. Flexible modeling of stationkeeping forces and connecting force mechanisms (anchor lines, ropes, thrusters) are included.
- *WAVERES* - Nonlinear potential flow wave resistance calculation using built-in, fast and easy element generation.

3.2.2 Hydrodynamic Damping

As mentioned in Section 3.2.1 hydrodynamic damping for marine vessels is mainly caused by:

Potential Damping: We recall from the beginning of Section 3.2 that *added mass, damping* and *restoring* forces and moments are encountered when a body is forced to oscillate with the wave excitation frequency in the absence of incident waves. The radiation-induced damping term is usually referred to as *potential damping*. However, the contribution from the potential damping terms compared to other dissipative terms like viscous damping are usually negligible.

Skin Friction: Linear skin friction due to laminar boundary layer theory is important when considering the low-frequency motion of the vessel (Faltinsen and Sortland 1987). Hence, this effect should be considered when designing the control system. In addition to linear skin friction, there will be a high-frequency contribution due to a turbulent boundary layer. This is usually referred to as a quadratic or nonlinear skin friction.

Wave Drift Damping: Wave drift damping can be interpreted as added resistance for surface vessels advancing in waves. This type of damping is derived from 2nd-order wave theory. Wave drift damping is the most important damping contribution to surge for higher sea states. This is due to the fact that the wave drift forces are proportional to

the square of the significant wave height. Wave drift damping in sway and yaw is small relative to eddy making damping (vortex shedding). A rule of thumb is that 2nd-order wave drift forces are less than 1% of the 1st-order wave forces when the significant wave height is equal to 1 m and 10% when the significant wave height is equal to 10 m.

Damping Due to Vortex Shedding: *D'Alambert's paradox* states that no hydrodynamic forces act on a solid moving completely submerged with constant velocity in a non-viscous fluid. In a viscous fluid, frictional forces are present such that the system is not conservative with respect to energy. The viscous damping force due to vortex shedding can be modeled as:

$$f(U) = -\frac{1}{2}\rho\, C_D(\mathrm{R}_n)\, A|U|U \tag{3.106}$$

where U is the speed of the vessel, A is the projected cross-sectional area under water, $C_D(\mathrm{R}_n)$ is the drag-coefficient based on the representative area, and ρ is the water density. This expression is recognized as one of the terms in *Morison's equation* (see Faltinsen 1990). The drag coefficient $C_D(\mathrm{R}_n)$ is a function of the *Reynolds number* (see Figure 3.3):

$$\mathrm{R}_n = \frac{UD}{\nu} \tag{3.107}$$

where D is the characteristic length of the body and ν is the kinematic viscosity coefficient ($\nu = 1.56 \cdot 10^{-6}$ for salt water at 5^o C with salinity 3.5%). Quadratic drag in 6 DOF is conveniently expressed as:

$$\mathbf{D}_n(\boldsymbol{\nu})\boldsymbol{\nu} = \begin{bmatrix} |\boldsymbol{\nu}|^\top \mathbf{D}_{n1}\boldsymbol{\nu} \\ |\boldsymbol{\nu}|^\top \mathbf{D}_{n2}\boldsymbol{\nu} \\ |\boldsymbol{\nu}|^\top \mathbf{D}_{n3}\boldsymbol{\nu} \\ |\boldsymbol{\nu}|^\top \mathbf{D}_{n4}\boldsymbol{\nu} \\ |\boldsymbol{\nu}|^\top \mathbf{D}_{n5}\boldsymbol{\nu} \\ |\boldsymbol{\nu}|^\top \mathbf{D}_{n6}\boldsymbol{\nu} \end{bmatrix} \tag{3.108}$$

where $|\boldsymbol{\nu}| = [|u|, |v|, |w|, |p|, |q|, |r|]^\top$ and $\mathbf{D}_{ni}$ $(i = 1, \ldots, 6)$ are 6×6 matrices depending on ρ, C_D and A.

The different damping terms contribute to both linear and quadratic damping. However, it is in general difficult to separate these effects. In many cases, it is convenient to write total hydrodynamic damping as:

$$\mathbf{D}(\boldsymbol{\nu}) = \mathbf{D} + \mathbf{D}_n(\boldsymbol{\nu}) \tag{3.109}$$

where $\mathbf{D}$ is the *linear damping matrix* and $\mathbf{D}_n(\boldsymbol{\nu})$ is the *nonlinear damping matrix*. Hydrodynamic damping satisfies the following property:

Property 3.5 (Hydrodynamic Damping Matrix $\mathbf{D}(\boldsymbol{\nu})$)
For a rigid-body moving through an ideal fluid the hydrodynamic damping matrix will be real, non-symmetric and strictly positive:

$$\mathbf{D}(\boldsymbol{\nu}) > 0 \quad \forall \quad \boldsymbol{\nu} \in \mathbb{R}^6.$$

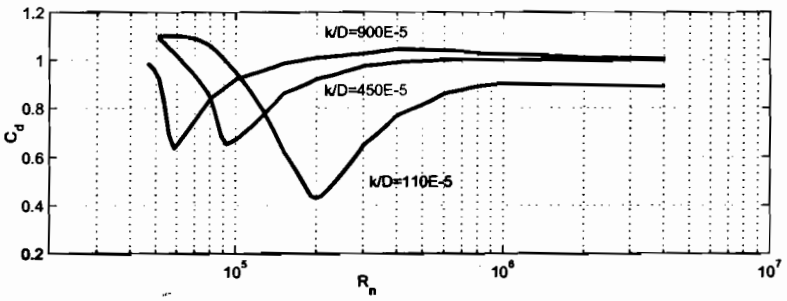


Figure 3.3: The drag coefficient C_d as a function of the Reynold's number Re and surface roughness curves k/D for a rough cylinder in a steady incident flow. k is the average surface height and D is the cylinder diameter (Faltinsen 1990).

Example 3.3 (Damping Model for Dynamic Positioning of Ships)
For low speed ships with xz-symmetry the surge mode can be decoupled from the steering modes (sway and yaw). Hence, the linearized damping forces and moments (neglecting heave, roll, and pitch) can be written:

$$\mathbf{D} = -\begin{bmatrix} X_u & 0 & 0 \\ 0 & Y_v & Y_r \\ 0 & N_v & N_r \end{bmatrix} \tag{3.110}$$

For low speed applications it can also be assumed that $N_v = Y_r$ such that $\mathbf{D} = \mathbf{D}^\top$.

In practice it is difficult to determine the nonlinear and off-diagonal terms. It is therefore a good idea to use different damping models depending on the regime of the control system. This is shown in Figure 3.4 where the significance of linear and quadratic damping is illustrated.

Example 3.4 (Ad Hoc Damping Model for High Speed Maneuvers)
For maneuvers at high speed nonlinear damping $\mathbf{D}_n(\nu)$ must be included such that:

$$\mathbf{D}(\nu) = \mathbf{D} + \mathbf{D}_n(\nu) \tag{3.111}$$

where $\mathbf{D}$ is given by (3.110). A first attempt could be to use a damping model motivated by Morrison's equation, *that is:*

$$\mathbf{D}_n(\nu) = -\text{diag}\{X_{|u|u}\,|u|\,, Y_{|v|v}\,|v|\,, N_{|r|r}\,|r|\} \tag{3.112}$$

where the quadratic terms $X_{|u|u}, Y_{|v|v}$, and $N_{|r|r}$ are given by (3.106).

It should be noted that it is important to include both linear and quadratic damping, since only quadratic damping in the model will cause an oscillatory behavior at low speed. The main reason is that linear damping is needed for exponential convergence to zero. For marine vessels, linear damping will always be present due to linear skin friction (Faltinsen and Sortland 1987).

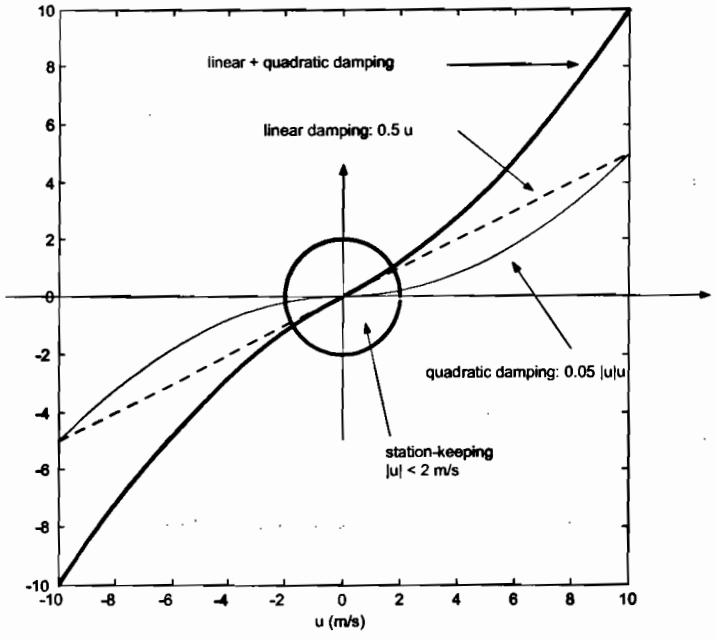


Figure 3.4: Linear and quadratic damping and their regimes (low and high speed).

Example 3.5 (Nonlinear Damping Model for High Speed Maneuvers)
In Blanke (1981) a more detailed model including nonlinear coupling terms is proposed. This is a simplification of Norrbin's nonlinear model (Norrbin 1970). Motivated by this a more general expression (assuming that surge is decoupled) is:

$$\mathrm{D}_n(\boldsymbol{\nu}) = \begin{bmatrix} -X_{|u|u}\,|u| & 0 & 0 \\ 0 & -Y_{|v|v}\,|v| - Y_{|r|v}\,|r| & -Y_{|v|r}\,|v| - Y_{|r|r}\,|r| \\ 0 & -N_{|v|v}\,|v| - N_{|r|v}\,|r| & -N_{|v|r}\,|v| - N_{|r|r}\,|r| \end{bmatrix} \tag{3.113}$$

For large ships $|r|\,r$ and $|r|\,v$ are small. This suggests that (Blanke 1981):

$$\mathrm{D}_n(\boldsymbol{\nu}) = \begin{bmatrix} -X_{|u|u}\,|u| & 0 & 0 \\ 0 & -Y_{|v|v}\,|v| & -Y_{|v|r}\,|v| \\ 0 & -N_{|v|v}\,|v| & -N_{|v|r}\,|v| \end{bmatrix} \tag{3.114}$$

Example 3.6 (Damping Model for Low-Speed Underwater Vehicles)
In general, the damping of an underwater vehicle moving in 6 DOF at high speed will be highly nonlinear and coupled. Nevertheless, one rough approximation could be to assume that the vehicle is performing a non-coupled motion. This suggests a diagonal structure of $\mathrm{D}(\boldsymbol{\nu})$ with only linear and quadratic damping terms on the diagonal:

$$\begin{aligned} \mathbf{D}(\boldsymbol{\nu}) \;=\; & -\mathrm{diag}\{X_u, Y_v, Z_w, K_p, M_q, N_r\} \\ & -\mathrm{diag}\{X_{|u|u}|u|, Y_{|vv|}|v|, Z_{|w|w}|w|, K_{|p|p}|p|, M_{|q|q}|q|, N_{|r|r}|r|\} \end{aligned} \tag{3.115}$$

As for ships quadratic damping can be neglected during station-keeping but not in high speed maneuvering situations.

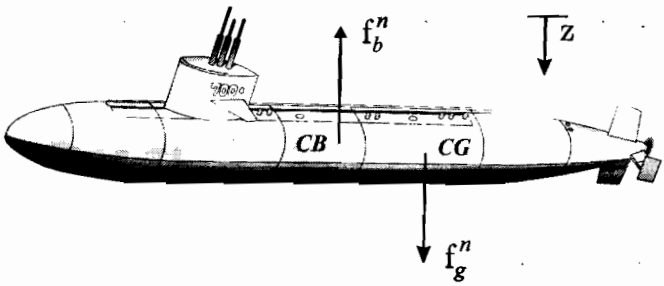


Figure 3.5: Gravitational and buoyancy forces acting on the center of gravity (CG) and center of buoyancy (CB) of a submarine.

3.2.3 Restoring Forces and Moments

Besides the mass and damping forces, underwater vehicles and floating vessels will also be affected by gravity and buoyancy forces. In hydrodynamic terminology, the gravitational and buoyancy forces are called *restoring forces*, and they are equivalent to the spring forces in a *mass-damper-spring* system. In the derivation of the restoring forces and moments:

- underwater vehicles
- surface vessels (ships, semi-submersibles, and high-speed craft)

will be treated separately.

UNDERWATER VEHICLES (SUBMERGED BODIES)

As shown in Figure 3.5 the gravitational force $\mathbf{f}_g^b$ will act through the center of gravity (*CG*) defined by $\mathbf{r}_g^b = [x_g, y_g, z_g]^\top$. Similarly, the buoyancy force $\mathbf{f}_b^b$ will act through the center of buoyancy (*CB*) defined by $\mathbf{r}_b^b = [x_b, y_b, z_b]^\top$.

Let m be the mass of the vessel including water in free floating space, ∇ the volume of fluid displaced by the vehicle, g the acceleration of gravity (positive downwards), and ρ the water density. According to the SNAME (1950) notation, the submerged weight of the body and buoyancy force are defined as:

$$W = mg, \qquad B = \rho g \nabla \tag{3.116}$$

Hence:

$$\mathbf{f}_g^n = \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} \quad \text{and} \quad \mathbf{f}_b^n = -\begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix} \tag{3.117}$$

Notice that the z-axis is taken to be positive downwards such that gravity is positive and buoyancy is negative. By applying the results from Section 2.2.1, the weight and buoyancy force can be transformed to the body-fixed coordinate system by:

$$\mathbf{f}_g^b = \mathbf{R}_b^n(\Theta)^{-1}\mathbf{f}_g^n, \qquad \mathbf{f}_b^b = \mathbf{R}_b^n(\Theta)^{-1}\mathbf{f}_b^n \tag{3.118}$$

where $\mathbf{R}_b^n(\Theta)$ is the Euler angle coordinate transformation matrix defined in Section 2.2.1. According to (3.82), the sign of the restoring forces and moments $\mathbf{f}_i^b$ and $\mathbf{m}_i^b = \mathbf{r}_i^b \times \mathbf{f}_i^b$, $i \in \{g, b\}$, must be changed when moving these terms to the left-hand side of (3.82), that is, the vector $\mathbf{g}(\boldsymbol{\eta})$. Consequently, the restoring force and moment vector in the body-fixed coordinate system is:

$$\begin{aligned} \mathbf{g}(\boldsymbol{\eta}) &= -\begin{bmatrix} \mathbf{f}_g^b + \mathbf{f}_b^b \\ \mathbf{r}_g^b \times \mathbf{f}_g^b + \mathbf{r}_b^b \times \mathbf{f}_b^b \end{bmatrix} \\ &= -\begin{bmatrix} \mathbf{R}_b^n(\Theta)^{-1}(\mathbf{f}_g^b + \mathbf{f}_b^b) \\ \mathbf{r}_g^b \times \mathbf{R}_b^n(\Theta)^{-1}\mathbf{f}_g^b + \mathbf{r}_b^b \times \mathbf{R}_b^n(\Theta)^{-1}\mathbf{f}_b^n \end{bmatrix} \end{aligned} \tag{3.119}$$

Expanding this expression yields:

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} (W-B)\sin\theta \\ -(W-B)\cos\theta\sin\phi \\ -(W-B)\cos\theta\cos\phi \\ -(y_gW - y_bB)\cos\theta\cos\phi + (z_gW - z_bB)\cos\theta\sin\phi \\ (z_gW - z_bB)\sin\theta + (x_gW - x_bB)\cos\theta\cos\phi \\ -(x_gW - x_bB)\cos\theta\sin\phi - (y_gW - y_bB)\sin\theta \end{bmatrix} \tag{3.120}$$

Matlab: Computation of $\mathbf{g}(\boldsymbol{\eta})$
The restoring forces can be computed by using the GNC toolbox commands:

```
r_g = [0, 0, 0]          % location of CG with respect to O
r_b = [0, 0, -10]        % location of CB with respect to O
m   = 1000               % mass
g   = 9.81               % acceleration of gravity
W   = m*g;               % weight
B   = W;                 % buoyancy

% pitch and roll angles
theta = 10*(180/pi); phi = 30*(pi/180)

% g-vector
g = gvect(W,B,theta,phi,r_g,r_b)
```

The numerical result is:

$$\mathbf{g} = 10^4 \cdot [0, 0, 0, 1.8324, 9.0997, 0]^\top$$

Equation (3.120) is the Euler angle representation of the hydrostatic forces and moments. An alternative representation can be found by applying *unit quaternions*. Then $\mathbf{R}_b^n(\mathbf{q})$ replaces $\mathbf{R}_b^n(\Theta)$ in (3.118); see Section 2.2.2.

A neutrally buoyant underwater vehicle will satisfy:

$$W = B \tag{3.121}$$

It is convenient to design underwater vehicles with $B > W$ (positive buoyancy) such that the vehicle will surface automatically in the case of an emergency situation, for instance power failure. In this case, the magnitude of B should only be slightly larger than W. If the vehicle is designed such that $B \gg W$, too much control energy is needed to keep the vehicle submerged. Hence, a trade-off between positive buoyancy and controllability must be made.

Example 3.7 (Neutrally Buoyant Underwater Vehicles)
Let the distance between the center of gravity CG *and the center of buoyancy* CB *be defined by the vector:*

$$\mathbf{BG} = [\overline{BG}_x,\ \overline{BG}_y,\ \overline{BG}_z]^\top = [x_g - x_b,\ y_g - y_b,\ z_g - z_b]^\top \tag{3.122}$$

For neutrally buoyant vehicles $W = B$, and (3.120) therefore simplifies to:

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\overline{BG}_y W\ \cos\theta\cos\phi + \overline{BG}_z W\ \cos\theta\sin\phi \\ \overline{BG}_z W\ \sin\theta + \overline{BG}_x W\ \cos\theta\cos\phi \\ -\overline{BG}_x W\ \cos\theta\sin\phi - \overline{BG}_y W\ \sin\theta \end{bmatrix} \tag{3.123}$$

An even simpler representation is obtained for vehicles where the CG and CB are located vertically on the z-axis, that is $x_b = x_g$ and $y_g = y_b$. This yields:

$$\mathbf{g}(\boldsymbol{\eta}) = [\ 0,\ \ 0,\ \ 0,\ \ \overline{BG}_z W\cos\theta\sin\phi,\ \ \overline{BG}_z W\sin\theta,\ \ 0\ ]^\top \tag{3.124}$$

SURFACE VESSELS (SHIPS AND SEMI-SUBMERSIBLES)

Formula (3.120) should only be used for completely submerged vehicles. Static stability considerations due to restoring forces are usually referred to as *metacentric stability* in the hydrostatic literature. A metacentric stable vessel will resist inclinations away from its steady-state or equilibrium points in heave, roll, and pitch.

For surface vessels, the restoring forces will depend on the vessel's metacentric height, the location of the *CG* and the *CB* as well as the shape and size of the water plane. Let A_{wp} denote the water plane area and:

$$\begin{aligned} \overline{GM}_T &= \text{transverse metacentric height (m)} \\ \overline{GM}_L &= \text{longitudinal metacentric height (m)} \end{aligned} \tag{3.125}$$

The metacentric height $\overline{GM}_i$, where $i \in \{T, L\}$,is the distance between the metacenter M_i and center of gravity G as shown in Figures 3.6 and 3.7.

Definition 3.2 (Metacenter)
The theoretical point at which an imaginary vertical line through the centre of buoyancy (CB) intersects another imaginary vertical line through a new centre of buoyancy created when the body is displaced, or tilted, in the water.

Gravity and Buoyancy

For a floating vessel at rest, buoyancy and weight are in balance:

$$mg = \rho g \nabla \tag{3.126}$$

Let z denote the displacement in heave and let $z = 0$ denote the equilibrium position corresponding to the nominal displaced water volume ∇. Hence, the hydrostatic force in heave will be the difference between the gravitational and the buoyancy forces:

$$\begin{aligned} Z &= mg - \rho g \left(\nabla + \delta\nabla(z)\right) \\ &= -\rho g \delta\nabla(z) \end{aligned} \tag{3.127}$$

where the change in displaced water $\delta\nabla(z)$ is due to variations in heave position z. This can be written:

$$\delta\nabla(z) = \int_0^z A_{wp}(\zeta) d\zeta \tag{3.128}$$

where $A_{wp}(\zeta)$ is the water plane area of the vessel as a function of the heave position. For conventional rigs and ships, however, it can be assumed that $A_{wp}(\zeta) \approx A_{wp}(0)$ is constant for small perturbations in z. Hence, the restoring force Z will be linear in z, that is:

$$Z \approx \underbrace{-\rho g A_{wp}(0)}_{Z_z} z \tag{3.129}$$

Recall that if a floating vessel is forced downwards by an external force such that $z > 0$, the buoyancy force becomes larger than the constant gravitational force since the submerged volume ∇ increases by $\delta\nabla$ to $\nabla + \delta\nabla$. This is physically equivalent to a spring with stiffness $Z_z = -\rho g A_{wp}(0)$ and position z. The restoring force decomposed in the b-frame, $\delta\mathbf{f}_r^b$, can therefore be written:

$$\begin{aligned} \delta\mathbf{f}_r^b &= \mathbf{R}_b^n(\boldsymbol{\Theta})^{-1} \delta\mathbf{f}_r^n \\ &= \mathbf{R}_b^n(\boldsymbol{\Theta})^{-1} \begin{bmatrix} 0 \\ 0 \\ -\rho g \int_0^z A_{wp}(\zeta) d\zeta \end{bmatrix} \\ &= -\rho g \begin{bmatrix} -\sin\theta \\ \cos\theta \sin\phi \\ \cos\theta \cos\phi \end{bmatrix} \int_0^z A_{wp}(\zeta) d\zeta \end{aligned} \tag{3.130}$$

From Figure 3.6 it is seen that the moment arms in roll and pitch can be related to the moment arms $GM_T \sin\phi$ and $GM_L \sin\theta$ in roll and pitch, and a z-direction force pair with magnitude $W = B = \rho g \nabla$. Hence:

$$\mathbf{r}_r^b = \begin{bmatrix} -\overline{GM}_L \sin\theta \\ \overline{GM}_T \sin\phi \\ 0 \end{bmatrix} \tag{3.131}$$

$$\mathbf{f}_r^b = \mathbf{R}_b^n(\boldsymbol{\Theta})^{-1} \begin{bmatrix} 0 \\ 0 \\ -\rho g \nabla \end{bmatrix} = -\rho g \nabla \begin{bmatrix} -\sin\theta \\ \cos\theta \sin\phi \\ \cos\theta \cos\phi \end{bmatrix} \tag{3.132}$$

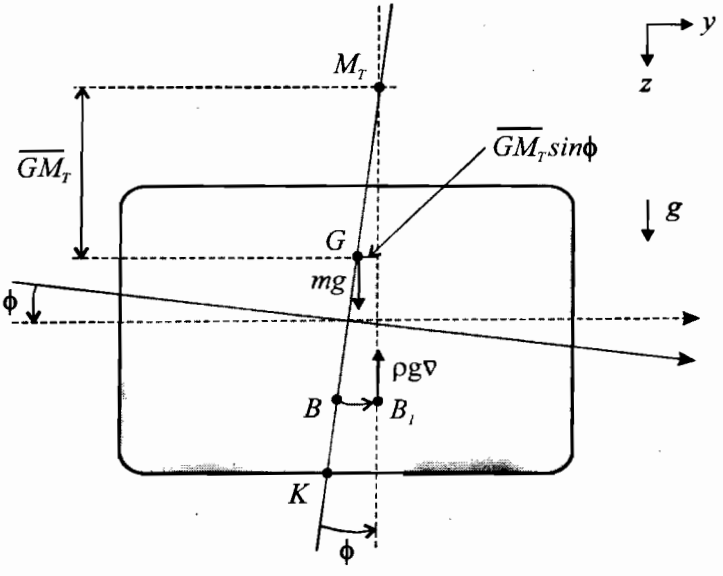


Figure 3.6: Transverse metacentric stability. Notice that $mg = \rho g \nabla$. A similar figure can be drawn to illustrate lateral metacentric stability by simply replacing M_T and ϕ with M_L and θ.

Neglecting the moment contribution due to $\delta\mathbf{f}_r^b$ (only considering $\mathbf{f}_r^b$) implies that the restoring moment becomes:

$$\begin{aligned}\mathbf{m}_r^b &= \mathbf{r}_r^b \times \mathbf{f}_r^b \\ &= -\rho g \nabla \begin{bmatrix} \overline{GM}_T \sin\phi\cos\theta\cos\phi \\ \overline{GM}_L \sin\theta\cos\theta\cos\phi \\ (-\overline{GM}_L\cos\theta + \overline{GM}_T)\sin\phi\sin\theta \end{bmatrix}\end{aligned} \tag{3.133}$$

The assumption that $\mathbf{r}_r^b \times \delta\mathbf{f}_r^b = \mathbf{0}$ (no moments due to heave) is a good assumption since this term is small compared to $\mathbf{r}_r^b \times \mathbf{f}_r^b$. The restoring forces and moments are finally written:

$$\mathbf{g}(\boldsymbol{\eta}) = -\begin{bmatrix} \delta\mathbf{f}_r^b \\ \mathbf{m}_r^b \end{bmatrix} \tag{3.134}$$

or on component form:

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} -\rho g \int_0^z A_{wp}(\zeta)d\zeta \ \sin\theta \\ \rho g \int_0^z A_{wp}(\zeta)d\zeta \ \cos\theta\sin\phi \\ \rho g \int_0^z A_{wp}(\zeta)d\zeta \ \cos\theta\cos\phi \\ \rho g \nabla \overline{GM}_T \sin\phi\cos\theta\cos\phi \\ \rho g \nabla \overline{GM}_L \sin\theta\cos\theta\cos\phi \\ \rho g \nabla(-\overline{GM}_L\cos\theta + \overline{GM}_T)\sin\phi\sin\theta \end{bmatrix} \tag{3.135}$$

Linear (Small Angle) Theory for Boxed Shaped Vessels

For surface vessels it is convenient to use a linear approximation:

$$\mathbf{g}(\boldsymbol{\eta}) \approx \mathbf{G}\boldsymbol{\eta} \tag{3.136}$$

which can be derived by assuming that ϕ, θ, and z are small. Assuming that:

$$\int_0^z A_{wp}(\zeta)d\zeta \approx A_{wp}(0)z$$

and

$$\begin{aligned} \sin\theta &\approx \theta, \qquad \cos\theta \approx 1 \\ \sin\phi &\approx \phi, \qquad \cos\phi \approx 1 \end{aligned}$$

implies that:

$$\mathbf{g}(\boldsymbol{\eta}) \approx \begin{bmatrix} -\rho g A_{wp}(0)\, z\theta \\ \rho g A_{wp}(0)\, z\phi \\ \rho g A_{wp}(0)\, z \\ \rho g \nabla \overline{GM}_T\, \phi \\ \rho g \nabla \overline{GM}_L\, \theta \\ \rho g \nabla(-\overline{GM}_L + \overline{GM}_T)\, \phi\theta \end{bmatrix} \approx \begin{bmatrix} 0 \\ 0 \\ \rho g A_{wp}(0) z \\ \rho g \nabla \overline{GM}_T \phi \\ \rho g \nabla \overline{GM}_L \theta \\ 0 \end{bmatrix} \tag{3.137}$$

Hence

$$\mathbf{G} = \mathrm{diag}\{0, 0, \rho g A_{wp}(0), \rho g \nabla \overline{GM}_T, \rho g \nabla \overline{GM}_L, 0\} \tag{3.138}$$

which can be used in a linearized model:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}\boldsymbol{\nu} + \mathbf{G}\boldsymbol{\eta} = \boldsymbol{\tau} + \mathbf{g}_0 + \mathbf{w} \tag{3.139}$$

This model is based on the assumption of yz-symmetry. In the asymmetrical case $\mathbf{G}$ takes the form:

$$\mathbf{G} = \mathbf{G}^\top = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_z & 0 & -Z_\theta & 0 \\ 0 & 0 & 0 & -K_\phi & 0 & 0 \\ 0 & 0 & -M_z & 0 & -M_\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} > 0 \tag{3.140}$$

where

$$-Z_z = \rho g A_{wp}(0) \tag{3.141}$$

$$-Z_\theta = \rho g \int\int_{A_{wp}} x dA \tag{3.142}$$

$$-M_z = -Z_\theta \tag{3.143}$$

$$-K_\phi = \rho g \nabla(z_g - z_g) + \rho g \int\int_{A_{wp}} y^2 dA = \rho g \nabla \overline{GM}_T \tag{3.144}$$

$$-M_\theta = \rho g \nabla(z_g - z_b) + \rho g \int\int_{A_{wp}} x^2 dA = \rho g \nabla \overline{GM}_L \tag{3.145}$$

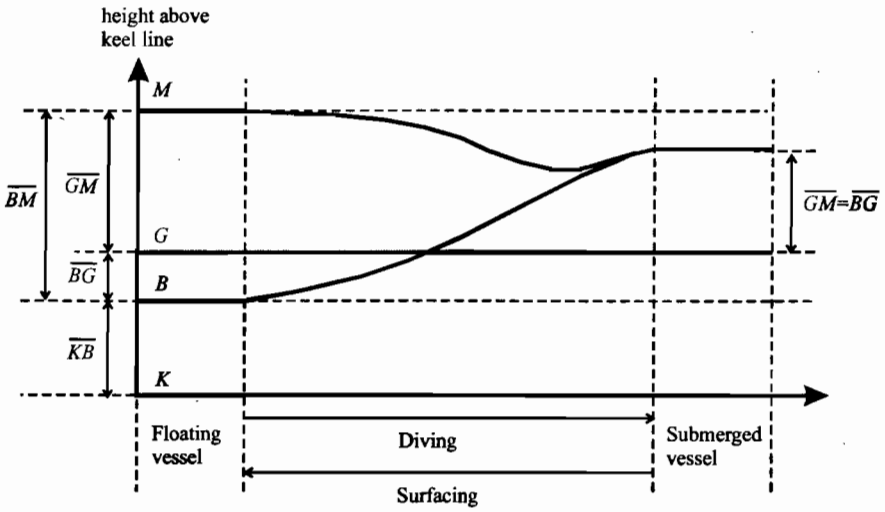


Figure 3.7: Metacenter M, center of gravity G and center of buoyancy B for a submerged and a floating vessel. K is the keel line.

Computation of Metacenter Height for Surface Vessels

The metacenter height can be computed by using basic hydrostatics:

$$\overline{GM}_T = \overline{BM}_T - \overline{BG}, \qquad \overline{GM}_L = \overline{BM}_L - \overline{BG} \tag{3.146}$$

This relationship is seen directly from Figure 3.7 where M_T and M_L denote the transverse and longitudinal metacenters (intersections between the vertical lines through B and B_1 when ϕ and θ approaches zero). The symbol K is used to denote the keel line. For small inclinations (ϕ and θ are small) the transverse and longitudinal radius of curvature can be approximated by:

$$\overline{BM}_T = \frac{I_T}{\nabla}, \qquad \overline{BM}_L = \frac{I_L}{\nabla} \tag{3.147}$$

where the moments of area about the water planes are defined as:

$$I_L = \int\int_{A_{wp}} x^2 dA, \qquad I_T = \int\int_{A_{wp}} y^2 dA \tag{3.148}$$

For conventional ships an upper bound on these integrals can be found by considering a rectangular water plane area $A_{wp} = BL$ where B and L are the beam and length of the hull, respectively. This yields:

$$I_L < \frac{1}{12} L^3 B, \qquad I_T < \frac{1}{12} B^3 L \tag{3.149}$$

These formulas can be used as a first estimate when programming a vessel simulator.

Definition 3.3 (Metacenter Stability)
A floating vessel is said to be transverse metacentrically stable if:

$$\overline{GM}_T \geq \overline{GM}_{T,\min} > 0 \tag{3.150}$$

and longitudinal metacentrically stable if:

$$\overline{GM}_L \geq \overline{GM}_{L,\min} > 0 \tag{3.151}$$

The longitudinal stability requirement (3.151) is easy to satisfy for ships since the pitching motion is quite limited. The lateral requirement, however, is an important design criterion used to predescribe sufficient stability in roll to avoid that the vessel does not roll around. For instance, for large ferries carrying passengers and cars, the lateral stability requirement can be as high as $\overline{GM}_{T,\min} = 0.8$ (m) to guarantee a proper stability margin in roll.

A trade-off between stability and comfort should be made since a large stability margin will result in large restoring forces which can be quite uncomfortable for passengers.

3.2.4 Ballast Systems

In addition to the restoring forces $\mathbf{g}(\boldsymbol{\eta})$ described in Section 3.2.3 the vessel can be pretrimmed by pumping water between the ballast tanks of the vessel. This implies that the vessel can be trimmed in *heave, pitch* and *roll* where restoring forces are present.

Let the *desired pretrimming values* be:

$$z = z_d, \quad \phi = \phi_d, \quad \text{and} \quad \theta = \theta_d$$

The equilibrium states corresponding to these values are found by considering the steady-state solution of:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{3.152}$$

which under assumption of zero acceleration/speed $\dot{\boldsymbol{\nu}} = \boldsymbol{\nu} = \mathbf{0}$, and $\boldsymbol{\tau} = \mathbf{0}$ reduces to:

$$\mathbf{g}(\boldsymbol{\eta}_d) = \mathbf{g}_o + \mathbf{w} \tag{3.153}$$

where $\boldsymbol{\eta}_d = [-, -, -, z_d, \phi_d, \theta_d, -]^\top$; only three states are used for pretrimming.

The ballast vector $\mathbf{g}_o$ is computed by using hydrostatic analyses. Consider a marine vessel with n ballast tanks of volumes $V_i \leq V_{i,\max}$ $(i = 1, \ldots, n)$. For each ballast tank the water volume is defined as:

$$V_i(h_i) = \int_o^{h_i} A_i(h)dh \approx A_i h_i, \qquad (A_i(h) = \text{constant}) \tag{3.154}$$

where $A_i(h)$ is the area of the ballast tank at height h. Hence, the volume of the water column in each ballast tank can be computed by measuring the water height h_i. Next, assume that the ballast tanks are located at:

$$\mathbf{r}_i^b = [x_i, y_i, z_i]^\top, \quad (i = 1, \ldots, n) \tag{3.155}$$

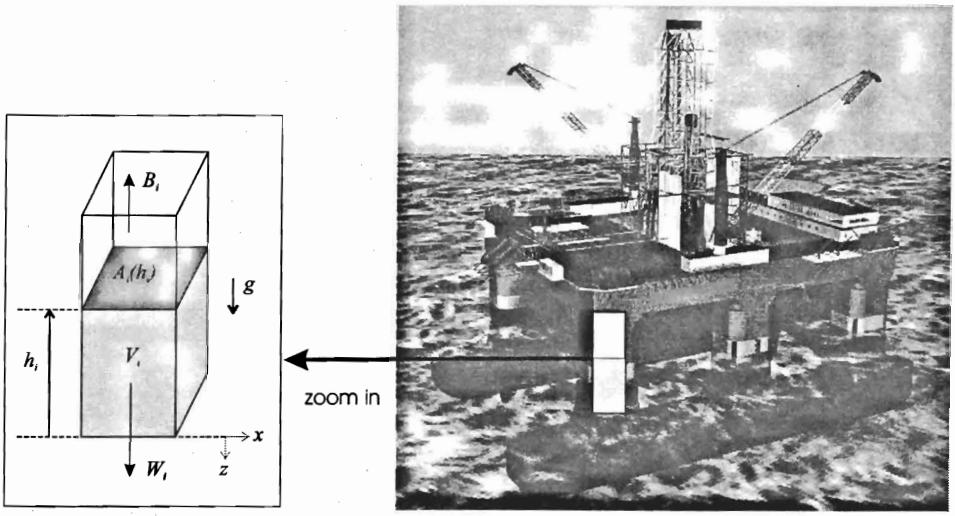


Figure 3.8: Semi-submersible ballast tanks.

where $\mathbf{r}_i^b$ is the vector from the coordinate origin O to the geometric center of tank i. Alternatively, the vector $\mathbf{r}_{ci}^b$ from CG to the volume center can be used since:

$$\mathbf{r}_i^b = \mathbf{r}_g^b + \mathbf{r}_{ci}^b \tag{3.156}$$

The difference between the gravitational W_i and buoyancy B_i forces in heave can now be computed for each ballast tank as (see Figure 3.8):

$$\begin{aligned} Z_{\text{ballast}} &= W_i - B_i \\ &= \rho g \sum_{i=1}^{n} V_i - \rho g \sum_{i=1}^{n} (V_{i,\max} - V_i) \\ &= \rho g \sum_{i=1}^{n} (2V_i - V_{i,\max}) \end{aligned} \tag{3.157}$$

The moments due to the ballast heave force are then found from:

$$\begin{aligned} \mathbf{m} &= \mathbf{r} \times \mathbf{f} \\ &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ Z_{\text{ballast}} \end{bmatrix} = \begin{bmatrix} yZ_{\text{ballast}} \\ -xZ_{\text{ballast}} \\ 0 \end{bmatrix} \end{aligned} \tag{3.158}$$

implying that the roll and pitch moments due to ballast are:

$$K_{\text{ballast}} = \rho g \sum_{i=1}^{n} y_i (2V_i - V_{i,\max}) \tag{3.159}$$

$$M_{\text{ballast}} = -\rho g \sum_{i=1}^{n} x_i (2V_i - V_{i,\max}) \tag{3.160}$$

Finally, this gives:

$$\mathbf{g}_o = \begin{bmatrix} 0 \\ 0 \\ Z_{\text{ballast}} \\ K_{\text{ballast}} \\ M_{\text{ballast}} \\ 0 \end{bmatrix} = \rho g \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^{n} (2V_i - V_{i,\max}) \\ \sum_{i=1}^{n} y_i(2V_i - V_{i,\max}) \\ -\sum_{i=1}^{n} x_i(2V_i - V_{i,\max}) \\ 0 \end{bmatrix} \tag{3.161}$$

Conditions for Manual Pretrimming

Distribution of water between the ballast tanks can be done manually by pumping water until the desired water levels h_i in each tank is reached or automatically by using feedback control. For manual operation, the steady-state relationships between water levels h_i and the desired pretrimming values z_d, ϕ_d, and θ_d are needed. Trimming is usually done under the assumptions that ϕ_d and θ_d are small such that according to (3.140):

$$\mathbf{g}(\boldsymbol{\eta}_d) \approx \mathbf{G}\boldsymbol{\eta}_d \tag{3.162}$$

Since, we are only concerned with the *heave, roll,* and *pitch* modes it is convenient to use the reduced order system:

$$\begin{aligned} \mathbf{G}^r &= \begin{bmatrix} -Z_z & 0 & -Z_\theta \\ 0 & -K_\phi & 0 \\ -M_z & 0 & -M_\theta \end{bmatrix} \\ \mathbf{g}_o^r &= \rho g \begin{bmatrix} \sum_{i=1}^{n} (2V_i - V_{i,\max}) \\ \sum_{i=1}^{n} y_i(2V_i - V_{i,\max}) \\ -\sum_{i=1}^{n} x_i(2V_i - V_{i,\max}) \end{bmatrix} \\ \boldsymbol{\eta}_d^r &= [z_d, \phi_d, \theta_d]^\top \\ \mathbf{w}^r &= [w_3, w_4, w_5]^\top \end{aligned}$$

From (3.153) and (3.140) it is seen that:

$$\begin{aligned} \mathbf{G}^r \boldsymbol{\eta}_d^r &= \mathbf{g}_o^r + \mathbf{w}^r \qquad (3.163) \\ &\Updownarrow \\ \begin{bmatrix} -Z_z & 0 & -Z_\theta \\ 0 & -K_\phi & 0 \\ -M_z & 0 & -M_\theta \end{bmatrix} \begin{bmatrix} z_d \\ \phi_d \\ \theta_d \end{bmatrix} &= \begin{bmatrix} \rho g \sum_{i=1}^{n} (2V_i - V_{i,\max}) + w_3 \\ \rho g \sum_{i=1}^{n} y_i(2V_i - V_{i,\max}) + w_4 \\ -\rho g \sum_{i=1}^{n} x_i(2V_i - V_{i,\max}) + w_5 \end{bmatrix} \end{aligned}$$

For simplicity, assume that the disturbances in *heave, roll,* and *pitch* have means of zero. Consequently, (3.163) with $\mathbf{w}^r = [w_3, w_4, w_5]^\top = \mathbf{0}$ can be written:

$$\begin{aligned} \mathbf{H}\boldsymbol{\upsilon} &= \mathbf{y} \qquad (3.164) \\ &\Updownarrow \end{aligned}$$

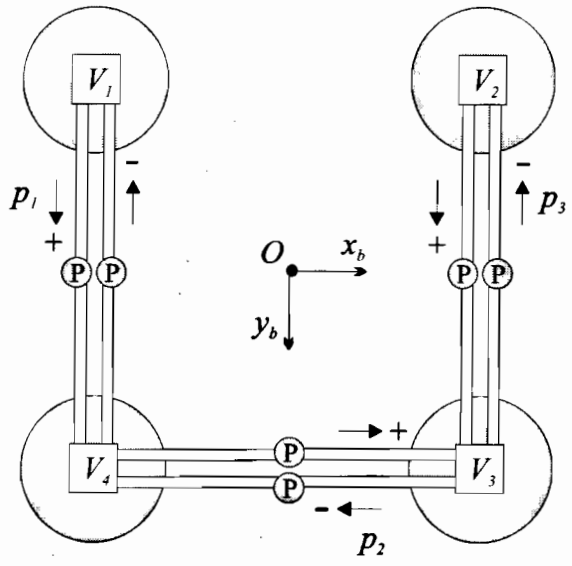


Figure 3.9: Semi-submersible with four ballast tanks. V_i (m^3) is the water volume in leg $i = 1, \ldots, 4$ and p_j (m^3/s) is the volume flow for water pump $j = 1, \ldots, 3$.

$$2\rho g \begin{bmatrix} 1 & \cdots & 1 & 1 \\ y_1 & \cdots & y_{n-1} & y_n \\ -x_1 & \cdots & -x_{n-1} & -x_n \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix} = \begin{bmatrix} -Z_z z_d - Z_\theta \theta_d + \rho g \sum_{i=1}^{n} V_{i,\max} \\ -K_\phi \phi_d + \rho g \sum_{i=1}^{n} y_i V_{i,\max} \\ -M_z z_d - M_\theta \theta_d - \rho g \sum_{i=1}^{n} x_i V_{i,\max} \end{bmatrix} \tag{3.165}$$

where $\boldsymbol{v}$ is a vector of tank volumes:

$$\boldsymbol{v} = [V_1, V_2, ..., V_n]^\top \tag{3.166}$$

The solution of (3.164) is found by using the pseudo-inverse:

$$\begin{aligned} \boldsymbol{v} &= \mathbf{H}^\dagger \mathbf{y} \\ &= \mathbf{H}^\top (\mathbf{H}\mathbf{H}^\top)^{-1} \mathbf{y} \end{aligned} \tag{3.167}$$

where it is assumed that $n \geq 3$ and that $\mathbf{H}\mathbf{H}^\top$ has full rank. Finally, the desired water heights can be computed from:

$$V_i(h_i) = \int_o^{h_i} A_i(h) dh \quad \overset{A_i(h)=A_i}{\Longrightarrow} \quad h_i = \frac{V_i}{A_i}, \quad (i = 1, \ldots, n) \tag{3.168}$$

Example 3.8 (Semi-Submersible Ballast Control)
Consider the semi-submersible shown in Figure 3.9 with 4 ballast tanks located at $\mathbf{r}_1^b = [-x, -y], \mathbf{r}_2^b = [x, -y], \mathbf{r}_3^b = [x, y]$ *and* $\mathbf{r}_4^b = [-x, y]$. *In addition, yz-symmetry implies that* $Z_\theta = M_z = 0$ *while the diagonal elements in* $\mathbf{G}^r$ *are nonzero. It is also assumed that*

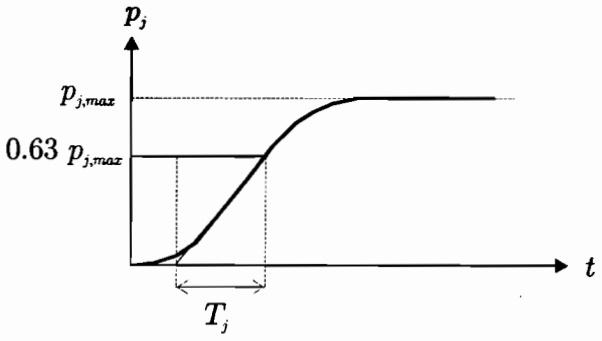


Figure 3.10: The time constant T_j for pump j is found by commanding a step $p_{d_j} = p_{j,max}$ as shown in the plot.

the ballast tanks are of same size such that $V_{1,\max} = V_{2,\max} = V_{3,\max} = V_{4,\max} = V_{\max}$. *Hence:*

$$\mathbf{H} = 2\rho g \begin{bmatrix} 1 & 1 & 1 & 1 \\ -y & -y & y & y \\ x & -x & -x & x \end{bmatrix}$$

$$\mathbf{y} = \begin{bmatrix} -Z_z z_d + 4\rho g V_{\max} \\ -K_\phi \phi_d \\ -M_\theta \theta_d \end{bmatrix} = \begin{bmatrix} \rho g (A_{wp}(0) z_d + 4V_{\max}) \\ \rho g \nabla \overline{GM}_T \phi_d \\ \rho g \nabla \overline{GM}_L \theta_d \end{bmatrix}$$

The pseudo-inverse of $\mathbf{H}$ *is:*

$$\mathbf{H}^\dagger = \mathbf{H}^\top (\mathbf{H}\mathbf{H}^\top)^{-1} = \frac{1}{8\rho g} \begin{bmatrix} 1 & -\frac{1}{y} & \frac{1}{x} \\ 1 & -\frac{1}{y} & -\frac{1}{x} \\ 1 & \frac{1}{y} & -\frac{1}{x} \\ 1 & \frac{1}{y} & \frac{1}{x} \end{bmatrix}$$

which finally gives the water volumes V_i *corresponding to the desired values* z_d, ϕ_d, *and* θ_d:

$$\boldsymbol{v} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \frac{1}{8\rho g} \begin{bmatrix} 1 & -\frac{1}{y} & \frac{1}{x} \\ 1 & -\frac{1}{y} & -\frac{1}{x} \\ 1 & \frac{1}{y} & -\frac{1}{x} \\ 1 & \frac{1}{y} & \frac{1}{x} \end{bmatrix} \begin{bmatrix} \rho g (A_{wp}(0) z_d + 4V_{\max}) \\ \rho g \nabla \overline{GM}_T \phi_d \\ \rho g \nabla \overline{GM}_L \theta_d \end{bmatrix}$$

Automatic Pretrimming using Feedback from z,ϕ, and θ

In the manual pretrimming case it was assumed that $\mathbf{w}^r = \mathbf{0}$. This assumption can be removed by using feedback from z, ϕ, and θ. The closed-loop dynamics of a PID-controlled water pump can be described by a 1st-order model with amplitude saturation:

$$T_j \dot{p}_j + p_j = \text{sat}(p_{d_j}) \tag{3.169}$$

where T_j (s) is a positive time constant, p_j (m^3/s) is the volumetric flow rate produced by Pump $j = 1, \ldots, m$, and p_{d_j} is the pump set-point. As shown in Figure 3.9, one separate water pump can be used to pump water in each direction. This implies that the water pump capacity is different for positive and negative flow directions. Moreover:

$$\text{sat}(p_{d_j}) = \begin{cases} p^+_{j,\max} & p_j > p^+_{j,\max} \\ p_{d_j} & p^-_{j,\max} \leq p_{d_j} \leq p^+_{j,\max} \\ p^-_{j,\max} & p_{d_j} < p^-_{j,\max} \end{cases} \tag{3.170}$$

The pump time constant T_j is found from a step response as shown in Figure 3.10.

The volume flow $\dot{V}_i$ to Tank i is given by linear combinations of flows corresponding to the pumps/pipelines supporting Tank i. For the semi-submersible shown in Figure 3.9, we obtain:

$$\begin{aligned} \dot{V}_1 &= -p_1 \\ \dot{V}_2 &= -p_3 \\ \dot{V}_3 &= p_2 + p_3 \\ \dot{V}_4 &= p_1 - p_2 \end{aligned}$$

More generally, the water flow model can be written:

$$\mathbf{T}\dot{\mathbf{p}} + \mathbf{p} = \text{sat}(\mathbf{p}_d) \tag{3.171}$$

$$\dot{\boldsymbol{\upsilon}} = \mathbf{L}\mathbf{p} \tag{3.172}$$

where $\text{sat}(\mathbf{p}_d) = [\text{sat}(p_{d_1}), ..., \text{sat}(p_{d_m})]^\top$, $\mathbf{p} = [p_1, ..., p_m]^\top$, and $\boldsymbol{\upsilon} = [V_1, ..., V_n]^\top$ $(m \geq n)$. The mapping from the water volume vector $\boldsymbol{\upsilon}$ to $\boldsymbol{\eta}^r$ is (see Figure 3.11):

$$\mathbf{G}^r \boldsymbol{\eta}^r = \mathbf{g}^r_o(\boldsymbol{\upsilon}) + \mathbf{w}^r \tag{3.173}$$

Example 3.9 (Semi-Submersible Ballast Control, Continues)
Consider the semi-submersible in Example 3.8. The water flow model corresponding to Figure 3.9 becomes:

$$\boldsymbol{\upsilon} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}, \quad \mathbf{p} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 1 \\ 1 & -1 & 0 \end{bmatrix} \tag{3.174}$$

reflecting that there is three pumps and four water volumes connected through the configuration matrix $\mathbf{L}$.

A feedback control system for automatic trimming is shown in Figure 3.11. The ballast controllers can be chosen to be of PID-type, for instance:

$$\mathbf{p}_d = \mathbf{H}_{\text{pid}}(s)\mathbf{G}^r(\boldsymbol{\eta}^r_d - \boldsymbol{\eta}^r) \tag{3.175}$$

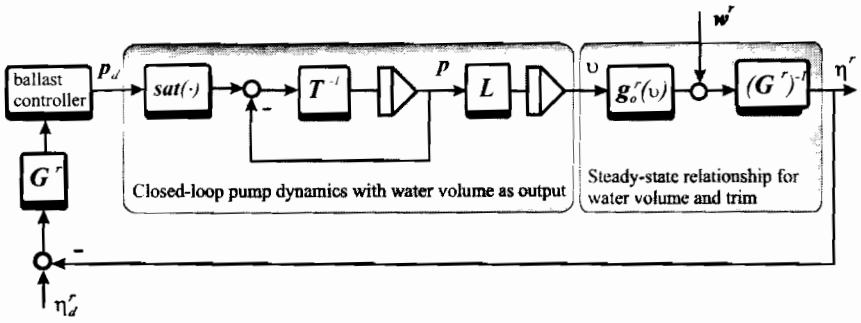


Figure 3.11: Ballast control system using feedback from z, ϕ and θ.

where $\boldsymbol{\eta}_d^r = [z_d, \phi_d, \theta_d]^\top$ and:

$$\mathbf{H}_{\text{pid}}(s) = \text{diag}\{h_{1,\text{pid}}(s), h_{2,\text{pid}}(s), ..., h_{m,\text{pid}}(s)\} \tag{3.176}$$

is a diagonal transfer matrix containing m PID controllers. Integral action in the controllers is needed to compensate for non-zero environmental disturbances $\mathbf{w}^r$.

The SeaLaunch Trim and Heel Correction System

An example of a highly sophisticated pretrimming system is the *SeaLaunch* trim and heel correction system; see Section 1.1.5. This system is designed such that the platform maintains constant roll and pitch angles during changes in weight. The most critical operation is when the rocket is transported from the garage on one side of the platform to the launch pad. During this operation the water pumps operate at their maximum capacity to counteract the shift in weight. An automatic feedback system controls the pumps to maintain the correct water level in each of the legs during transportation of the rocket; see Figure 3.12. This is necessary to keep the platform levelled.

3.3 6 DOF Equations of Motion

3.3.1 Nonlinear Equations of Motion

In this section different representations and properties of the marine vessel equations of motion are discussed. It will also be shown how various body-symmetries can be used to simplify the equations of motion.

6 DOF Body-Fixed Vector Representation

In Section 3.1 it was shown that the nonlinear equations of motion in the body-fixed frame can be written as:

$$\mathbf{M}\,\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\,\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\,\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{3.177}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{3.178}$$

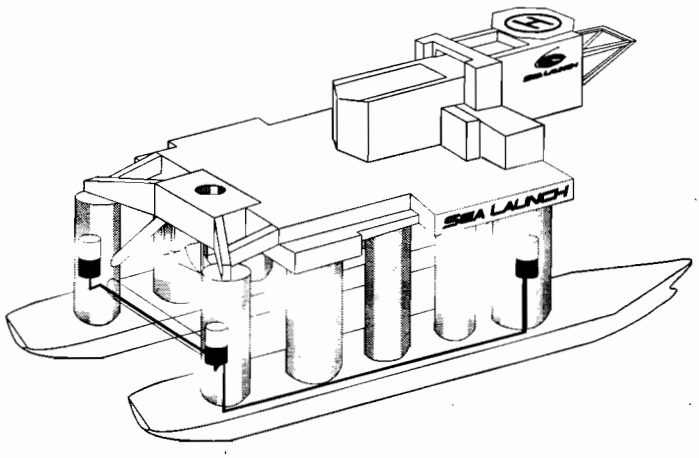


Figure 3.12: Ballast tanks for the SeaLaunch trim and heel correction system. Courtesy to SeaLaunch.

where

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_{RB} + \mathbf{M}_A && (3.179)\\ \mathbf{C}(\boldsymbol{\nu}) &= \mathbf{C}_{RB}(\boldsymbol{\nu}) + \mathbf{C}_A(\boldsymbol{\nu}) && (3.180)\\ \mathbf{D}(\boldsymbol{\nu}) &= \mathbf{D}_P(\boldsymbol{\nu}) + \mathbf{D}_S(\boldsymbol{\nu}) + \mathbf{D}_W(\boldsymbol{\nu}) + \mathbf{D}_M(\boldsymbol{\nu}) && (3.181) \end{aligned}$$

6 DOF NED Vector Representation

The NED representation is obtained by applying the following kinematic transformations (assuming that $\mathbf{J}^{-1}(\boldsymbol{\eta})$ exists–i.e., $\theta \neq \pm\pi/2$):

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \mathbf{J}(\boldsymbol{\eta})\,\boldsymbol{\nu} && \Longleftrightarrow && \boldsymbol{\nu} &= \mathbf{J}^{-1}(\boldsymbol{\eta})\,\dot{\boldsymbol{\eta}}\\ \dot{\boldsymbol{\eta}}' &= \mathbf{J}(\boldsymbol{\eta})\,\dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}(\boldsymbol{\eta})\,\boldsymbol{\nu} && \Longleftrightarrow && \dot{\boldsymbol{\nu}} &= \mathbf{J}^{-1}(\boldsymbol{\eta})\left[\ddot{\boldsymbol{\eta}} - \dot{\mathbf{J}}(\boldsymbol{\eta})\mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}\right] \end{aligned}$$

Next $\boldsymbol{\nu}$ and $\dot{\boldsymbol{\nu}}$ in (3.177) can be replaced by $\dot{\boldsymbol{\eta}}$ and $\ddot{\boldsymbol{\eta}}$ by using the transformations:

$$\begin{aligned} \mathbf{M}^*(\boldsymbol{\eta}) &= \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{M}\,\mathbf{J}^{-1}(\boldsymbol{\eta})\\ \mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta}) &= \mathbf{J}^{-\top}(\boldsymbol{\eta})\left[\mathbf{C}(\boldsymbol{\nu}) - \mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\mathbf{J}}(\boldsymbol{\eta})\right]\mathbf{J}^{-1}(\boldsymbol{\eta})\\ \mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta}) &= \mathbf{J}^{-\top}(\boldsymbol{\eta})\,\mathbf{D}(\boldsymbol{\nu})\,\mathbf{J}^{-1}(\boldsymbol{\eta})\\ \mathbf{g}^*(\boldsymbol{\eta}) &= \mathbf{J}^{-\top}(\boldsymbol{\eta})\,\mathbf{g}(\boldsymbol{\eta}) \end{aligned} \qquad (3.182)$$

This results in the NED vector representation:

$$\mathbf{M}^*(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} + \mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \mathbf{g}^*(\boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})(\boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w}) \qquad (3.183)$$

Properties of the Body-Fixed Vector Representation

The following properties are observed for the body-fixed vector representation:

Property 3.6 (System Inertia Matrix M)
For a rigid body the system inertia matrix is strictly positive if and only if $\mathbf{M}_A > 0$, *that is:*

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A > 0$$

If the body is at rest (or at most is moving at low speed) under the assumption of an ideal fluid (see Property 3.3), the system inertia matrix is always positive definite, that is:

$$\mathbf{M} = \mathbf{M}^\top > 0$$

where $\mathbf{M}$ *is defined as:*

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & -X_{\dot{v}} & -X_{\dot{w}} & -X_{\dot{p}} & mz_g - X_{\dot{q}} & -my_g - X_{\dot{r}} \\ -X_{\dot{v}} & m - Y_{\dot{v}} & -Y_{\dot{w}} & -mz_g - Y_{\dot{p}} & -Y_{\dot{q}} & mx_g - Y_{\dot{r}} \\ -X_{\dot{w}} & -Y_{\dot{w}} & m - Z_{\dot{w}} & my_g - Z_{\dot{p}} & -mx_g - Z_{\dot{q}} & -Z_{\dot{r}} \\ -X_{\dot{p}} & -mz_g - Y_{\dot{p}} & my_g - Z_{\dot{p}} & I_x - K_{\dot{p}} & -I_{xy} - K_{\dot{q}} & -I_{zx} - K_{\dot{r}} \\ mz_g - X_{\dot{q}} & -Y_{\dot{q}} & -mx_g - Z_{\dot{q}} & -I_{xy} - K_{\dot{q}} & I_y - M_{\dot{q}} & -I_{yz} - M_{\dot{r}} \\ -my_g - X_{\dot{r}} & mx_g - Y_{\dot{r}} & -Z_{\dot{r}} & -I_{zx} - K_{\dot{r}} & -I_{yz} - M_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix}$$

Property 3.7 (Coriolis and Centripetal Matrix C) *For a rigid body moving through an ideal fluid the Coriolis and centripetal matrix* $\mathbf{C}(\boldsymbol{\nu})$ *can always be parameterized such that it is skew-symmetric, that is:*

$$\mathbf{C}(\boldsymbol{\nu}) = -\mathbf{C}^\top(\boldsymbol{\nu}), \qquad \forall \boldsymbol{\nu} \in \mathbb{R}^6$$

Proof. $\mathbf{C}(\boldsymbol{\nu})$ is skew-symmetric under the assumptions that the matrices $\mathbf{C}_{RB}(\boldsymbol{\nu})$ and $\mathbf{C}_A(\boldsymbol{\nu})$ are skew-symmetric. ■

The Assumption of Zero Wave Frequency

For a marine vessel, $\mathbf{M}$, $\mathbf{C}(\boldsymbol{\nu})$ and $\mathbf{D}(\boldsymbol{\nu})$ will also depend on the wave frequency ω and thus the speed of the vessel through the frequency of encounter. This relationship has not been established for a general vessel in 6 DOF. However, in control design asymptotic values can be used, since only the low-frequency components are of interest. Hence, it will be assumed that:

Assumption 3.1 (Zero Wave Frequency)
The low-frequency control model of a marine vessel can be approximated by:

$$\mathbf{M} = \lim_{\omega \to 0} \mathbf{M}(\omega); \qquad \mathbf{C}(\boldsymbol{\nu}) = \lim_{\omega \to 0} \mathbf{C}(\boldsymbol{\nu},\omega); \qquad \mathbf{D}(\boldsymbol{\nu}) = \lim_{\omega \to 0} \mathbf{D}(\boldsymbol{\nu},\omega) \tag{3.184}$$

This assumption implies that $\dot{\mathbf{M}} = \mathbf{0}$ (zero frequency assumption) such that the following holds:

$$\mathbf{s}^\top [\dot{\mathbf{M}} - 2\mathbf{C}(\boldsymbol{\nu})]\mathbf{s} \overset{\dot{\mathbf{M}}=\mathbf{0}}{=} -2\mathbf{s}^\top \mathbf{C}(\boldsymbol{\nu})\mathbf{s} \overset{\mathbf{C}=-\mathbf{C}^\top}{=} 0, \qquad \forall \mathbf{s} \in \mathbb{R}^6 \tag{3.185}$$

This relationship has its analogy in the dynamic description of robot manipulators, where the $\mathbf{C}$ matrix can be calculated from the so-called Christoffel symbols. Christoffel symbols, however, are not defined for vessels in terms of body-fixed velocities.

Properties of the NED Vector Representation

As in the body-fixed vector representation it is straightforward to show that:

(1) $\mathbf{M}^*(\boldsymbol{\eta}) = \mathbf{M}^*(\boldsymbol{\eta})^\top > 0 \;\; \forall \;\; \boldsymbol{\eta} \in \mathbb{R}^6$

(2) $\mathbf{s}^\top \left[\dot{\mathbf{M}}^*(\boldsymbol{\eta}) - 2\mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta})\right] \mathbf{s} = 0 \;\; \forall \;\; \mathbf{s} \in \mathbb{R}^6, \; \boldsymbol{\nu} \in \mathbb{R}^6, \; \boldsymbol{\eta} \in \mathbb{R}^6$

(3) $\mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta}) \; > \; 0 \;\; \forall \;\; \boldsymbol{\nu} \in \mathbb{R}^6, \; \boldsymbol{\eta} \in \mathbb{R}^6$

if $\mathbf{M} = \mathbf{M}^\top > 0$ and $\dot{\mathbf{M}} = \mathbf{0}$. The proofs are left as an exercise. It should be noted that $\mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta})$ will not be skew-symmetrical although $\mathbf{C}(\boldsymbol{\nu})$ is skew-symmetrical.

3.3.2 Linearized Equations of Motion

The following assumption will be applied when deriving the linearized vessel model:

Assumption 3.2 (Small Roll and Pitch Angles)
The roll and pitch angles:

$$\phi, \theta \text{ are small} \tag{3.186}$$

These are good assumptions for vessels where the pitch and roll motions are limited–i.e., highly metacentric stable vessels.

Vessel Parallel Coordinate System

When deriving the linearized equations of motion it is convenient to introduce a *vessel parallel* coordinate system. This is a coordinate system fixed to the vessel with axes parallel to an Earth-fixed reference frame usually the *NED* reference frame.

Assumption 3.2 implies that:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \overset{\phi=\theta=0}{\approx} \mathbf{P}(\psi)\boldsymbol{\nu} \tag{3.187}$$

where

$$\mathbf{P}(\psi) = \begin{bmatrix} \mathbf{R}(\psi) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{3.188}$$

and $\mathbf{R}(\psi) = \mathbf{R}_{z,\psi}$ is the rotation matrix in yaw.

Definition 3.4 (Vessel Parallel Coordinate System)
The vessel parallel coordinate system is defined as:

$$\boldsymbol{\eta}_p = \mathbf{P}^\top(\psi)\boldsymbol{\eta} \tag{3.1}$$

where $\boldsymbol{\eta}_p$ is the NED position/attitude decomposed in body coordinates and $\mathbf{P}(\psi)$ is given (3.188). Notice that $\mathbf{P}^\top(\psi)\mathbf{P}(\psi) = \mathbf{I}_{6\times 6}$.

Low Speed Applications (Station-Keeping)

It is convenient to express the kinematic equations of motion in vessel parallel (VP) coordinates when using linear theory. From Definition 3.4 it is seen that:

$$\begin{aligned}\dot{\boldsymbol{\eta}}_p &= \dot{\mathbf{P}}^\top(\psi)\boldsymbol{\eta} + \mathbf{P}^\top(\psi)\dot{\boldsymbol{\eta}} \\ &= \dot{\mathbf{P}}^\top(\psi)\mathbf{P}(\psi)\boldsymbol{\eta}_p + \mathbf{P}^\top(\psi)\mathbf{P}(\psi)\boldsymbol{\nu} \\ &= r\mathbf{S}\boldsymbol{\eta}_p + \boldsymbol{\nu}\end{aligned} \tag{3.190}$$

where $r = \dot{\psi}$ and:

$$\mathbf{S} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.191}$$

For low speed applications $r \approx 0$. Hence, (3.190) reduces to 6 pure integrators:

$$\dot{\boldsymbol{\eta}}_p \approx \boldsymbol{\nu} \tag{3.192}$$

This model is attractive since it is linear in $\boldsymbol{\nu}$. In fact, this is the main idea for using VP coordinates in ship and rig control design.

The gravitational and buoyancy forces can also be expressed in terms of VP coordinates. For small angles ϕ and θ it is seen that (see Section 3.2.3):

$$\mathbf{g}(\boldsymbol{\eta}) \overset{\phi=\theta=0}{\approx} \mathbf{P}^\top(\psi)\mathbf{G}\boldsymbol{\eta} = \underbrace{\mathbf{P}^\top(\psi)\mathbf{G}\mathbf{P}(\psi)}_{\mathbf{G}}\boldsymbol{\eta}_p = \mathbf{G}\boldsymbol{\eta}_p \tag{3.193}$$

Notice that this formula confirms that the restoring forces of a leveled floating vessel ($\phi = \theta = 0$) is independent of the yaw angle ψ. This can be illustrated by considering the following two examples:

Neutrally Buoyant Submersible: For a neutrally buoyant submersible ($W = B$) with $x_g = x_b$ and $y_g = y_b$, Assumption 3.2 implies that, see (3.124):

$$\mathbf{G} = \text{diag}\{0, 0, 0, 0, (z_g - z_b)W, (z_g - z_b)W, 0\} \tag{3.194}$$

which is independent of the yaw angle ψ. Hence, (3.194) satisfies (3.193).

Surface Vessel: For a surface vessel $\mathbf{G}$ is defined by (3.140). Thanks to the special structure of $\mathbf{G}$, that is:

$$\mathbf{G} = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 3} & \begin{matrix} 0 \\ 0 \end{matrix} \\ \mathbf{0}_{3\times 2} & \mathbf{G}^r & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & 0 \end{matrix} & \begin{matrix} 0 & 0 & 0 \end{matrix} & 0 \end{bmatrix}, \quad \mathbf{G}^r = \begin{bmatrix} -Z_z & 0 & -Z_\theta \\ 0 & -K_\phi & 0 \\ -M_z & 0 & -M_\theta \end{bmatrix} \tag{3.195}$$

It is again seen that $\mathbf{P}^\top(\psi)\mathbf{G}\mathbf{P}(\psi) \equiv \mathbf{G}$.

Assumption 3.2 for low-speed applications $\boldsymbol{\nu} \approx \mathbf{0}$ implies that the nonlinear Coriolis, centripetal, damping, restoring, and buoyancy forces and moments can be linearized about $\boldsymbol{\nu} = \mathbf{0}$ and $\phi = \theta = 0$. Since $\mathbf{C}(\mathbf{0}) = \mathbf{0}$ and $\mathbf{D}_n(\mathbf{0}) = \mathbf{0}$ it makes sense to approximate:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \underbrace{\mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}}_{\mathbf{0}} + \underbrace{[\mathbf{D} + \mathbf{D}_n(\boldsymbol{\nu})]\boldsymbol{\nu}}_{\mathbf{D}\boldsymbol{\nu}} + \underbrace{\mathbf{g}(\boldsymbol{\eta})}_{\mathbf{G}\boldsymbol{\eta}_p} = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{3.196}$$

which can be written:

$$\dot{\boldsymbol{\eta}}_p = \boldsymbol{\nu} \tag{3.197}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} + \mathbf{G}\boldsymbol{\eta}_p = \boldsymbol{\tau} + \mathbf{w} \tag{3.198}$$

This is a *linear time invariant* (LTI) state-space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{3.199}$$

where $\mathbf{x} = [\boldsymbol{\eta}_p^\top, \boldsymbol{\nu}^\top]^\top$, $\mathbf{u} = \boldsymbol{\tau}$, and:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{G} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1} \end{bmatrix} \tag{3.200}$$

This model is the foundation for DP and PM control systems design; see Chapter 11. Notice that the *NED* positions are computed from $\boldsymbol{\eta}_p$ by using:

$$\boldsymbol{\eta} = \mathbf{P}(\psi)\boldsymbol{\eta}_p \tag{3.201}$$

Hence, the control system can be based on feedback from the states $(\boldsymbol{\eta}_p, \boldsymbol{\nu})$ while $\boldsymbol{\eta}$ is presented to the human operator using (3.201).

Vessels in Transit (Cruise Condition)

For vessels in transit the cruise speed is assumed to satisfy:

$$u = u_o \tag{3.202}$$

This suggests that

$$\mathbf{N}(u_o) = \frac{\partial}{\partial \boldsymbol{\nu}} \{\mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}\}|_{\boldsymbol{\nu}=\boldsymbol{\nu}_o} \tag{3.203}$$

where $\boldsymbol{\nu}_o = [u_o, 0, 0, 0, 0, 0]^\top$. Defining $\Delta\boldsymbol{\nu} = \boldsymbol{\nu} - \boldsymbol{\nu}_o$, yields:

$$\dot{\boldsymbol{\eta}}_p = \Delta\boldsymbol{\nu} + \boldsymbol{\nu}_o \tag{3.204}$$

$$\mathbf{M}\Delta\dot{\boldsymbol{\nu}} + \mathbf{N}(u_o)\Delta\boldsymbol{\nu} + \mathbf{G}\boldsymbol{\eta}_p = \boldsymbol{\tau} + \mathbf{w} \tag{3.205}$$

This corresponds to a *linear parameter varying* (LPV) model:

$$\dot{\mathbf{x}} = \mathbf{A}(u_o)\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} + \mathbf{F}\boldsymbol{\nu}_o \tag{3.206}$$

where $\mathbf{x} = [\boldsymbol{\eta}_p^\top, \Delta\boldsymbol{\nu}^\top]^\top$, $\mathbf{u} = \boldsymbol{\tau}$, and:

$$\mathbf{A}(u_o) = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{G} & -\mathbf{M}^{-1}\mathbf{N}(u_o) \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1} \end{bmatrix} \tag{3.207}$$

$$\mathbf{E} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \tag{3.208}$$

Notice that $\mathbf{A}(u_o)$ depends on the forward speed u_o. This suggests that the control law for transit (maneuvering) should be gain scheduled with respect to forward speed u_o.

Notice that station-keeping resulted in a LTI model, while maneuvering implies that a LPV model must be used.

3.4 Model Transformations using Matlab

When deriving the nonlinear equations of motion it is convenient to specify inertia, damping, gravitational, and buoyancy forces in different reference frames exploiting the structural properties of these expressions. For instance, the translational and rotational parts of the system inertia matrix is decoupled if the coordinate system is located in the CG. These properties are derived by defining a transformation matrix that transforms the generalized velocities, accelerations, and forces between two points both located in the b-frame. The transformation matrix used for this purpose is called the *system transformation matrix.*

3.4.1 System Transformation Matrix

In order to exploit the b-frame model properties and to derive expressions for the different matrices in different b-frame origins, the *system transformation matrix* is derived from (3.12) according to:

$$\begin{aligned} \mathbf{v}_p^b &= \mathbf{v}_o^b + \boldsymbol{\omega}_{nb}^b \times \mathbf{r}_p^b \\ &= \mathbf{v}_o^b - \mathbf{S}(\mathbf{r}_p^b)\boldsymbol{\omega}_{nb}^b \\ &= \mathbf{v}_o^b + \mathbf{S}^\top(\mathbf{r}_p^b)\boldsymbol{\omega}_{nb}^b \end{aligned} \tag{3.209}$$

where $\mathbf{r}_p^b = [x_p, y_p, z_p]^\top$ is a vector from the body-fixed coordinate origin O to an arbitrarily point P in the b-frame, and $\boldsymbol{\omega}_{nb}^b$ denotes the angular velocity vector of the two b-frames with respect to the n-frame; see Figure 3.13.

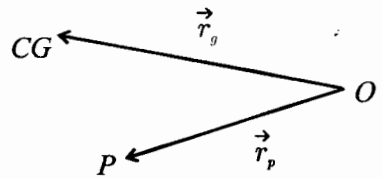


Figure 3.13: Definitions of vectors and coordinate systems.

Definition 3.5 (System Transformation Matrix)
The transformation matrix:

$$\mathbf{H}(\mathbf{r}_p^b) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{S}^\top(\mathbf{r}_p^b) \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix}, \qquad \mathbf{H}^{-1}(\mathbf{r}_p^b) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{S}(\mathbf{r}_p^b) \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{3.210}$$

transforms the linear and angular velocity vectors between the two points O and P *in the* b*-frame:*

$$\begin{bmatrix} \mathbf{v}_p^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix} = \mathbf{H}(\mathbf{r}_p^b) \begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix} \tag{3.211}$$

$$\Updownarrow$$

$$\boldsymbol{\nu}_p = \mathbf{H}(\mathbf{r}_p^b)\boldsymbol{\nu} \tag{3.212}$$

Similarly, the generalized force vector $\boldsymbol{\tau}$ can be transformed from O *to an arbitrarily point* P *by:*

$$\begin{bmatrix} \mathbf{f}_o^b \\ \mathbf{m}_o^b \end{bmatrix} = \begin{bmatrix} \mathbf{f}_p^b \\ \mathbf{r}_p^b \times \mathbf{f}_p^b + \mathbf{m}_p^b \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{S}(\mathbf{r}_p^b) & \mathbf{I}_{3\times3} \end{bmatrix} \begin{bmatrix} \mathbf{f}_p^b \\ \mathbf{m}_p^b \end{bmatrix} \tag{3.213}$$

$$\Updownarrow$$

$$\boldsymbol{\tau} = \mathbf{H}^\top(\mathbf{r}_p^b)\boldsymbol{\tau}_p \tag{3.214}$$

Matlab:
The system transformation is implemented in the GNC toolbox as:

```
function H = Hmtrx(r)
% H = HMTRX(r) % 6x6 system transformation matrix

S = Smtrx(r);
H = [eye(3)      S'
     zeros(3,3) eye(3) ];
```

Definition 3.5 implies that the nonlinear equations of motion can be represented at an arbitrary defined point P by using the transformation matrix $\mathbf{H}(\mathbf{r}_p^b)$. Consider the nonlinear equations

of motion with respect to the b-frame coordinate origin O:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{3.215}$$

This expression can be transformed to a point P in the b-frame by:

$$\begin{aligned}\underbrace{\mathbf{H}^{-\top}(\mathbf{r}_p^b)\mathbf{M}\mathbf{H}^{-1}(\mathbf{r}_p^b)}_{\mathbf{M}_p}\dot{\boldsymbol{\nu}}_p + \underbrace{\mathbf{H}^{-\top}(\mathbf{r}_p^b)\mathbf{C}(\boldsymbol{\nu})\mathbf{H}^{-1}(\mathbf{r}_p^b)}_{\mathbf{C}_p(\boldsymbol{\nu})}\boldsymbol{\nu}_p \\ + \underbrace{\mathbf{H}^{-\top}(\mathbf{r}_p^b)\mathbf{D}(\boldsymbol{\nu})\mathbf{H}^{-1}(\mathbf{r}_p^b)}_{\mathbf{D}_p(\boldsymbol{\nu})}\boldsymbol{\nu}_p + \underbrace{\mathbf{H}^{-\top}(\mathbf{r}_p^b)\mathbf{g}(\boldsymbol{\eta})}_{\mathbf{g}_p(\boldsymbol{\eta})} = \underbrace{\mathbf{H}^{-\top}(\mathbf{r}_p^b)\boldsymbol{\tau}}_{\boldsymbol{\tau}_p}\end{aligned} \tag{3.216}$$

From this expression the following useful formulas can be derived:

$$\mathbf{M} = \mathbf{H}^{\top}(\mathbf{r}_p^b)\mathbf{M}_p\mathbf{H}(\mathbf{r}_p^b) \tag{3.217}$$

$$\mathbf{C}(\boldsymbol{\nu}) = \mathbf{H}^{\top}(\mathbf{r}_p^b)\mathbf{C}_p(\boldsymbol{\nu})\mathbf{H}(\mathbf{r}_p^b) \tag{3.218}$$

$$\mathbf{D}(\boldsymbol{\nu}) = \mathbf{H}^{\top}(\mathbf{r}_p^b)\mathbf{D}_p(\boldsymbol{\nu})\mathbf{H}(\mathbf{r}_p^b) \tag{3.219}$$

$$\mathbf{g}(\boldsymbol{\eta}) = \mathbf{H}^{\top}(\mathbf{r}_p^b)\mathbf{g}(\boldsymbol{\eta}) \tag{3.220}$$

These expressions can be used to specify the inertia, damping, and restoring forces in different reference frames in order to exploit different physical properties.

3.4.2 Computation of the System Inertia Matrix

It is convenient to specify the rigid-body system inertia matrix (3.55) with respect to the CG such that:

$$\begin{aligned}\mathbf{M}_{RB}^{cg} &= \begin{bmatrix} m\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{I}_c \end{bmatrix} \\ &= \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_x^{cg} & -I_{xy}^{cg} & -I_{zx}^{cg} \\ 0 & 0 & 0 & -I_{xy}^{cg} & I_y^{cg} & -I_{yz}^{cg} \\ 0 & 0 & 0 & -I_{zx}^{cg} & -I_{yz}^{cg} & I_z^{cg} \end{bmatrix}\end{aligned} \tag{3.221}$$

The expression for $\mathbf{M}_{RB}^{cg}$ is uniquely defined by 7 parameters: $\{m, I_x^{cg}, I_y^{cg}, I_z^{cg}, -I_{xy}^{cg}, -I_{zx}^{cg}, -I_{yz}^{cg}\}$. It can be transformed to the b-frame coordinate origin O by specifying the vector $\mathbf{r}_p^b = \mathbf{r}_g^b = [x_g, y_g, z_g]^{\top}$ such that the points P and CG coincides. Accordingly, (3.217)

implies that:

$$\begin{aligned}
\mathbf{M}_{RB} &= \mathbf{H}^{\top}(\mathbf{r}_g^b)\mathbf{M}_{RB}^{cg}\mathbf{H}(\mathbf{r}_g^b) \\
&= \begin{bmatrix} m\mathbf{I}_{3\times 3} & -m\mathbf{S}(\mathbf{r}_g^b) \\ m\mathbf{S}(\mathbf{r}_g^b) & \underbrace{\mathbf{I}_c - m\mathbf{S}^2(\mathbf{r}_g^b)}_{\mathbf{I}_o} \end{bmatrix} \\
&= \begin{bmatrix} m & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_x & -I_{xy} & -I_{zx} \\ mz_g & 0 & -mx_g & -I_{xy} & I_y & -I_{yz} \\ -my_g & mx_g & 0 & -I_{zx} & -I_{yz} & I_z \end{bmatrix}
\end{aligned} \tag{3.222}$$

which is recognized as (3.55).

Matlab:
The 6×6 rigid body system inertia matrix $\mathbf{M}_{RB}$ about an arbitrarily point O can be computed by using the following Matlab™ commands:

```
r_g = [x_g y_g z_g]'   % location of the CG with respect to O
I_c = [  Ix  -Ixy -Ixz % 3x3 inertia matrix about CG
        -Ixy   Iy  -Iyz
        -Ixz -Iyz  Iz ]

MRB_CG = [  m*eye(3)   zeros(3,3)
          zeros(3,3)         I_c  ]

MRB = Hmtrx(r_g)'*MRB_CG*Hmtrx(r_g)
```

Added Mass System Inertia Matrix

For hydrodynamic added mass this is somewhat more complicated. Let us define the *center of added mass* (*CA*) as the point in the body where the total added mass forces act (resultant of the added mass forces in the x-, y-, and z-directions). The *CA* can be approximated by using potential theory from which the center of pressure will be a good approximation of the *CA*. The center of pressure is usually computed by using Wamit, which is a hydrodynamic computation program.

If the *CA* is used as foundation for the added mass system inertia matrix, the upper left submatrix becomes diagonal (zero off-diagonal elements), that is:

$$\mathbf{M}_A^{ca} := \begin{bmatrix} \mathbf{A}_{11} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{A}_{22} \end{bmatrix} \tag{3.223}$$

where

$$\begin{aligned}
\mathbf{A}_{11} &= -\text{diag}\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}\} \\
\mathbf{A}_{22} &= \text{full matrix depending on the added mass principal axes}
\end{aligned}$$

such that $\mathbf{M}_A^{ca}$ is defined in terms of 9 parameters. This structure is motivated by the rigid-body matrix $\mathbf{M}_{RB}^{cg}$.

In many cases $\mathbf{M}_A$ is known in the *CG* or *CB*. In these cases it is straightforward to transform $\mathbf{M}_A$ to the *b*-frame origin *O* by using the system transformation matrix and the vector $\mathbf{r}_g^b$. This principle can be illustrated by assuming that $\mathbf{M}_A$ is known in the *CG*. For instance, assume that $\mathbf{M}_A^{cg}$ can be described by 6 parameters according to:

$$\mathbf{M}_A^{cg} \approx -\begin{bmatrix} \text{diag}\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}\} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \text{diag}\{K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\} \end{bmatrix} \tag{3.224}$$

where for simplicity a diagonal structure of $\mathbf{M}_A^{cg}$ is used. This is often the best estimate you have unless you are using a hydrodynamic software program that computes a full $\mathbf{M}_A^{cg}$ or $\mathbf{M}_A^{ca}$ matrix. For slender bodies, the diagonal elements can be found by using strip theory or semi-empirical methods (Faltinsen 1990).

The system inertia matrix (3.224) is transformed to the *b*-frame origin *O* by choosing $\mathbf{r}_p^b = \mathbf{r}_g^b$ in (3.217). Consequently:

$$\begin{aligned} \mathbf{M}_A &= \mathbf{H}^\top(\mathbf{r}_g^b)\mathbf{M}_A^{cg}\mathbf{H}(\mathbf{r}_g^b) \\ &= \begin{bmatrix} -X_{\dot{u}} & 0 & 0 & 0 & -z_g X_{\dot{u}} & y_g X_{\dot{u}} \\ 0 & -Y_{\dot{v}} & 0 & z_g Y_{\dot{v}} & 0 & -x_g Y_{\dot{v}} \\ 0 & 0 & -Z_{\dot{w}} & -y_g Z_{\dot{w}} & x_g Z_{\dot{w}} & 0 \\ 0 & z_g Y_{\dot{v}} & -y_g Z_{\dot{w}} & -z_g^2 Y_{\dot{v}} - y_g^2 Z_{\dot{w}} - K_{\dot{p}} & x_g y_g Z_{\dot{w}} & x_g z_g Y_{\dot{v}} \\ -z_g X_{\dot{u}} & 0 & x_g Z_{\dot{w}} & x_g y_g Z_{\dot{w}} & -z_g^2 X_{\dot{u}} - x_g^2 Z_{\dot{w}} - M_{\dot{q}} & y_g z_g X_{\dot{u}} \\ y_g X_{\dot{u}} & -x_g Y_{\dot{v}} & 0 & x_g z_g Y_{\dot{v}} & y_g z_g X_{\dot{u}} & -y_g^2 X_{\dot{u}} - x_g^2 Y_{\dot{v}} - N_{\dot{r}} \end{bmatrix} \end{aligned} \tag{3.225}$$

Finally, the system inertia matrix with respect to *O* is computed as the sum:

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A \tag{3.226}$$

Matlab:
The 6×6 system inertia matrix about *O* is computed in GNC toolbox according to:

```
% added mass with respect to CG=CA
MA_CG = diag([-Xudot,-Yvdot,-Zwdot,-Kpdot,-Mqdot,-Nrdot])

% added mass with respect to O
MA = Hmtrx(r_g)'*MA_CG*Hmtrx(r_g)

% system inertia matrix
M = MRB + MA
```

Symmetry Considerations of the System Inertia Matrix

We have seen that the 6 DOF nonlinear equations of motion, in their most general representation, require that a large number of hydrodynamic derivatives is known. From a practical point of view this is an unsatisfactory situation. However, the number of unknown parameters can be drastically reduced by using body symmetry conditions.

The expression (3.224) was based on the assumption that $\mathbf{M}_A$ was diagonal in the *CG* while $\mathbf{M}_{RB}$ as defined by (3.222) is a full matrix. If the general expression for $\mathbf{M}_A$ is used (full matrix in *CG),* body symmetries will still simplify the system inertia matrix

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A$$

considerably. For instance, it is straightforward to verify the following cases (notice that $m_{ij} = m_{ji}$):

(i) xy-plane of symmetry (bottom/top symmetry).

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} & 0 & 0 & 0 & m_{16} \\ m_{21} & m_{22} & 0 & 0 & 0 & m_{26} \\ 0 & 0 & m_{33} & m_{34} & m_{35} & 0 \\ 0 & 0 & m_{43} & m_{44} & m_{45} & 0 \\ 0 & 0 & m_{53} & m_{54} & m_{55} & 0 \\ m_{61} & m_{62} & 0 & 0 & 0 & m_{66} \end{bmatrix}$$

(ii) xz-plane of symmetry (port/starboard symmetry).

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & m_{13} & 0 & m_{15} & 0 \\ 0 & m_{22} & 0 & m_{24} & 0 & m_{26} \\ m_{31} & 0 & m_{33} & 0 & m_{35} & 0 \\ 0 & m_{42} & 0 & m_{44} & 0 & m_{46} \\ m_{51} & 0 & m_{53} & 0 & m_{55} & 0 \\ 0 & m_{62} & 0 & m_{64} & 0 & m_{66} \end{bmatrix}$$

(iii) yz-plane of symmetry (fore/aft symmetry)

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & 0 & 0 & m_{15} & m_{16} \\ 0 & m_{22} & m_{23} & m_{24} & 0 & 0 \\ 0 & m_{32} & m_{33} & m_{34} & 0 & 0 \\ 0 & m_{42} & m_{43} & m_{44} & 0 & 0 \\ m_{51} & 0 & 0 & 0 & m_{55} & m_{56} \\ m_{61} & 0 & 0 & 0 & m_{65} & m_{66} \end{bmatrix}$$

(iv) xz- and yz-planes of symmetry (port/starboard and fore/aft symmetries).

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & 0 & 0 & m_{15} & 0 \\ 0 & m_{22} & 0 & m_{24} & 0 & 0 \\ 0 & 0 & m_{33} & 0 & 0 & 0 \\ 0 & m_{42} & 0 & m_{44} & 0 & 0 \\ m_{51} & 0 & 0 & 0 & m_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & m_{66} \end{bmatrix}$$

More generally, the resulting inertia matrix for a body with ij- and jk-planes of symmetry is formed by the intersection $\mathbf{M}_{ij \cap jk} = \mathbf{M}_{ij} \cap \mathbf{M}_{jk}$.

(v) xz-, yz- and xy-planes of symmetry (port/starboard, fore/aft and bottom/top symmetries).

$$\mathbf{M} = \text{diag}\{m_{11}, m_{22}, m_{33}, m_{44}, m_{55}, m_{66}\}$$

3.4.3 Computation of the Coriolis-Centrifugal Matrix

The Coriolis-centrifugal matrix can be computed directly by applying Theorem 3.2.

Matlab:
The GNC toolbox commands:

```
CA  = m2c(MA,nu)
CRB = m2c(MRB,nu)
```

imply that the resulting **M**- and **C**-matrices can be computed as:

```
M = MRB + MA
C = CRB + CA
```

Alternatively, the **C**-matrix can be computed as:

```
C = m2c(M,nu)
```

3.4.4 Computation of the Damping Matrix

The resulting hydrodynamic damping force will act at a point which can be defined as the "*center of dissipative forces*" (*CD*). Since force is equal to the product of pressure and area, this point will depend on the cross-sectional areas in the x-,y-, and z-directions, and the magnitude of the damping forces along the respective axes.

In the *CD* the damping matrix $\mathbf{D}(\boldsymbol{\nu})$ will have a diagonal structure similar to that of added mass with respect to the *CA*. Unfortunately, it is time consuming to compute the location of the *CD* and how it changes with speed, load condition, draft and trim. This can, however, be done by using computational fluid dynamics (CFD) for instance. A rough approximation is to use the formulas for quadratic drag in the x-,y- and z-directions to compute the *CD* since the viscous terms are the dominating dissipative forces.

If the *CD* is unknown, a first approximation could be to assume that the damping matrix is diagonal in the *CG*. Moreover, the damping matrix is written as the sum of the linear and nonlinear parts:

$$\mathbf{D}(\boldsymbol{\nu}) = \mathbf{D} + \mathbf{D}_n(\boldsymbol{\nu}) \tag{3.227}$$

where:

$$\mathbf{D}^{cg} \approx -\text{diag}\{X_u, Y_v, Z_w, K_p, M_q, N_r\} \tag{3.228}$$

and

$$\mathbf{D}_n^{cg}(\boldsymbol{\nu}) \approx -\text{diag}\{X_{|u|u}|u|, Y_{|vv|}|v|, Z_{|w|w}|w|, K_{|p|p}|p|, M_{|q|q}|q|, N_{|r|r}|r|\} \tag{3.229}$$

Transforming these expressions to O, yields:

$$\mathbf{D} = \mathbf{H}^{\top}(\mathbf{r}_g^b)\mathbf{D}^{cg}\mathbf{H}(\mathbf{r}_g^b)$$

$$= \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & 0 \\ 0 & 0 & -Z_w \\ 0 & z_g Y_v & -y_g Z_w \\ -z_g X_u & 0 & x_g Z_w \\ y_g X_u & -x_g Y_v & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -z_g X_u & y_g X_u \\ z_g Y_v & 0 & -x_g Y_v \\ -y_g Z_w & x_g Z_w & 0 \\ -z_g^2 Y_v - y_g^2 Z_w - K_p & x_g y_g Z_w & x_g z_g Y_v \\ x_g y_g Z_w & -z_g^2 X_u - x_g^2 Z_w - M_q & y_g z_g X_u \\ x_g z_g Y_v & y_g z_g X_u & -y_g^2 X_u - x_g^2 Y_v - N_r \end{bmatrix}$$

and

$$\mathbf{D}_n(\boldsymbol{\nu}) = \mathbf{H}^{\top}(\mathbf{r}_g^b)\mathbf{D}_n^{cg}(\boldsymbol{\nu})\mathbf{H}(\mathbf{r}_g^b)$$

$$= \begin{bmatrix} -X_{u|u|}\,|u| & 0 & 0 \\ 0 & -Y_{v|v|}\,|v| & 0 \\ 0 & 0 & -Z_{w|w|}\,|w| \\ 0 & z_g Y_{v|v|}\,|v| & -y_g Z_{w|w|}\,|w| \\ -z_g X_{u|u|}\,|u| & 0 & x_g Z_{w|w|}\,|w| \\ y_g X_{u|u|}\,|u| & -x Y_{v|v|}\,|v| & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -z_g X_{u|u|}\,|u| \\ z_g Y_{v|v|}\,|v| & 0 \\ -y_g Z_{w|w|}\,|w| & x_g Z_{w|w|}\,|w| \\ -z_g^2 Y_{v|v|}\,|v| - y_g^2 Z_{w|w|}\,|w| - K_{p|p|}\,|p| & x_g y_g Z_{w|w|}\,|w| \\ x_g y_g Z_{w|w|}\,|w| & -z_g^2 X_{u|u|}\,|u| - x_g^2 Z_{w|w|}\,|w| - M_{q|q|}\,|q| \\ x_g z_g Y_{v|v|}\,|v| & y_g z_g X_{u|u|}\,|u| \end{bmatrix}$$

$$\begin{bmatrix} y_g X_{u|u|}\,|u| \\ -x_g Y_{v|v|}\,|v| \\ 0 \\ x_g z_g Y_{v|v|}\,|v| \\ y_g z_g X_{u|u|}\,|u| \\ -y_g^2 X_{u|u|}\,|u| - x_g^2 Y_{v|v|}\,|v| - N_{r|r|}\,|r| \end{bmatrix}$$

Matlab:
The expression for the damping matrix is generated in GNC toolbox by typing:

```
%D_CG = damping matrix with respect to CG
D = Hmtrx(r_g)'*D_CG*Hmtrx(r_g)
```

Simplicity Considerations of the Damping Matrix

The expression for the linear part of the damping matrix follows the same symmetry considerations as the system inertia matrix; see Section 3.4.2.

3.4.5 Computation of the Restoring Forces and Moments

In this section it is necessary to distinguish submersibles from floating vessels.

Underwater Vehicles (Submerged Bodies)

If the gravitational and buoyancy forces are computed in the *CG*, Formula (3.120) for submersibles reduces to:

$$\mathbf{g}^{cg}(\boldsymbol{\eta}) = \begin{bmatrix} (W-B)\sin\theta \\ -(W-B)\cos\theta\sin\phi \\ -(W-B)\cos\theta\cos\phi \\ y_b B\cos\theta\cos\phi - z_b B\cos\theta\sin\phi \\ -z_b B\sin\theta - x_b B\cos\theta\cos\phi \\ x_b B\cos\theta\sin\phi + y_b B\sin\theta \end{bmatrix} \tag{3.230}$$

where x_b, y_b, and z_b are the coordinates of *CB* with respect to *CG*. This expressions can be transformed from the *CG* to the *b*-frame coordinate origin *O* by:

$$\mathbf{g}(\boldsymbol{\eta}) = \mathbf{H}^{\top}(\mathbf{r}_g^b)\mathbf{g}^{cg}(\boldsymbol{\eta}) \tag{3.231}$$

Matlab:
The restoring forces and moments are generated in the GNC toolbox according to:

```
g_CG = gvect(W,B,theta,phi,[0,0,0]',[xb,yb,zb]')
g    = Hmtrx(r_g)'*g_CG
```

Surface Vessels (Ships and Semi-Submersibles)

For floating vessels the expression (3.135) can be transformed to *CG* by using (3.231). In practice it is common to assume that small angle (linear) theory holds. Hence:

$$\mathbf{G}^{cg} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_z & 0 & 0 & 0 \\ 0 & 0 & 0 & -K_\phi & 0 & 0 \\ 0 & 0 & 0 & 0 & -M_\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.232}$$

yields:

$$\mathbf{G} = \mathbf{H}^{\top}(\mathbf{r}_g^b)\mathbf{G}^{cg}\mathbf{H}(\mathbf{r}_g^b) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_z & y_g Z_z & -x_g Z_z & 0 \\ 0 & 0 & y_g Z_z & -y_g^2 Z_z - K_\phi & x_g y_g Z_z & 0 \\ 0 & 0 & -x_g Z_z & x_g y_g Z_z & -x_g^2 Z_z - M_\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.233}$$

Matlab:
The 6×6 system spring stiffness matrix $\mathbf{G}$ about an arbitrarily point O is computed by using the GNC toolbox function `Gmtrx.m`:

```
A_wp = 1000              % water plane area
nabla = 10000            % displasement
GMT = 1                  % transverse metacentric heights
GML = 2                  % lateral metacentric heights
r_g = [1 0 10]'          % location of CG w.r.t. O

% Spring stiffness matrix
G = Gmtrx(nabla,A_wp,GMT,GML,r_g)
```

This produces the numerical result:

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10055250 & 0 & -10055250 & 0 \\ 0 & 0 & 0 & 100552500 & 0 & 0 \\ 0 & 0 & -10055250 & 0 & 211160250 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

3.5 Standard Models for Marine Vessels

In this section the main results of Chapter 3 are applied to derive a set of standard models for ships, semi-submersibles, and underwater vehicles. The following subsystems are discussed:

- **Surge model:** velocity u
- **Maneuvering model (sway and yaw):** velocities v and r
- **Horizontal motion (surge, sway, and yaw):** velocities u, v, and r
- **Longitudinal motion (surge, heave, and pitch):** velocities u, w, and q
- **Lateral motion: (sway, roll, and yaw):** velocities v, p, and r

Figure 3.14: Offshore supply vessel.

Several models for ships, semi-submersibles, and underwater vehicles are included in the Matlab™ GNC Toolbox as m-files. The toolbox can be downloaded from:

http://www.marinecybernetics.com

A list of these models are found by typing: *help vesselmodels*. A Simulink library for simulation of marine control systems applied to different vessels is also available.

3.5.1 3 DOF Horizontal Model

The horizontal motion of a ship or semi-submersible is usually described by the motion components in surge, sway, and yaw. Therefore, we choose $\boldsymbol{\nu} = [u, v, r]^\top$ and $\boldsymbol{\eta} = [n, e, \psi]^\top$. This implies that the dynamics associated with the motion in heave, roll, and pitch are neglected, that is $w = p = q = 0$.

In addition, low-speed applications–i.e., dynamically positioned ships where $U \approx 0$, and maneuvering at high speed are treated separately.

Low-Speed Model for Dynamically Positioned Ship

For the horizontal motion of a vessel the kinematic equations of motion reduce from the general 6 DOF expression (2.16) to one principal rotation about the z-axis:

$$\mathbf{J}(\boldsymbol{\eta}) \overset{3\,\text{DOF}}{=} \mathbf{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.234}$$

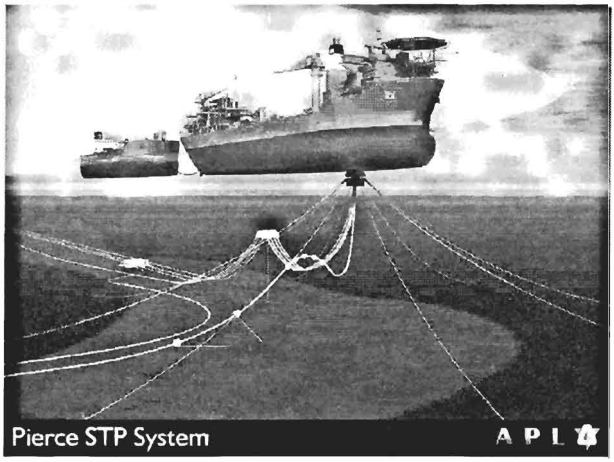


Figure 3.15: Dynamically positioned ship.

Assuming that the ship has homogeneous mass distribution and xz-plane symmetry so that:

$$I_{xy} = I_{yz} = 0 \tag{3.235}$$

Let the b-frame coordinate origin be set in the center line of the ship, such that $y_g = 0$. Under the previously stated assumptions, the matrices (3.55) and (3.66) associated with the rigid-body dynamics reduce to:

$$\mathbf{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix}, \quad \mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \tag{3.236}$$

Notice that surge is decoupled from sway and yaw in $\mathbf{M}_{RB}$ due to symmetry considerations of the system inertia matrix (see Section 3.4.2). For simplicity, it is assumed that the center of added mass is equal to the center of gravity. This allows for the following reduction of (3.92) and (3.102):

$$\mathbf{M}_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -Y_{\dot{r}} & -N_{\dot{r}} \end{bmatrix}, \quad \mathbf{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & Y_{\dot{v}} v + Y_{\dot{r}} r \\ 0 & 0 & -X_{\dot{u}} u \\ -Y_{\dot{v}} v - Y_{\dot{r}} r & X_{\dot{u}} u & 0 \end{bmatrix} \tag{3.237}$$

Hence, $\mathbf{M} = \mathbf{M}^\top$ and $\mathbf{C}(\boldsymbol{\nu}) = -\mathbf{C}^\top(\boldsymbol{\nu})$, that is:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{3.238}$$

$$\mathbf{C}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -(m - Y_{\dot{v}})v - (mx_g - Y_{\dot{r}})r \\ 0 & 0 & (m - X_{\dot{u}})u \\ (m - Y_{\dot{v}})v + (mx_g - Y_{\dot{r}})r & -(m - X_{\dot{u}})u & 0 \end{bmatrix} \tag{3.239}$$

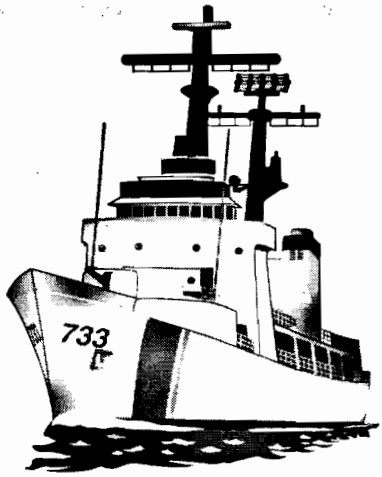


Figure 3.16: Ship maneuvering.

As for the system inertia matrix, linear damping in surge is decoupled from sway and yaw. This implies that:

$$\mathbf{D} = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \tag{3.240}$$

Linear damping is a good assumption for low-speed applications (see Section 3.2.2). Similarly the quadratic velocity terms given by $\mathbf{C}(\nu)\nu$ are negligible in DP. Hence, a model that is well suited for ship positioning is:

$$\dot{\eta} = \mathbf{R}(\psi)\nu \tag{3.241}$$

$$\mathbf{M}\dot{\nu} + \mathbf{D}\nu = \tau \tag{3.242}$$

where

$$\tau = \mathbf{B}\mathbf{u} \tag{3.243}$$

Here $\mathbf{B}$ is the control matrix describing the thruster configuration and $\mathbf{u}$ is the control input.

Nonlinear Maneuvering Model

At higher speeds the assumptions that $\mathbf{D}(\nu) = [\mathbf{D} + \mathbf{D}_n(\nu)] \approx \mathbf{D}$ and $\mathbf{C}(\nu) \approx \mathbf{0}$ are violated. Hence, the nonlinear velocity terms must be included; see Section 3.2.2. This suggests the following 3 DOF nonlinear maneuvering model:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{3.244}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau} \tag{3.245}$$

3.5.2 Decoupled Models for Forward Speed/Maneuvering

For vessels moving at constant (or at least slowly-varying) forward speed

$$U = \sqrt{u^2 + v^2} \approx u \tag{3.246}$$

the 3 DOF maneuvering model of Section 3.5.1 can be decoupled in a forward speed (surge) model and a sway-yaw subsystem for maneuvering.

Forward Speed Model

Starboard-port symmetry implies that surge is decoupled from sway and yaw. Hence, the surge equation in Section 3.5.1 can be written in component form as:

$$(m - X_{\dot{u}})\dot{u} - X_u u - X_{|u|u}\,|u|\,u = \tau_1 \tag{3.247}$$

where τ_1 is the sum of control forces in surge. Notice that both linear and quadratic damping have been included in order to cover low- and high-speed applications. For this decoupled model a forward speed controller can be designed by only using the forward speed u for feedback.

2 DOF Linear Maneuvering Model (Sway-Yaw Subsystem)

A linear maneuvering model is based on the assumption that the cruise speed:

$$u = u_o \approx \text{constant} \tag{3.248}$$

while v and r are assumed to be small.

Representation 1: The 2nd and 3rd rows in (3.239) with $u = u_o$, yields:

$$\begin{aligned} \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} &= \begin{bmatrix} (m - X_{\dot{u}})u_o r \\ (m - Y_{\dot{v}})u_o v + (mx_g - Y_{\dot{r}})u_o r - (m - X_{\dot{u}})u_o v \end{bmatrix} \\ &= \begin{bmatrix} 0 & (m - X_{\dot{u}})u_o \\ (X_{\dot{u}} - Y_{\dot{v}})u_o & (mx_g - Y_{\dot{r}})u_o \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} \end{aligned} \tag{3.249}$$

Assuming that the ship is controlled by a single rudder:

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{b}\delta \\ &= \begin{bmatrix} -Y_\delta \\ -N_\delta \end{bmatrix} \delta \end{aligned} \tag{3.250}$$

and that:

$$\mathbf{D}(\boldsymbol{\nu}) = \mathbf{D} + \mathbf{D}_n(\boldsymbol{\nu}) \approx \mathbf{D} \tag{3.251}$$

then:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_o)\boldsymbol{\nu} = \mathbf{b}\delta \tag{3.252}$$

where $\boldsymbol{\nu} = [v, r]^\top$ and:

$$\mathbf{M} = \begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{3.253}$$

$$\mathbf{N}(u_o) = \begin{bmatrix} -Y_v & (m - X_{\dot{u}})u_o - Y_r \\ (X_{\dot{u}} - Y_{\dot{v}})u_o - N_v & (mx_g - Y_{\dot{r}})u_o - N_r \end{bmatrix} \tag{3.254}$$

$$\mathbf{b} = \begin{bmatrix} -Y_\delta \\ -N_\delta \end{bmatrix} \tag{3.255}$$

Representation 2: An alternative representation is (Davidson and Schiff 1946):

$$\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_{RB} \tag{3.256}$$

where

$$\boldsymbol{\tau}_{RB} = -\begin{bmatrix} Y_\delta \\ N_\delta \end{bmatrix}\delta + \begin{bmatrix} Y_{\dot{v}} & Y_{\dot{r}} \\ N_{\dot{v}} & N_{\dot{r}} \end{bmatrix}\dot{\boldsymbol{\nu}} + \begin{bmatrix} Y_v & Y_r \\ N_v & N_r \end{bmatrix}\boldsymbol{\nu} \tag{3.257}$$

The rudder angle δ is defined such that a positive rudder angle yields a positive yaw rate r. Furthermore it is assumed that the hydrodynamic forces $\boldsymbol{\tau}_{RB}$ in (3.256) is linear in δ, $\dot{\boldsymbol{\nu}}$ and $\boldsymbol{\nu}$ as defined by (3.257). Substituting (3.257) into (3.256), yields:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_o)\boldsymbol{\nu} = \mathbf{b}\delta \tag{3.258}$$

where

$$\mathbf{M} = \begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{3.259}$$

$$\mathbf{N}(u_o) = \begin{bmatrix} -Y_v & mu_o - Y_r \\ -N_v & mx_g u_o - N_r \end{bmatrix} \tag{3.260}$$

$$\mathbf{b} = \begin{bmatrix} -Y_\delta \\ -N_\delta \end{bmatrix} \tag{3.261}$$

In this model the *Munk moment* $(X_{\dot{u}} - Y_{\dot{v}})u_o r$ is missing in the yaw equation (see the last row in (3.254)). This is a destabilizing moment known from aerodynamics which tries to turn the vessel; see Faltinsen (1990, pp. 188–189) . This term is, however, often included in a nonlinear viscous term $N_{ur}ur$ instead. Also notice that the less important terms $X_{\dot{u}}u_o r$ and $Y_{\dot{r}}u_o r$ are removed from $\mathbf{N}(u_o)$ when compared to (3.254). All missing terms terms are due to the $\mathbf{C}_A(\boldsymbol{\nu})$ matrix which is omitted in (3.257).

1 DOF Autopilot Model (Yaw Subsystem)

A linear autopilot model for course control can be derived from the maneuvering model

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_o)\boldsymbol{\nu} = \mathbf{b}\delta \tag{3.262}$$

by defining the yaw rate r as output:

$$r = \mathbf{c}^\top \boldsymbol{\nu}, \quad \mathbf{c}^\top = [0, 1] \tag{3.263}$$

Hence, application of the *Laplace transformation* yields:

$$\frac{r}{\delta}(s) = \frac{K(1 + T_3 s)}{(1 + T_1 s)(1 + T_2 s)} \tag{3.264}$$

This is referred to as *Nomoto's 2nd-order model* (Nomoto *et al.* 1957). The *1st-order Nomoto model* is obtained by defining the equivalent time constant $T = T_1 + T_2 - T_3$ such that:

$$\frac{r}{\delta}(s) = \frac{K}{(1 + Ts)} \tag{3.265}$$

Finally, $\dot{\psi} = r$ yields:

$$\frac{\psi}{\delta}(s) = \frac{K}{s(1 + Ts)} \tag{3.266}$$

which is the transfer function that is used in most commercial autopilot systems.

3.5.3 Longitudinal and Lateral Models

The 6 DOF equations of motion can in many cases be divided into two non-interacting (or lightly interacting) subsystems:

- **Longitudinal subsystem:** states u, w, q, and θ
- **Lateral subsystem:** states v, p, r, ϕ, and ψ

This decomposition is good for slender bodies (large length/width ratio) as shown in Figure 3.17; typical applications are aircraft, missiles, and submarines (Gertler and Hagen 1967, Feldman 1979, Tinker 1982). This can also be seen from the expression of the system inertia matrix in the case of starboard-port symmetry (see Section 3.4.2):

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & m_{13} & 0 & m_{15} & 0 \\ 0 & m_{22} & 0 & m_{24} & 0 & m_{26} \\ m_{31} & 0 & m_{33} & 0 & m_{35} & 0 \\ 0 & m_{42} & 0 & m_{44} & 0 & m_{46} \\ m_{51} & 0 & m_{53} & 0 & m_{55} & 0 \\ 0 & m_{62} & 0 & m_{64} & 0 & m_{66} \end{bmatrix} \tag{3.267}$$

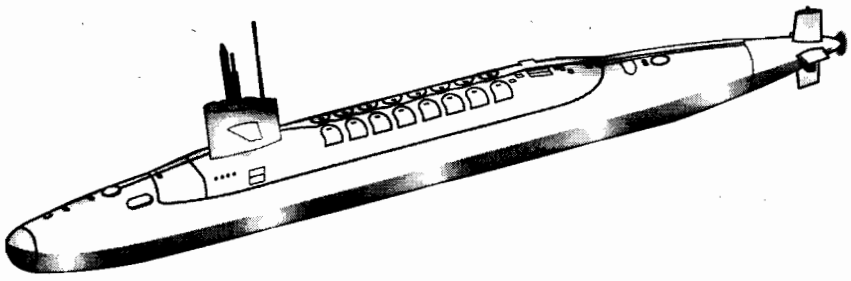


Figure 3.17: Slender body submarine (large length/width ratio).

which clearly confirms that the two subsystems:

$$\mathbf{M}_{\text{long}} = \begin{bmatrix} m_{11} & m_{13} & m_{15} \\ m_{31} & m_{33} & m_{35} \\ m_{51} & m_{53} & m_{55} \end{bmatrix}, \quad \mathbf{M}_{\text{lat}} = \begin{bmatrix} m_{22} & m_{24} & m_{26} \\ m_{42} & m_{44} & m_{46} \\ m_{62} & m_{64} & m_{66} \end{bmatrix} \tag{3.268}$$

are decoupled.

Longitudinal Subsystem

Under the assumption that the lateral states v, p, r, ϕ are small, the longitudinal kinematics for surge, heave, and pitch are, see (2.16) and (2.26):

$$\begin{bmatrix} \dot{d} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} -\sin\theta \\ 0 \end{bmatrix} u \tag{3.269}$$

For simplicity, it is assumed that higher order damping can be neglected, that is $\mathbf{D}_n(\boldsymbol{\nu}) = \mathbf{0}$. Coriolis is, however, modelled by assuming that $u \gg 0$ and that 2nd-order terms in v, w, p, q, and r are small. Hence, from (3.66) it is seen that:

$$\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \begin{bmatrix} m(y_g q + z_g r)p - m(x_g q - w)q - m(x_g r + v)r \\ -m(z_g p - v)p - m(z_g q + u)q + m(x_g p + y_g q)r \\ m(x_g q - w)u - m(z_g r + x_g p)v + m(z_g q + u)w \\ +(I_{yz} q + I_{xz} p - I_z r)p + (-I_{xz} r - I_{xy} q + I_x p)r \end{bmatrix}$$

such that:

$$\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} \approx \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & 0 & -mu \\ 0 & 0 & mx_g u \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} \tag{3.270}$$

Notice that $\mathbf{C}_{RB}(\boldsymbol{\nu}) \neq -\mathbf{C}_{RB}^{\top}(\boldsymbol{\nu})$ for the decoupled model. Assuming a diagonal $\mathbf{M}_A$ as in Example 3.2, the corresponding added mass terms are:

$$\begin{aligned}\mathbf{C}_A(\boldsymbol{\nu})\boldsymbol{\nu} &= \begin{bmatrix} -Z_{\dot{w}}wq + Y_{\dot{v}}vr \\ -Y_{\dot{v}}vp + X_{\dot{u}}uq \\ (Z_{\dot{w}}-X_{\dot{u}})uw + (N_{\dot{r}}-K_{\dot{p}})pr \end{bmatrix} \\ &\approx \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & X_{\dot{u}}u \\ 0 & (Z_{\dot{w}}-X_{\dot{u}})u & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} \end{aligned} \tag{3.271}$$

According to 3.6) and (3.120) with $W = B$ and $\boldsymbol{x}_g = \boldsymbol{x}_b$, the dynamics becomes:

$$\begin{bmatrix} m - X_{\dot{u}} & -X_{\dot{w}} & mz_g - X_{\dot{q}} \\ -X_{\dot{w}} & m - Z_{\dot{w}} & -mx_g - Z_{\dot{q}} \\ mz_g - X_{\dot{q}} & -mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \end{bmatrix}$$

$$+ \begin{bmatrix} -X_u & -X_w & -X_q \\ -Z_u & -Z_w & -Z_q \\ -M_u & -M_w & -M_q \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -(m - X_{\dot{u}})u \\ 0 & (Z_{\dot{w}} - X_{\dot{u}})u & mx_g u \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ W\, BG_z \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_3 \\ \tau_5 \end{bmatrix} \tag{3.272}$$

This model is the basis for forward speed control (state u) and depth/diving autopilot design (states w, q, θ). If the forward speed is stabilized by a forward speed controller such that:

$$u = u_o = \text{constant} \tag{3.273}$$

forward speed can be eliminated from the longitudinal equations of motion such that:

$$\begin{bmatrix} m - Z_{\dot{w}} & -mx_g - Z_{\dot{q}} \\ -mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -Z_w & -Z_q \\ -M_w & -M_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & -(m - X_{\dot{u}})u_o \\ (Z_{\dot{w}} - X_{\dot{u}})u_o & mx_g u_o \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ BG_z W \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_3 \\ \tau_5 \end{bmatrix}$$

Moreover, if $\dot{w} = w = 0$ (constant depth) and θ is small such that $\sin\theta \approx \theta$, the *linear pitch dynamics* becomes:

$$(I_y - M_{\dot{q}})\ddot{\theta} - M_q\dot{\theta} + BG_z W\theta = \tau_5 \tag{3.274}$$

where the natural frequency is:

$$\omega_\theta = \sqrt{\frac{BG_z\, W}{(I_y - M_{\dot{q}})}} \tag{3.275}$$

Lateral Subsystem

Under the assumption that the longitudinal states u, w, p, r, ϕ and θ are small, the lateral kinematics, see (2.16) and (2.26), reduce to:

$$\dot{\phi} = p \tag{3.276}$$
$$\dot{\psi} = r \tag{3.277}$$

Again it is assumed that higher order velocity terms can be neglected so that $\mathbf{D}_n(\boldsymbol{\nu}) = \mathbf{0}$, and that the Coriolis terms in $u = u_o$ are the most important, see (3.66):

$$\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \begin{vmatrix} -m(y_g p + w)p + m(z_g r + x_g p)q - m(y_g r - u)r \\ -m(y_g q + z_g r)u + m(y_g p + w)v + m(z_g p - v)w \\ m(x_g r + v)u + m(y_g r - u)v - m(x_g p + y_g q)w \end{vmatrix}$$

$$\begin{bmatrix} +(-I_{yz}q - I_{xz}p + I_z r)q + (I_{yz}r + I_{xy}p - I_y q)r \\ +(-I_{yz}r - I_{xy}p + I_y q)p + (I_{xz}r + I_{xy}q - I_x p)q \end{bmatrix}$$

Hence:

$$\mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} \approx \begin{bmatrix} 0 & 0 & mu_o \\ 0 & 0 & 0 \\ 0 & 0 & mx_g u_o \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} \tag{3.278}$$

Under the assumption of a diagonal $\mathbf{M}_A$ structure as in Example 3.2, the corresponding added mass terms are:

$$\begin{aligned} \mathbf{C}_A(\boldsymbol{\nu})\boldsymbol{\nu} &= \begin{bmatrix} Z_{\dot{w}}wp - X_{\dot{u}}ur \\ (Y_{\dot{v}} - Z_{\dot{w}})vw + (M_{\dot{q}} - N_{\dot{r}})qr \\ (X_{\dot{u}} - Y_{\dot{v}})uv + (K_{\dot{p}} - M_{\dot{q}})pq \end{bmatrix} \\ &\approx \begin{bmatrix} 0 & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 \\ (X_{\dot{u}} - Y_{\dot{v}})u & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} \end{aligned} \tag{3.279}$$

Next, assume that $W = B$, $x_g = x_b$, and $y_g = y_b$. Then (3.6) and (3.120) reduces to:

$$\begin{bmatrix} m - Y_{\dot{v}} & -mz_g - Y_{\dot{p}} & mx_g - Y_{\dot{r}} \\ -mz_g - Y_{\dot{p}} & I_x - K_{\dot{p}} & -I_{zx} - K_{\dot{r}} \\ mx_g - Y_{\dot{r}} & -I_{zx} - K_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \end{bmatrix}$$

$$+ \begin{bmatrix} -Y_v & -Y_p & -Y_r \\ -M_v & -M_p & -M_r \\ -N_v & -N_p & -N_r \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} \tag{3.280}$$

$$+ \begin{bmatrix} 0 & 0 & (m - X_{\dot{u}})u \\ 0 & 0 & 0 \\ (X_{\dot{u}} - Y_{\dot{v}})u & 0 & mx_g u \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ W\,BG_z \sin\phi \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_2 \\ \tau_4 \\ \tau_6 \end{bmatrix}$$

For vehicles where $\dot{p}$ and p are small (small roll motions) and the speed is $u = u_o$, this reduces to:

$$\begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} -Y_v & -Y_r \\ -N_v & -N_r \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} + \begin{bmatrix} 0 & (m - X_{\dot{u}})u_o \\ (X_{\dot{u}} - Y_{\dot{v}})u_o & mx_g u_o \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} = \begin{bmatrix} \tau_2 \\ \tau_6 \end{bmatrix}$$

which is the sway-yaw maneuvering model (Representation 1). The decoupled *linear roll equation* under the assumption of a small ϕ is:

$$(I_x - K_{\dot{p}})\ddot{\phi} - K_p\dot{\phi} + WBG_z\phi = \tau_4 \tag{3.281}$$

for which the natural frequency is:

$$\omega_\phi = \sqrt{\frac{BG_z W}{(I_x - K_{\dot{p}})}} \tag{3.282}$$

3.6 Exercises

Exercise 3.1 *Consider a neutrally buoyant spherical shaped underwater vehicle. The radius of the sphere is* $R = 1.0$ *(m) and the density of water is* $\rho = 1000$ *(kg/m*^{3}*). For simplicity it is assumed that* $\mathbf{M}_A = \mathbf{M}_{RB}$*. The linear part of the damping matrix is:*

$$\mathbf{D} = \text{diag}\{100, 150, 200, 1, 2, 5\}$$

The vertical distance between the CG and CB are 0.2 *m and both points are located on the vertical axes through the center of sphere. Let the body-fixed coordinate system* O *be located at the CB.*

a) Compute the Coriolis and centripetal matrix $\mathbf{C}(\nu)$ *as a function* ν *and explain why this term can be neglected during station-keeping.*

b) Compute the mass m*. The inertia matrix with respect to CG is*

$$\mathbf{I}_{cg} = \text{diag}\{1000, 1000, 500\}$$

Compute the inertia matrix $\mathbf{I}_o$ *with respect to the body-fixed coordinate origin* O*.*

c) Compute the time constants for the 1st-order systems in surge, sway, *and* yaw *under the assumption of station-keeping.*

d) Compute the heave, roll, *and* pitch *periods under the assumption of station-keeping.*

Hint: *The volume of the sphere is* $V = \frac{4}{3}\pi R^3$*.*

Exercise 3.2 *Open the container ship model container.m (remember to type gnc in Matlab to add the GNC toolbox path to the Matlab path) in the Matlab editor.*

a) Write the m-file model in the standard form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \underbrace{[\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu})]}_{\mathbf{N}(\boldsymbol{\nu})}\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w}$$

by putting all m-file terms into the proper matrices and vectors. It is convenient to use the matrix $\mathbf{N}(\boldsymbol{\nu})$ *instead of the matrices* $\mathbf{C}(\boldsymbol{\nu})$ *and* $\mathbf{D}(\boldsymbol{\nu})$.

b) Is it possible to split the matrix $\mathbf{N}(\boldsymbol{\nu})$ *into two unique matrices* $\mathbf{C}(\boldsymbol{\nu})$ *and* $\mathbf{D}(\boldsymbol{\nu})$*? Hint: for each* $\mathbf{M}$ *there exists at least one* $\mathbf{C}(\boldsymbol{\nu})$.

c) Write a Matlab or Simulink program and simulate the container ship for different maneuvers.

Exercise 3.3 *Consider a marine vessel in 6 DOF where:*

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}$$

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

where $\boldsymbol{\nu} = [u, v, w, p, q, r]^\top$ *and* $\boldsymbol{\eta} = [n, e, d, \phi, \theta, \psi]^\top$. *Let* $V(\boldsymbol{\eta}, \boldsymbol{\nu})$ *be a positive definite energy function:*

$$V(\boldsymbol{\eta}, \boldsymbol{\nu}) = \underbrace{\frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu}}_{\text{kinetic energy}} + \underbrace{\frac{1}{2}\boldsymbol{\eta}^\top \mathbf{K}_p\boldsymbol{\eta}}_{\text{potential energy}}$$

where $\mathbf{M}^\top = \mathbf{M} > 0$ *and* $\mathbf{K}_p^\top = \mathbf{K}_p > 0$.
a) Show that:

$$\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = \boldsymbol{\nu}^\top \left[\boldsymbol{\tau} - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{g}(\boldsymbol{\eta}) + \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\boldsymbol{\eta}\right]$$

b) Assuming that $\boldsymbol{\nu}^T\mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} > 0$, *find a feedback control law* $\boldsymbol{\tau}$ *such that:*

$$\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = -\boldsymbol{\nu}^\top[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d]\boldsymbol{\nu} \leq 0, \quad \mathbf{K}_d > 0$$

and explain why this makes the vessel come to rest, $\boldsymbol{\nu}(t) \to 0$ *as* $t \to \infty$.
c) The assumption that $\mathbf{K}_p^\top = \mathbf{K}_p > 0$ *is relaxed to* $\mathbf{K}_p > 0$, *that is* $\mathbf{x}^\top\mathbf{K}_p\mathbf{x} > 0$, $\mathbf{x} \neq \mathbf{0}$. *Is it possible to find a control* $\boldsymbol{\tau}$ *for this case satisfying*

$$\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = -\boldsymbol{\nu}^\top[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d]\boldsymbol{\nu} \leq 0$$

Hint: $\mathbf{x}^\top\mathbf{A}\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\mathbf{x} + \frac{1}{2}\mathbf{x}^\top(\mathbf{A} - \mathbf{A}^\top)\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$ *since* $\mathbf{x}^\top(\mathbf{A} - \mathbf{A}^\top)\mathbf{x} = 0$.

Chapter 4

Models for Wind, Waves and Ocean Currents

In Chapters 2 and 3 a nonlinear model structure for marine vessels in 6 DOF was derived. In this chapter simulation models for environmental disturbances are presented. These include models for:

- Wind
- Waves
- Ocean currents

The purpose of the chapter is to present models for simulation, testing, and verification of feedback control systems–i.e., of systems in *closed loop*. Hydrodynamic models for more accurate computations of sealoads and *open loop* prediction of vessel motion are found in Faltinsen (1990), Newman (1977), and Sarpkaya (1981).

Superposition of Wind and Wave Disturbances

For control system design it is common to assume the *principle of superposition* when considering wind and wave disturbances. For most marine control applications this is a good approximation. In general, the environmental disturbances will be highly nonlinear, and both additive and multiplicative to the dynamic equations of motion. These effects are included in vessel simulators that are produced for human operators. When simulating a system under feedback control, many of these effects are suppressed in closed loop.

In Chapter 3 it was shown that the nonlinear dynamic equations of motion can be written:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{4.1}$$

The principle of superposition suggests that the wind and wave-induced disturbances are added to the right-hand side of (4.1) by defining:

$$\mathbf{w} = \mathbf{w}_{\text{wind}} + \mathbf{w}_{\text{wave}} \tag{4.2}$$

where $\mathbf{w}_{\text{wind}} \in \mathbb{R}^6$ and $\mathbf{w}_{\text{wave}} \in \mathbb{R}^6$ represent the generalized forces due to wind and waves. Simple models for simulation of generalized wind and wave forces are presented in Sections 4.1 and 4.2.

Equations of Relative Motion for Simulation of Ocean Currents

The effect due to ocean currents is usually simulated in closed-loop by introducing the relative velocity vector:

$$\boldsymbol{\nu}_r = \boldsymbol{\nu} - \boldsymbol{\nu}_c \tag{4.3}$$

where $\boldsymbol{\nu}_c \in \mathbb{R}^6$ is the body-fixed current velocity vector.

For a slowly-varying current profile, $\dot{\boldsymbol{\nu}}_c \approx 0$, the equations of motion become:

$$\underbrace{\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta})}_{\text{rigid-body terms}} + \underbrace{\mathbf{M}_A\dot{\boldsymbol{\nu}} + \mathbf{C}_A(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r}_{\text{hydrodynamic terms}} = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{4.4}$$

In the linear case this reduces to:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}\boldsymbol{\nu}_r + \mathbf{G}\boldsymbol{\eta} = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{4.5}$$

Models for simulation of ocean currents in terms of $\boldsymbol{\nu}_c$ are presented in Section 4.3.

4.1 Wind Models

Wind is defined as the movement of air relative to the surface of the Earth. Mathematical models of wind forces and moments are used in vessel control systems to improve the performance and robustness of the system in extreme conditions. Some of these models are presented in the forthcoming sections.

4.1.1 Wind Forces and Moments

Let V_w and ψ_w denote the wind speed and direction, respectively. In order to determine the local velocity h (m) above the sea surface a boundary-layer profile can be used, see Bretschneider (1969):

$$V_w(h) = V_w(10) \cdot (h/10)^{1/7} \tag{4.6}$$

where $V_w(10)$ is the relative wind velocity 10 (m) above the sea surface.

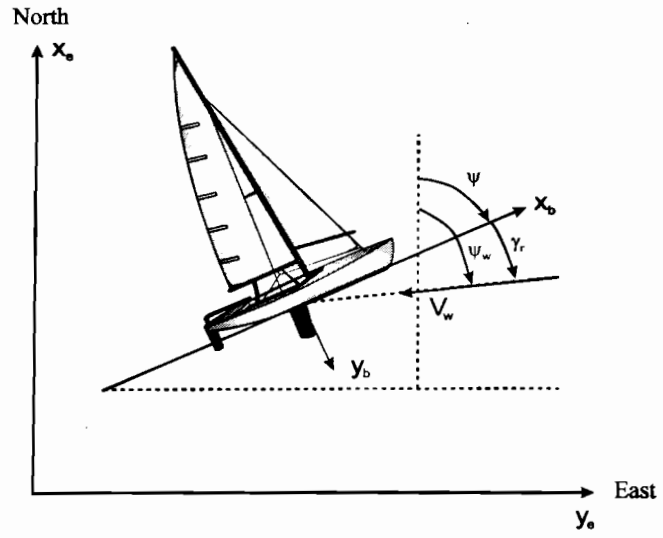


Figure 4.1: Definition of wind speed V_w and direction γ_r.

The wind forces and moments acting on a vessel can then be defined in terms of relative wind speed V_r and the angle γ_r according to:

$$V_r = \sqrt{u_r^2 + v_r^2}, \qquad \gamma_r = \tan^{-1}(v_r/u_r) = \psi_w - \psi \tag{4.7}$$

where the components of V_r in the x- and y-directions are:

$$u_r = V_w \cos(\gamma_r) - u, \qquad v_r = V_w \sin(\gamma_r) - v \tag{4.8}$$

Here, $\gamma_R = \psi_w - \psi$ can be interpreted as the angle of the wind relative to the ship bow; see Figure 4.1.

The wind speed V_w and its direction can be measured by a wind sensor. These measurements should be proper filtered since only the mean wind forces and moments can be compensated for by the autopilot. In fact, since the inertia of the vessel is so large, it is unnecessary for the control system to compensate for wind gust. In order to implement wind feedforward compensation for a surface vessel, a 3 DOF wind model as a function of relative wind speed and direction, V_r and γ_r, is needed. For this purpose, the generalized force vector is:

$$\mathbf{w}_{\text{wind}} = [X_{\text{wind}}, Y_{\text{wind}}, N_{\text{wind}}]^\top \tag{4.9}$$

Two models for numerical computation of $X_{\text{wind}}, Y_{\text{wind}}$, and N_{wind} will now be discussed.

4.1.2 Wind Resistance of Merchant Ships (Isherwood 1972)

Isherwood (1972) suggested that one can write the wind forces (surge and sway) and moment (yaw) according to:

$$X_{\text{wind}} = \frac{1}{2} C_X(\gamma_r) \rho_a V_r^2 A_T \quad \text{(N)} \tag{4.10}$$

$$Y_{\text{wind}} = \frac{1}{2} C_Y(\gamma_r) \rho_a V_r^2 A_L \quad \text{(N)} \tag{4.11}$$

$$N_{\text{wind}} = \frac{1}{2} C_N(\gamma_r) \rho_a V_r^2 A_L L \quad \text{(N m)} \tag{4.12}$$

where C_X and C_Y are the empirical force coefficients, C_N is a moment coefficient, ρ_a (kg/m^3) is the density of air, A_T (m^2) and A_L (m^2) are the transverse and lateral projected areas, and L (m) is the overall length of the ship. Notice that V_r is given in knots.

In the work of Isherwood (1972) the measured data were analyzed by multiple regression techniques using the following 8 parameters:

L	–	length overall
B	–	beam
A_L	–	lateral projected area
A_T	–	transverse projected area
A_{SS}	–	lateral projected area of superstructure
S	–	length of perimeter of lateral projection of model excluding waterline and slender bodies such as masts and ventilators
C	–	distance from bow of centroid of lateral projected area
M	–	number of distinct groups of masts or king posts seen in lateral projection; king posts close against the bridge front are not included

Table 4.1: Wind force parameters in surge, sway, and yaw (Isherwood 1972).

γ_r (deg)	A_0	A_1	A_2	A_3	A_4	A_5	A_6	S.E.
0	2.152	−5.00	0.243	−0.164	–	–	–	0.086
10	1.714	−3.33	0.145	−0.121	–	–	–	0.104
20	1.818	−3.97	0.211	−0.143	–	–	0.033	0.096
30	1.965	−4.81	0.243	−0.154	–	–	0.041	0.117
40	2.333	−5.99	0.247	−0.190	–	–	0.042	0.115
50	1.726	−6.54	0.189	−0.173	0.348	–	0.048	0.109
60	0.913	−4.68	–	−0.104	0.482	–	0.052	0.082
70	0.457	−2.88	–	−0.068	0.346	–	0.043	0.077
80	0.341	−0.91	–	−0.031	–	–	0.032	0.090
90	0.355	–	–	–	−0.247	–	0.018	0.094
100	0.601	–	–	–	−0.372	–	−0.020	0.096
110	0.651	1.29	–	–	−0.582	–	−0.031	0.090
120	0.564	2.54	–	–	−0.748	–	−0.024	0.100
130	−0.142	3.58	–	0.047	−0.700	–	−0.028	0.105
140	−0.677	3.64	–	0.069	−0.529	–	−0.032	0.123
150	−0.723	3.14	–	0.064	−0.475	–	−0.032	0.128
160	−2.148	2.56	–	0.081	–	1.27	−0.027	0.123
170	−2.707	3.97	−0.175	0.126	–	1.81	–	0.115
180	−2.529	3.76	−0.174	0.128	–	1.55	–	0.112
							Mean S.E.	0.103

γ_r (deg)	B_0	B_1	B_2	B_3	B_4	B_5	B_6	S.E.
10	0.096	0.22	–	–	–	–	–	0.015
20	0.176	0.71	–	–	–	–	–	0.023
30	0.225	1.38	–	0.023	–	−0.29	–	0.030
40	0.329	1.82	–	0.043	–	−0.59	–	0.054
50	1.164	1.26	0.121	–	−0.242	−0.95	–	0.055
60	1.163	0.96	0.101	–	−0.177	−0.88	–	0.049
70	0.916	0.53	0.069	–	–	−0.65	–	0.047
80	0.844	0.55	0.082	–	–	−0.54	–	0.046
90	0.889	–	0.138	–	–	−0.66	–	0.051
100	0.799	–	0.155	–	–	−0.55	–	0.050
110	0.797	–	0.151	–	–	−0.55	–	0.049
120	0.996	–	0.184	–	−0.212	−0.66	0.34	0.047
130	1.014	–	0.191	–	−0.280	−0.69	0.44	0.051
140	0.784	–	0.166	–	−0.209	−0.53	0.38	0.060
150	0.536	–	0.176	−0.029	−0.163	–	0.27	0.055
160	0.251	–	0.106	−0.022	–	–	–	0.036
170	0.125	–	0.046	−0.012	–	–	–	0.022
							Mean S.E.	0.044

γ_r (deg)	C_0	C_1	C_2	C_3	C_4	C_5	S.E.
10	0.0596	0.061	–	–	–	−0.074	0.0048
20	0.1106	0.204	–	–	–	−0.170	0.0074
30	0.2258	0.245	–	–	–	−0.380	0.0105
40	0.2017	0.457	–	0.0067	–	−0.472	0.0137
50	0.1759	0.573	–	0.0118	–	−0.523	0.0149
60	0.1925	0.480	–	0.0115	–	−0.546	0.0133
70	0.2133	0.315	–	0.0081	–	−0.526	0.0125
80	0.1827	0.254	–	0.0053	–	−0.443	0.0123
90	0.2627	–	–	–	–	−0.508	0.0141
100	0.2102	–	−0.0195	–	0.0335	−0.492	0.0146
110	0.1567	–	−0.0258	–	0.0497	−0.457	0.0163
120	0.0801	–	−0.0311	–	0.0740	−0.396	0.0179
130	−0.0189	–	−0.0488	0.0101	0.1128	−0.420	0.0166
140	0.0256	–	−0.0422	0.0100	0.0889	−0.463	0.0162
150	0.0552	–	−0.0381	0.0109	0.0689	−0.476	0.0141
160	0.0881	–	−0.0306	0.0091	0.0366	−0.415	0.0105
170	0.0851	–	−0.0122	0.0025	–	−0.220	0.0057
						Mean S.E.	0.0127

From regression analyses it was concluded that the measured data were best fitted to the following three equations:

$$C_X = A_0 + A_1\frac{2A_L}{L^2} + A_2\frac{2A_T}{B^2} + A_3\frac{L}{B} + A_4\frac{S}{L} + A_5\frac{C}{L} + A_6 M$$

$$C_Y = -\left(B_0 + B_1\frac{2A_L}{L^2} + B_2\frac{2A_T}{B^2} + B_3\frac{L}{B} + B_4\frac{S}{L} + B_5\frac{C}{L} + B_6\frac{A_{SS}}{A_L}\right)$$

$$C_N = -\left(C_0 + C_1\frac{2A_L}{L^2} + C_2\frac{2A_T}{B^2} + C_3\frac{L}{B} + C_4\frac{S}{L} + C_5\frac{C}{L}\right)$$

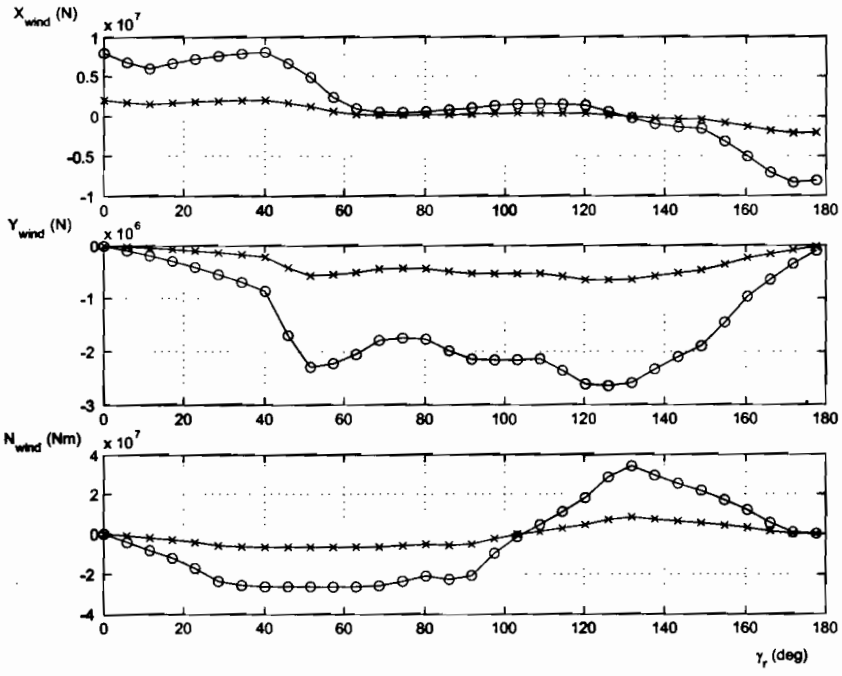


Figure 4.2: Wind forces and moment for $L = 100$, $B = 20$, $A_L = 400$, $A_T = 1000$, $A_{SS} = 100$, $S = 50$, $C = 20$ and $M = 1$ using the formulas of Isherwood (1972). Wind speed $V_r = 20$ m/s (cross) and $V_r = 40$ m/s (circels).

where A_i and B_i $(i = 0, \ldots, 6)$ and C_j $(j = 0, \ldots, 5)$ are tabulated in Table 4.1, together with the *residual standard errors* (S.E.). The signs of C_Y and C_N have been corrected to match the definition of γ_r in Figure 4.1.

Matlab:
The wind forces and moment (4.10)–(4.12) are plotted in Figure 4.2 using the GNC toolbox example file `ExWindForce.m`. The data sets of Isherwood (1972) are programmed in the Matlab function `windcoeff.m`:

```
[w_wind,cx,cy,cn] = windcoef(gamma_r,V_r,L,B,A_L,A_T,A_SS,S,C,M)
```

where the wind coefficients `cx`, `cy`, and `cn` are optional outputs.

4.1.3 Wind Resistance of Very Large Crude Carriers (OCIMF 1977)

Wind loads on very large crude carriers (VLCCs) can be computed by applying the following formulas (OCIMF 1977):

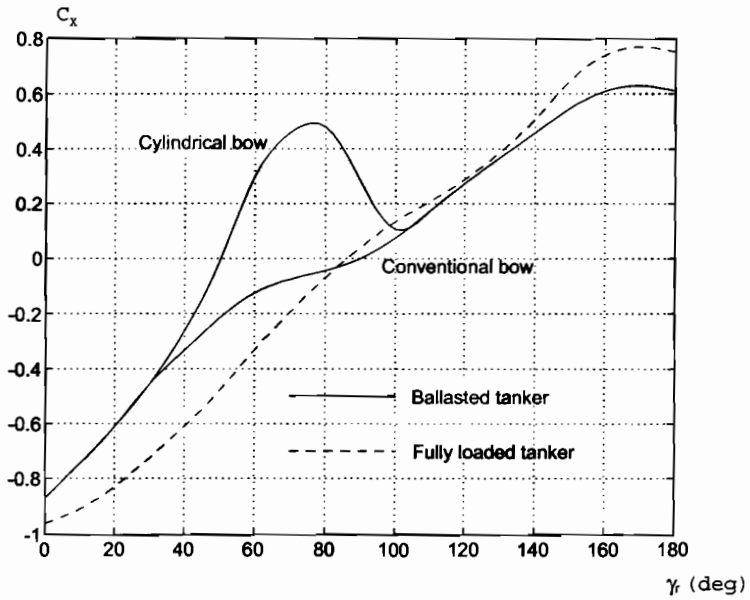


Figure 4.3: Longitudinal wind force coefficient C_X as a function of γ_r (OCIMF 1977).

$$X_{\text{wind}} = \frac{1}{7.6} C_X(\gamma_r) \rho_a V_r^2 A_T \quad \text{(N)} \tag{4.13}$$

$$Y_{\text{wind}} = \frac{1}{7.6} C_Y(\gamma_r) \rho_a V_r^2 A_L \quad \text{(N)} \tag{4.14}$$

$$N_{\text{wind}} = \frac{1}{7.6} C_N(\gamma_r) \rho_a V_r^2 A_L L \quad \text{(N m)} \tag{4.15}$$

Note that (4.13)–(4.15) only differs from (4.10)–(4.12) in the $1/2 \leftrightarrow 1/7.6$. This coefficient could easily be taken into C_X, C_Y, and C_N so that the only these coefficients differ for the different vessels.

Equations (4.13)–(4.15) are valid for vessels in the 150.000 to 500.000 (dwt) class. The non-dimensional force and moment coefficients C_X, C_Y, and C_N are given as a function of γ_r as shown in Figures 4.3–4.5 while ρ_a (kg/m^3) is the density of air. The constant 7.6 is a conversion factor. For ships that are symmetrical with respect to the xz- and yz-planes, the functions (4.13)–(4.15) can be approximated by:

$$C_X(\gamma_r) \approx c_x \cos\gamma_r, \quad C_Y(\gamma_r) \approx c_y \sin\gamma_r, \quad C_N(\gamma_r) \approx c_n \sin(2\gamma_r) \tag{4.16}$$

which are convenient formulas for computer simulations. Figures 4.3–4.5 indicate that $c_x \in \{-1.0, -0.8\}$, $c_y \in \{-1.0, -0.7\}$, and $c_n \in \{-0.2, -0.05\}$. However, the figures also indicate that these approximations should be used with care.

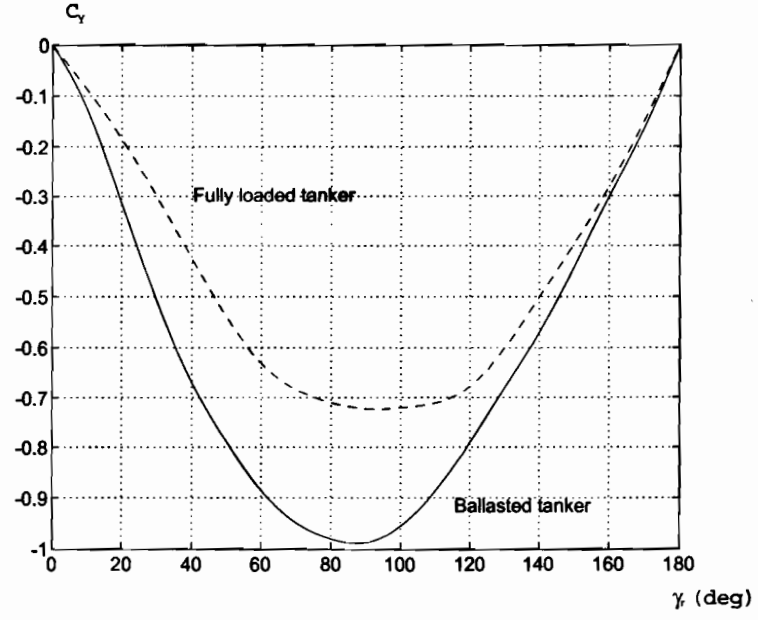


Figure 4.4: Lateral wind force coefficient C_Y as a function of γ_r (OCIMF 1977).

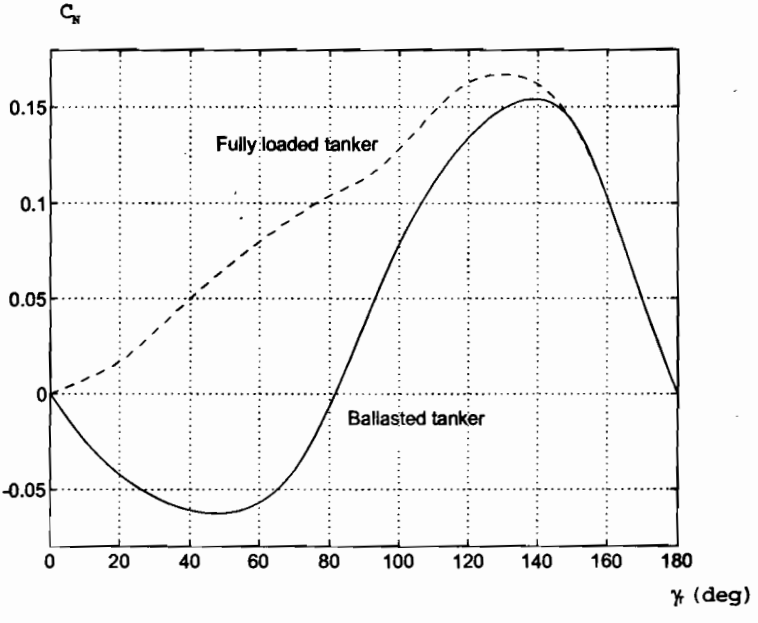


Figure 4.5: Wind moment coefficient C_N in yaw as a function of γ_r (OCIMF 1977).

4.1.4 Wind Resistance of Large Tankers and Medium Sized Ships

For wind resistance on large tankers in the 100.000 to 500.00 (dwt) class the reader is advised to consult Van Berlekom *et al.* (1974). Medium sized ships of the order 600 to 50.000 (dwt) is discussed by Wagner (1967).

A detailed analysis of wind resistance using semi-empirical loading functions is given by Blendermann (1986). The data sets for seven ships are included in the report.

4.1.5 Wind Resistance of Moored Ships and Floating Structures

Wind loads on moored ships are discussed by De Kat and Wichers (1991) while an excellent reference for huge pontoon type floating structures is Kitamura *et al.* (1997).

4.2 Models for Wind Generated Waves

The process of wave generation due to wind starts with small wavelets appearing on the water surface. This increases the drag force, which in turn allows short waves to grow. These short waves continue to grow until they finally break and their energy is dissipated. It is observed that a *developing sea,* or storm, starts with high frequencies creating a spectrum with a peak at a relative high frequency. A storm which has lasted for a long time is said to create a *fully developed sea.* After the wind has stopped, a low frequency decaying sea or swell is formed. These long waves form a wave spectrum with a low peak frequency.

If the swell from one storm interacts with the waves from another storm, a wave spectrum with two peak frequencies may be observed. In addition, tidal waves will generate a peak at a low frequency. Hence, the resulting wave spectrum might be quite complicated in cases where the weather changes rapidly; see Figure 4.6.

4.2.1 Nonlinear Models of Wave Spectra

The state-of-the-art wave spectra will now be presented. These models are used to derive linear approximations and transfer functions for computer simulations, autopilot wave filtering, and state reconstruction; see Sections 6.1–6.2.

Neumann Spectrum

The earliest spectral formulation is due to Neumann (1952) who proposed the *one-parameter* spectrum:

$$S(\omega) = C\omega^{-6} \exp\left(-2g^2\omega^{-2}V^{-2}\right), \quad (\mathrm{m^2\,s}) \tag{4.17}$$

where $S(\omega)$ is the wave elevation power spectral density function, C is an empirical constant, V is the wind speed, and g is the acceleration of gravity. Six years later Phillips (1958) showed that the high frequency part of the wave spectrum reached the asymptotic limit:

$$\lim_{\omega \gg 1} S(\omega) = \alpha\, g^2\, \omega^{-5} \tag{4.18}$$

where α is a positive constant. This limiting function of Phillips is still used as basis for most spectral formulations.

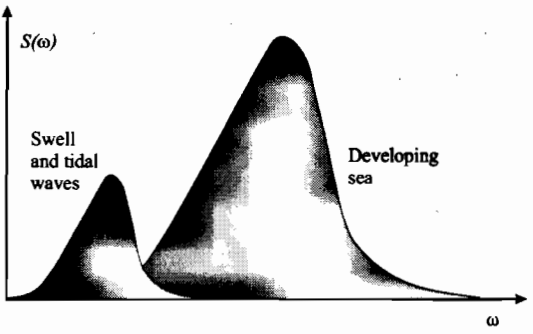


Figure 4.6: Two peaked wave spectrum.

Bretschneider Spectrum

The spectrum of Neumann was further extended to a two-parameter spectrum by Bretschneider (1959):

$$S(\omega) = 1.25\frac{\omega_0^4 H_s^2}{4}\omega^{-5}\exp\left(-1.25\left(\omega_0/\omega\right)^4\right), \quad (\mathrm{m^2\,s}) \tag{4.19}$$

where ω_0 is the *modal* or *peak frequency* of the spectrum and H_s is the *significant wave height* (mean of the one-third highest waves). This spectrum was developed for the North Atlantic, for unidirectional seas, infinite depth, no swell, and unlimited fetch. The significant wave height H_s is used to classify the type of sea in terms of sea state codes $0, 1, ..., 9$ as shown in Table 4.2.

Pierson–Moskowitz Spectrum

Pierson and Moskowitz (1963) have developed a two parameter wave spectral formulation for fully developed wind-generated seas from analyses of wave spectra in the North Atlantic Ocean:

$$S(\omega) = A\omega^{-5}\,\exp\left(-B\omega^{-4}\right), \quad (\mathrm{m^2\,s}) \tag{4.20}$$

and it is commonly known as the *PM-spectrum* (Pierson–Moskowitz spectrum). The PM-spectrum is used as basis for several spectral formulations but with different A and B values. In its original formulation, the PM-spectrum is only a one-parameter spectrum since only B changes with the sea state. The parameters are:

$$A = 8.1\cdot 10^{-3}\,g^2 = \text{constant} \tag{4.21}$$

$$B = 0.74\left(\frac{g}{V_{19.4}}\right)^4 = \frac{3.11}{H_s^2} \tag{4.22}$$

where $V_{19.4}$ is the wind speed at a height of 19.4 (m) over the sea surface.

Table 4.2: Definition of Sea State (SS) codes (Price and Bishop, 1974). Notice that the percentage probability for SS codes 0, 1, and 2 is summarized.

			Percentage probability		
Sea state code	Description of sea	Wave height observed (m)	World wide	North Atlantic	Northern North Atlantic
0	Calm (glassy)	0			
1	Calm (rippled)	0-0.1	11.2486	8.3103	6.0616
2	Smooth (wavelets)	0.1-0.5			
3	Slight	0.5-1.25	31.6851	28.1996	21.5683
4	Moderate	1.25-2.5	40.1944	42.0273	40.9915
5	Rough	2.5-4.0	12.8005	15.4435	21.2383
6	Very rough	4.0-6.0	3.0253	4.2938	7.0101
7	High	6.0-9.0	0.9263	1.4968	2.6931
8	Very high	9.0-14.0	0.1190	0.2263	0.4346
9	Phenomenal	Over 14.0	0.0009	0.0016	0.0035

Matlab:
The Pierson–Moskowitz spectrum is implemented in the GNC toolbox as:

```
[w,S,wo,V] = pierson(Hs,wmax,N)
```

The relationship between $V_{19.4}$ and H_s in (4.22) is based on the assumption that the waves can be represented by Gaussian random processes and that $S(\omega)$ is narrow-banded. From (4.22) it is seen that:

$$H_s = \frac{2.06}{g^2} V_{19.4}^2 \tag{4.23}$$

implying that the significant wave height is proportional to the square of the wind speed. This is shown in Figure 4.7 where the *sea state codes* and *Beaufort numbers* are plotted against each other, see Tables 4.2–4.3.

The *modal frequency* (peak frequency) ω_0 for the PM-spectrum is found by requiring that:

$$\left(\frac{dS(\omega)}{d\omega} \right)_{\omega=\omega_0} = 0 \tag{4.24}$$

Solving for ω_0 in (4.20), yields:

$$\omega_0 = \sqrt[4]{\frac{4B}{5}} \quad \Longrightarrow \quad T_0 = 2\pi \sqrt[4]{\frac{5}{4B}} \tag{4.25}$$

where T_0 is the *modal period.* Consequently, the maximum value of $S(\omega)$ is:

$$S_{\max} = S(\omega_0) = \frac{5A}{4B\omega_0} \exp\left(-5/4\right) \tag{4.26}$$

Table 4.3: Definition of Beaufort numbers (Price and Bishop, 1974).

Beaufort number	Description of wind	Wind speed (knots)
0	Calm	0-1
1	Light air	2-3
2	Light breeze	4-7
3	Gentle breeze	8-11
4	Moderate breeze	12-16
5	Fresh breeze	17-21
6	Strong breeze	22-27
7	Moderate gale	28-33
8	Fresh gale	34-40
9	Strong gale	41-48
10	Whole gale	49-56
11	Storm	57-65
12	Hurricane	More than 65

Wave Spectrum Moments

The different wave spectra can be classified by means of *wave spectrum moments*:

$$m_k = \int_0^\infty \omega^k \, S(\omega) \, d\omega, \qquad (k = 0, \ldots, N) \tag{4.27}$$

For $k = 0$, this yields:

$$m_0 = \int_0^\infty S(\omega) d\omega = \frac{A}{4B} \tag{4.28}$$

which simply states that the instantaneous wave elevation is Gaussian-distributed with zero mean and variance $\sigma^2 = A/4B$. Consequently,

$$\sigma = \sqrt{m_0} \tag{4.29}$$

can be interpreted as the RMS-value of the spectrum.

Under the assumption that the wave height is Rayleigh distributed it can be shown that (Price and Bishop 1974):

$$H_s = 4\sigma = 4\sqrt{m_0} \tag{4.30}$$

The corresponding wave moments for $k = 1$ and $k = 2$, become:

$$m_1 = 0.306 \frac{A}{B^{3/4}}, \quad m_2 = \frac{\sqrt{\pi}}{4} \frac{A}{\sqrt{B}} \tag{4.31}$$

and for the PM-spectrum, the *average wave period* is defined as:

$$T_1 = 2\pi \frac{m_0}{m_1} \tag{4.32}$$

while the *average zero-crossings period* is defined as:

$$T_z = 2\pi \sqrt{\frac{m_0}{m_2}} \tag{4.33}$$

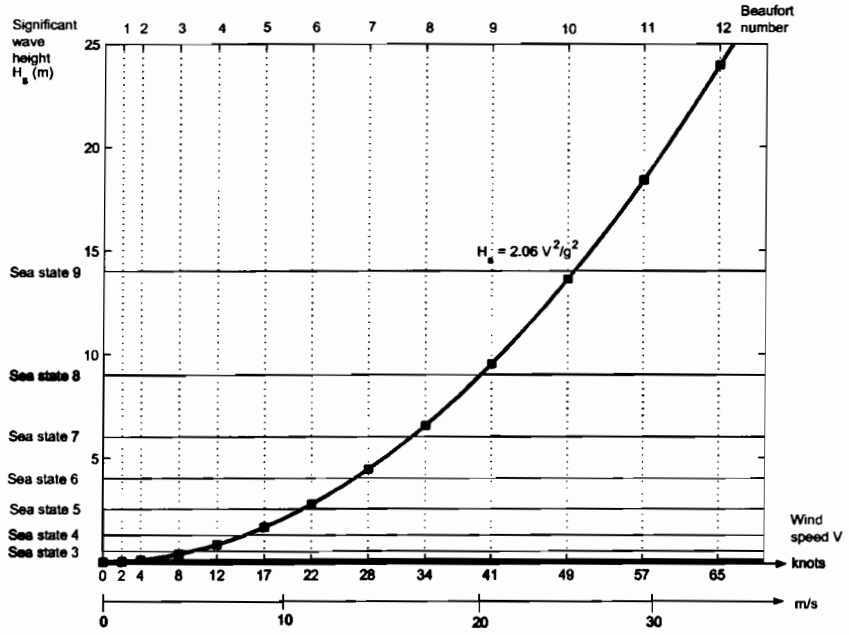


Figure 4.7: Plot showing the relationship between significant wave height, wind speed, Beaufort numbers, and sea state codes.

Modified Pierson–Moskowitz (MPM) Spectrum

For prediction of responses of marine vehicles and offshore structures in open sea, the International Ship and Offshore Structures Congress (2nd ISSC 1964), and the International Towing Tank Conference, 12th ITTC (1969) and 15th ITTC (1978) have recommended the use of a modified version of the PM-spectrum where:

$$A = \frac{4\pi^3 H_s^2}{T_z^4}, \qquad B = \frac{16\pi^3}{T_z^4} \tag{4.34}$$

This representation of the PM-spectrum has two parameters H_s and T_z, or alternatively T_0 and T_1 given by:

$$T_z = 0.710 T_0 = 0.921 T_1 \tag{4.35}$$

can be used.

Matlab:
The modified PM-spectrum is implemented in the GNC toolbox as:

```
[w,S,Tz] = mpierson(Hs,wo,wmax,N)
```

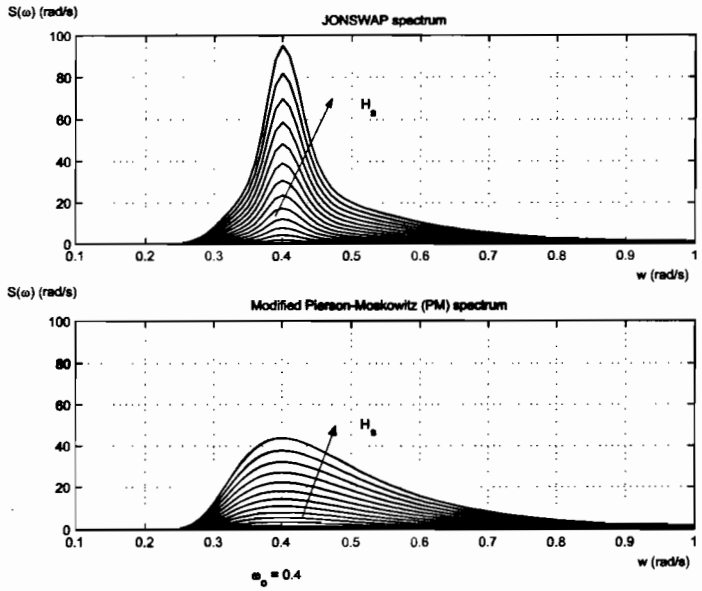


Figure 4.8: Plot showing the JONSWAP and Modified Pierson-Moskowitz spectra for $\omega_0 = 0.4$ rad/s and $H_s = 3, 4, ..., 14$ m.

The modified PM-spectrum should only be used for a fully developed sea with large (infinite) depth, no swell, and unlimited fetch. For non-fully developed seas the *JONSWAP* or *Torsethaugen* spectra are recommended.

JONSWAP Spectrum

In 1968 and 1969 an extensive measurement program was carried out in the North Sea, between the island Sylt in Germany and Iceland. The measurement program is known as the *Joint North Sea Wave Project* (JONSWAP) and the results from these investigations have been adopted as an ITTC standard by the 17th ITTC (1984). Since the JONSWAP spectrum is used to describe *non-fully developed seas*, the spectral density function will be more peaked than those representing fully developed spectra. The proposed spectral formulation is representative for wind-generated waves under the assumption of finite water depth and limited fetch. The spectral density function is written:

$$S(\omega) = 155\,\frac{H_s^2}{T_1^4}\omega^{-5}\exp\left(\frac{-944}{T_1^4}\omega^{-4}\right)\gamma^Y, \quad (\mathrm{m^2s}) \tag{4.36}$$

where Hasselmann et al. (1973) suggest that $\gamma = 3.3$ and:

$$Y = \exp\left[-\left(\frac{0.191\omega T_1 - 1}{\sqrt{2}\sigma}\right)^2\right] \tag{4.37}$$

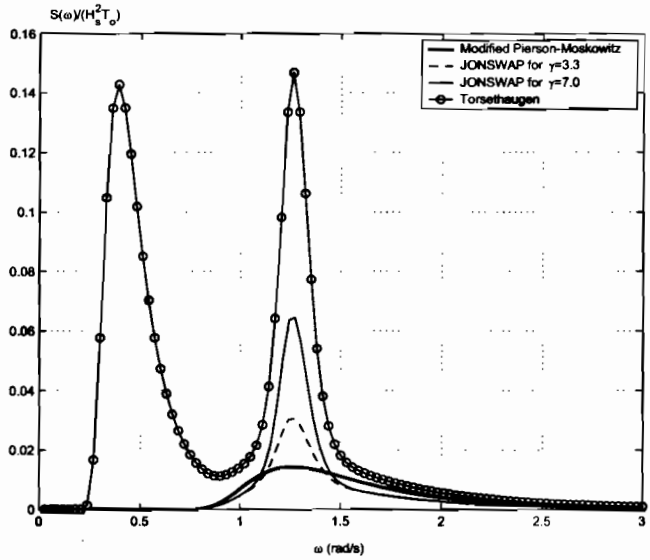


Figure 4.9: Comparison of different spectra for significant wave height $H_s = 10$ m and wave peak period $T_0 = 5$ s.

where

$$\sigma = \begin{cases} 0.07 & for \quad \omega \leq 5.24/T_1 \\ 0.09 & for \quad \omega > 5.24/T_1 \end{cases} \tag{4.38}$$

Alternative formulations can be derived in terms of the characteristic periods like T_0 and T_z by using:

$$T_1 = 0.834\, T_0 = 1.073\, T_z \tag{4.39}$$

Matlab:
The JONSWAP spectrum is included in the GNC toolbox as:

```
[w,S,Tz] = jonswap(Hs,wo,wmax,gamma,N)
```

Torsethaugen Spectrum

The *Torsethaugen spectrum* is an empirical, two peaked spectrum, which includes the effect of swell (low frequency peak) and newly developed waves (high frequency peak). The spectrum was developed for Norsk Hydro (Torsethaugen 1996), and standardized under the Norsok Standard (1999). The spectrum is developed using curve fitting of experimental data from the North Sea.

Matlab:
In Figure 4.10 the Torsethaugen spectrum is plotted for different peak frequencies and significant wave heights using the GNC toolbox function:

```
[w,S] = torset(Hs,wo,wmax,N)
```

If the peak frequency ω_0 is chosen to be less than approximately 0.6 (rad/s) the Torsethaugen spectrum reduces to a one peak spectrum where swell dominates. For peak frequencies $\omega_0 >$ 0.6 (rad/s) the two characteristic peaks shown in Figure 4.10 clearly appear. This is due to the fact that developing waves are high frequent compared to swell. This combined effect is very common in the North Sea, and it makes DP and autopilot design a challenging task in terms of wave filtering.

Matlab:
The different wave spectra in this section when plotted for same wave height and peak frequency are shown in Figure 4.9 using the wave demo option in the GNC toolbox script:

```
gncdemo
```

4.2.2 Linear Wave Response Models

The wave-induced forces and moments on a marine vessel in closed loop can be simulated by assuming a linear wave response model. If accuracy of the vessel motion is critical, a more detailed model for the wave loads should be applied.

A linear approximation of the spectral density function $S(\omega)$ can be found by writing the output $y(s)$ from the wave model as a linear filter:

$$y(s) = h(s)\, w(s) \tag{4.40}$$

where $w(s)$ is a zero-mean Gaussian white noise process with unity power across the spectrum:

$$P_{ww}(\omega) = 1.0 \tag{4.41}$$

and $h(s)$ is a transfer function to be determined. Hence, the power spectral density (PSD) function for $y(s)$ can be computed as:

$$P_{yy}(\omega) = |h(j\omega)|^2\, P_{ww}(\omega) = |h(j\omega)|^2 \tag{4.42}$$

The ultimate goal is to design an approximating $P_{yy}(\omega)$ to $S(\omega)$, for instance by means of nonlinear regression, such that $P_{yy}(\omega)$ reflects the energy distribution of $S(\omega)$ in the relevant frequency range. Linear approximations well suited for this purpose are discussed below.

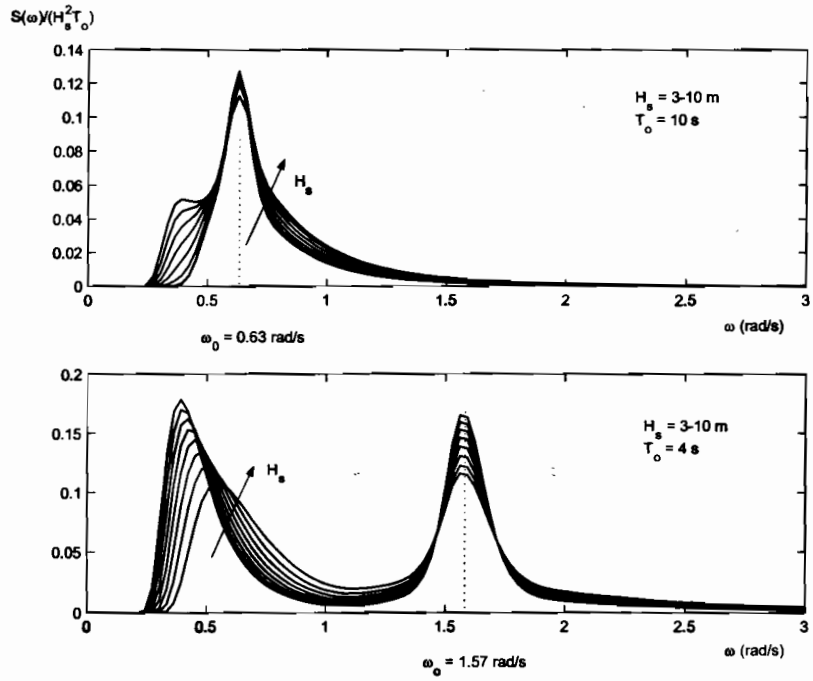


Figure 4.10: Torsethaugen spectrum: upper plot shows only one peak at $\omega_0 = 0.63$ rad/s representing swell and developing sea while the lower plot shows low-frequency swell and newly developing sea with peak frequency $\omega_0 = 1.57$ rad/s. Courtesy to Norsk Hydro.

2nd-Order Wave Response Transfer Function Approximation

Linear wave response approximations are usually preferred by ship control systems engineers, owing to their simplicity and applicability. The first applications were reported by Balchen *et al.* (1976) who proposed modeling the wave-frequency motion of a dynamically positioned ship in surge, sway, and yaw by three harmonic oscillators without damping. Later Sælid *et al.* (1983) introduced a damping term λ in the wave model to better fit the shape of the PM-spectrum. This model is written:

$$h(s) = \frac{K_w s}{s^2 + 2\lambda\omega_0 s + \omega_0^2} \tag{4.43}$$

and it is convenient to define the gain constant according to:

$$K_w = 2\lambda\omega_0\sigma \tag{4.44}$$

where σ is a constant describing the wave intensity, λ is a damping coefficient, and ω_0 is the dominating wave frequency. Consequently, substituting $s = j\omega$ yields the frequency

response:

$$h(j\omega) = \frac{j\,2(\lambda\omega_0\sigma)\omega}{(\omega_0^2 - \omega^2) + j\,2\lambda\omega_0\omega} \quad \Longrightarrow \quad |h(j\omega)| = \frac{2(\lambda\omega_0\sigma)\omega}{\sqrt{(\omega_0^2 - \omega^2)^2 + 4(\lambda\omega_0\omega)^2}} \tag{4.45}$$

From (4.42) we recall that:

$$P_{yy}(\omega) = |h(jw)|^2 = \frac{4(\lambda\omega_0\sigma)^2\omega^2}{(\omega_0^2 - \omega^2)^2 + 4(\lambda\omega_0\omega)^2} \tag{4.46}$$

Determination of σ and λ

Since the maximum value of $P_{yy}(\omega)$ and $S(\omega)$ are obtained for $\omega = \omega_0$, we have:

$$P_{yy}(\omega_0) = S(\omega_0) \tag{4.47}$$

$$\Updownarrow$$

$$\sigma^2 = \max_{0<\omega<\infty} S(\omega) \tag{4.48}$$

For the PM-spectrum (4.20) this implies:

$$\sigma = \sqrt{\frac{A}{\omega_0^5}\exp\left(-\frac{B}{\omega_0^4}\right)} \tag{4.49}$$

while the term $\gamma^{Y(\omega_0)}$ must be included for the JONSWAP spectrum. The damping ratio λ can be computed by requiring that the energy, that is the areas under $P_{yy}(\omega)$ and $S(\omega)$ of the spectra, be equal.

An alternative approach is to use nonlinear least-squares (NLS) to compute λ such that $P_{yy}(\omega)$ fits $S(\omega)$ in a least-squares sense; see Figure 4.12. This is demonstrated in Example 4.1 using the Matlab^{TM} optimization toolbox.

Example 4.1 (Nonlinear Least-Squares Optimization of Linear Spectra)
Consider the Matlab script ExLinspec.m *for computation of λ. The output of the nonlinear optimization process gives the following λ-values for the modified PM and JONSWAP spectra:*

	ω_0= 0.5	ω_0= 0.8	ω_0= 1.1	ω_0= 1.4	*Recommended values*
λ *(MPM)*	0.2575	0.2577	0.2590	0.2608	0.26
λ *(JONSWAP)*	0.1066	0.1034	0.1025	0.1021	0.10

The λ-value for both these spectra are independent of the wave height H_s. For the Torsethaugen spectrum the λ-values vary with both H_s and ω_0 as shown in Figure 4.11. The results of the curve fitting procedure for the three different spectra are shown in Figure 4.12. Since the Torsethaugen spectrum is a two peaked spectrum a second linear spectrum should be added to fit the swell peak at low frequencies.

Matlab:
```
function Pyy = Slin(lambda,w)
% Pyy = Slin(lambda,w) 2nd-order linear PSD function
% w = wave spectrum frequency (rad/s)
% lambda = relative damping factor
global sigma wo
Pyy = 4*(lambda*wo*sigma)^2*w.^2./((wo^2-w.^2).^2 +...
      4*(lambda*wo.*w).^2)
```

Matlab:
```
% Matlab script for plotting of nonlinear least-squares fit,
% see ExLinspec.m
global sigma wo

wo = 1.2; To = 2*pi/wo; Hs = 10; wmax = 3;

% Modified PM
subplot(311)
[w,S] = mpierson(Hs,wo,wmax); sigma = sqrt(max(S));
lambda = lsqcurvefit('Slin',0.1,w,S)
hold on; plot(w,Slin(lambda,w)/(Hs^2*To)); hold off;
legend('Modified PM spectrum','linear approximation')

% JONSWAP
subplot(312)
[w,S] = jonswap(Hs,wo,wmax); sigma = sqrt(max(S));
lambda = lsqcurvefit('Slin',0.1,w,S)
hold on; plot(w,Slin(lambda,w)/(Hs^2*To)); hold off;
legend('Modified PM spectrum','linear approximation')

% Torsethaugen
subplot(313)
[w,S] = torset(Hs,wo,wmax); sigma = sqrt(max(S));
lambda = lsqcurvefit('Slin',0.1,w,S)
hold on; plot(w,Slin(lambda,w)/(Hs^2*To)); hold off;
legend('Modified PM spectrum','linear approximation')
```

State-Space Representations of Linear Wave Spectra

A linear state-space model can be obtained from (4.43) by transforming this expression to the time-domain by defining $\dot{x}_{w1} = x_{w2}$ and $x_{w2} = y_w$ as state variables. This implies that the state-space model can be written:

$$\dot{\mathbf{x}}_w = \mathbf{A}_w \mathbf{x}_w + \mathbf{e}_w w_w \tag{4.50}$$

$$y_w = \mathbf{c}_w^\top \mathbf{x}_w \tag{4.51}$$

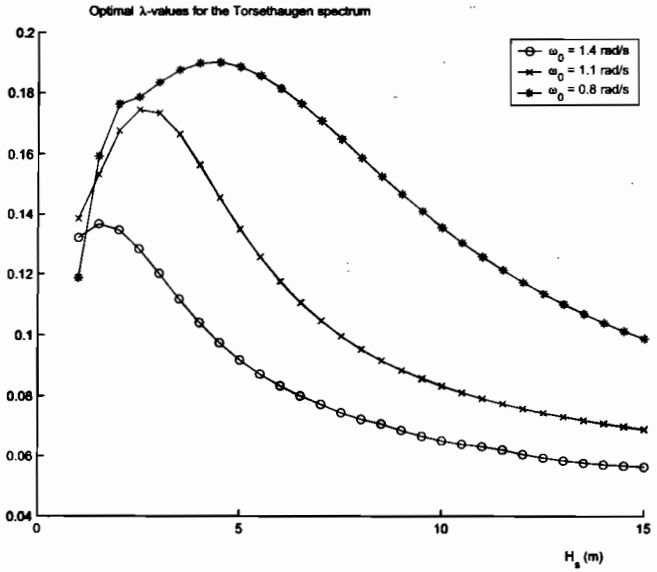


Figure 4.11: Least-squares optimal λ-values for the Torsethaugen spectrum for varying H_s and ω_0 when a linear spectrum is fitted to the high frequency peak of the spectrum.

where w_w is a zero-mean white noise process. Writing this expression in component form, yields:

$$\begin{bmatrix} \dot{x}_{w1} \\ \dot{x}_{w2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & -2\lambda\omega_0 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \end{bmatrix} + \begin{bmatrix} 0 \\ K_w \end{bmatrix} w_w \tag{4.52}$$

$$y_w = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \end{bmatrix} \tag{4.53}$$

Higher-Order Wave Response Transfer Function Approximations

An alternative wave transfer function based on five parameters has been proposed by Grimble *et al.* (1980*a*) and Fung and Grimble (1983). This model takes the form:

$$h(s) = \frac{K_w s^2}{s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4} \tag{4.54}$$

where a_i $(i = 1, \ldots, 4)$ are four parameters. Consequently, four differential equations are required to describe the wave model:

$$\begin{bmatrix} \dot{x}_{w1} \\ \dot{x}_{w2} \\ \dot{x}_{w3} \\ \dot{x}_{w4} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -a_4 & -a_3 & -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \\ x_{w3} \\ x_{w4} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ K_w \end{bmatrix} w_w \tag{4.55}$$

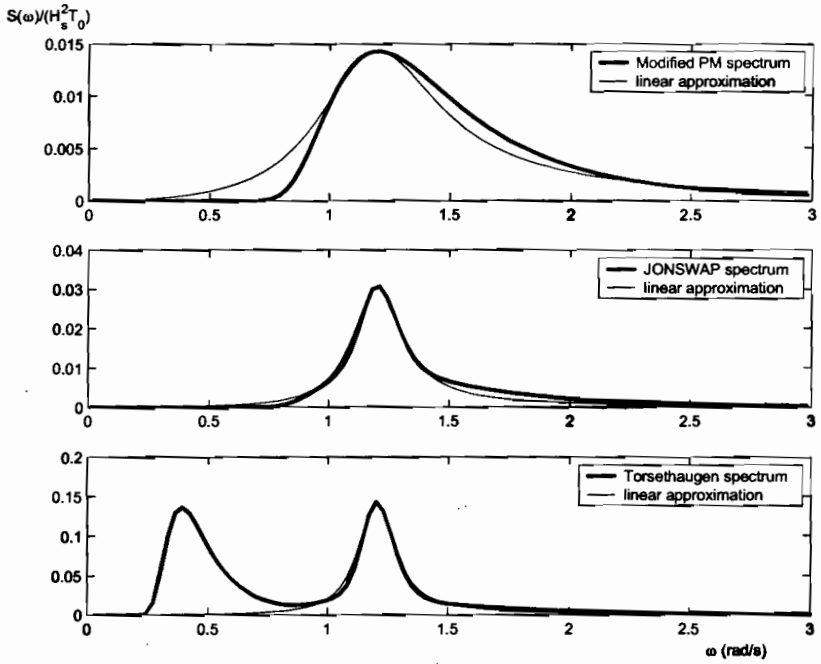


Figure 4.12: Nonlinear least-squares fit of a linear spectrum to the PM, JONSWAP, and Torsethaugen spectra. Only one peak is approximated for the Torsethaugen spectrum.

$$y_w = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_{w1} \\ x_{w2} \\ x_{w3} \\ x_{w4} \end{bmatrix} \tag{4.56}$$

The number of parameters can be reduced by assuming that the denominator can be factorized according to:

$$h(s) = \frac{K_w s^2}{(s^2 + 2\lambda\omega_0 s + \omega_0^2)^2} \tag{4.57}$$

Triantafyllou *et al.* (1983) have shown by applying a rational approximation to the Bretschneider spectrum that a satisfactory approximation of the high-frequency ship motion can be obtained by using the transfer function:

$$h(s) = \frac{K_w s^2}{(s^2 + 2\lambda\omega_0 s + \omega_0^2)^3} \tag{4.58}$$

which only has three unknown parameters λ, ω_0, and K_w. The advantage of the higher order models to the simple 2nd-order system (4.43) is that they will represent a more precise approximation to the wave spectrum response through a nonlinear least-squares curve fitting procedure. The disadvantage, of course, is higher model complexity and perhaps more parameters to determine.

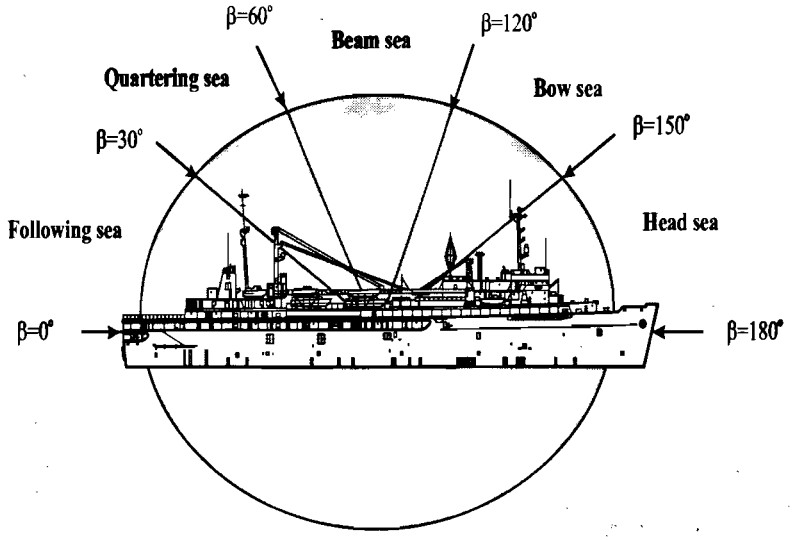


Figure 4.13: Definition of encounter angle β.

4.2.3 Frequency of Encounter

For a ship moving with forward speed U, the peak frequency of the spectrum ω_0 will be modified according to:

$$\omega_e(U, \omega_0, \beta) = \left| \omega_0 - \frac{\omega_0^2}{g} U \cos\beta \right| \tag{4.59}$$

where:

ω_e - encounter frequency (rad/s)
ω_0 - wave frequency (rad/s)
g - acceleration of gravity (m/s^2)
U - total speed of ship (m/s)
β - the angle between the heading and the direction of the wave (rad)

The definition of the encounter angle β is shown in Figure 4.13.

This suggests that the peak frequency of the wave spectrum for a heading controlled ship moving at speed $U > 0$ should be modified to incorporate the frequency of encounter. In the linear case, the transfer function (4.43) must be redefined to:

$$h(s) = \frac{K_w s}{s^2 + 2\lambda\omega_e s + \omega_e^2} \tag{4.60}$$

However, it should be noted that the wave frequency of a dynamically positioned ship can be perfectly described by $\omega_e = \omega_0$, since U is close to or equal to zero.

Matlab:
The GNC toolbox function:

```
w_e = encounter(w_o,U,beta)
```

computes the frequency of encounter.

4.2.4 Wave Forces and Moments

A marine control system can be simulated under influence of wave-induced disturbances by separating the *1st-order* and *2nd-order* effects:

- **1st-order effects:** wave frequency (WF) motion
- **2nd-order effects:** wave drift forces

For a surface vessel in 3 DOF the wave forces and moments, see (4.2), are easiest simulated by defining:

$$\mathbf{w}_{\text{waves}} = [X_{\text{waves}}, Y_{\text{waves}}, N_{\text{waves}}]^\top \tag{4.61}$$

where X_{waves}, Y_{waves}, and N_{waves} are generated by using linear theory. The wave model (4.60) suggests that:

$$X_{\text{waves}} = \frac{K_{w1}s}{s^2 + 2\lambda_1\omega_{e1}s + \omega_{e1}^2}w_1 + d_1 \tag{4.62}$$

$$Y_{\text{waves}} = \frac{K_{w2}s}{s^2 + 2\lambda_2\omega_{e2}s + \omega_{e2}^2}w_2 + d_2 \tag{4.63}$$

$$N_{\text{waves}} = \frac{K_{w3}s}{s^2 + 2\lambda_3\omega_{e3}s + \omega_{e3}^2}w_3 + d_3 \tag{4.64}$$

Here w_i $(i = 1, \ldots, 3)$ are Gaussian white noise processes. The amplitudes of X_{waves}, Y_{waves}, and N_{waves} are adjusted by choosing the constants K_{wi} $(i = 1, \ldots, 3)$ while the the spectra are parametrized in terms of the pairs λ_i and ω_{ei} $(i = 1, \ldots, 3)$. These values should be chosen to represent the true physical behavior.

The wave drift forces d_1 $(i = 1, \ldots, 3)$ are usually modelled as slowly-varying bias terms (*Wiener processes*):

$$\dot{d}_1 = w_4 \tag{4.65}$$

$$\dot{d}_2 = w_5 \tag{4.66}$$

$$\dot{d}_3 = w_6 \tag{4.67}$$

where w_i $(i = 4, \ldots, 6)$ are Gaussian white noise processes. These equations should be modified by using saturating elements to prevent d_i from exceeding a predescribed maximum physical limit, that is $|d_i| \leq d_{i,\max}$.

4.3 Models for Ocean Currents

Ocean currents are horizontal and vertical circulation systems of ocean waters produced by gravity, wind friction, and water density variation in different parts of the ocean. Besides *wind-generated currents*, the heat exchange at the sea surface together with salinity changes, develop an additional sea current component, usually referred to as *thermohaline currents*. A world map showing the most major ocean surface currents is found in Defant (1961).

The oceans are conveniently divided into two water spheres, the cold and warm water sphere. Since the Earth is rotating, the Coriolis force will try to turn the major currents to the East in the northern hemisphere and West in the southern hemisphere. Finally, the major ocean circulations will also have a tidal component arising from planetary interactions like gravity. In coastal regions and fjords, tidal components can reach very high speeds, in fact speeds of 2 to 3 (m/s) or more have been measured.

In order to simulate ocean currents and their effect on vessel motion, the following model will be applied:

$$\underbrace{\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta})}_{\text{rigid-body terms}} + \underbrace{\mathbf{M}_A\dot{\boldsymbol{\nu}} + \mathbf{C}_A(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r}_{\text{hydrodynamic terms}} = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{4.68}$$

where $\boldsymbol{\nu}_r = \boldsymbol{\nu} - \boldsymbol{\nu}_c$, and the current velocity vector is assumed to be slowly-varying, that is $\dot{\boldsymbol{\nu}}_c \approx \mathbf{0}$. Hence, the equations of motion becomes:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{C}_A(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{g}_o + \mathbf{w} \tag{4.69}$$

We will now turn our attention to models for $\boldsymbol{\nu}_c$.

Current Speed and Direction

The current speed is denoted by V_c while its direction relative to the moving vessel is conveniently expressed in terms of two angles: angle of attack α_c, and sideslip angle β_c as shown in Figure 2.8 in Section 2.4. For computer simulations the current velocity can be generated by using a 1st-order *Gauss–Markov Process*

$$\dot{V}_c + \mu V_c = w \tag{4.70}$$

where w is Gaussian white noise and $\mu \geq 0$ is a constant. If $\mu = 0$, this model reduces to a *random walk*, corresponding to time integration of *white noise*. A saturating element is usually used in the integration process to limit the current speed to:

$$V_{\min} \leq V_c(t) \leq V_{\max} \tag{4.71}$$

The direction of the current can be fixed by specifying constant values for α_c and β_c. Time-varying directions can easily be simulated by associating dynamics to α_c and β_c.

4.3.1 3D Irrotational Current Model

A 3D current model is obtained by transforming the current speed V_c from current axes to the b-frame by:

$$\begin{bmatrix} u_c^b \\ v_c^b \\ w_c^b \end{bmatrix} = \mathbf{R}_{y,\alpha_c}^\top \mathbf{R}_{z,-\beta_c}^\top \begin{bmatrix} V_c \\ 0 \\ 0 \end{bmatrix} \tag{4.72}$$

where the rotation matrices $\mathbf{R}_{y,\alpha_c}$ and $\mathbf{R}_{z,-\beta_c}$ are defined in Section 2.4. Assuming that the fluid is irrotational implies that:

$$\boldsymbol{\nu}_c = [u_c^b, v_c^b, w_c^b, 0, 0, 0]^\top \tag{4.73}$$

where u_c^b, v_c^b, and w_c^b are the b-frame current velocities. Expanding this expression yields:

$$u_c^b = V_c \cos\alpha_c \cos\beta_c \tag{4.74}$$
$$v_c^b = V_c \sin\beta_c \tag{4.75}$$
$$w_c^b = V_c \sin\alpha_c \cos\beta_c \tag{4.76}$$

4.3.2 2D Irrotational Current Model (Horizontal-Plane Model)

For the 2D case, the 3D equations (4.74)–(4.76) with $\alpha_c = 0$ reduces to:

$$u_c^b = V_c \cos\beta_c \tag{4.77}$$
$$v_c^b = V_c \sin\beta_c \tag{4.78}$$

since the component w_c^b is not used in the horizontal plane. Notice that:

$$V_c = \sqrt{(u_c^b)^2 + (v_c^b)^2} \tag{4.79}$$

Example 4.2 (Maneuvering Model)
Consider the maneuvering model in Section 3.5.2 which can be written in state-space form according to:

$$\begin{bmatrix} m_{11} & m_{12} & 0 \\ m_{21} & m_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} & 0 \\ d_{21} & d_{22} & 0 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} v - v_c^b \\ r \\ \psi \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix} \delta + \begin{bmatrix} Y_{\text{wind}} \\ N_{\text{wind}} \\ 0 \end{bmatrix} + \begin{bmatrix} Y_{\text{wave}} \\ N_{\text{wave}} \\ 0 \end{bmatrix} \tag{4.80}$$

where v is the sway velocity, r is the yaw rate, ψ is the yaw angle, δ is the rudder angle, and v_c^b is the transverse current velocity given by:

$$v_c^b = V_c \sin\beta_c \tag{4.81}$$

Augmenting the current model (4.70) to this model under assumption that β_c = constant, yields:

$$\begin{bmatrix} m_{11} & m_{12} & 0 & 0 \\ m_{21} & m_{22} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \\ \dot{V}_c \end{bmatrix} + \begin{bmatrix} d_{11} & d_{12} & 0 & -d_{11}\sin\beta_c \\ d_{21} & d_{22} & 0 & -d_{21}\sin\beta_c \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -\mu \end{bmatrix} \begin{bmatrix} v \\ r \\ \psi \\ V_c \end{bmatrix}$$

$$= \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \end{bmatrix} \delta + \begin{bmatrix} Y_{\text{wind}} \\ N_{\text{wind}} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} Y_{\text{wave}} \\ N_{\text{wave}} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} w$$

4.4 Exercises

Exercise 4.1 *Consider Example 4.2 and show how the wave models (4.63) and (4.64) corresponding to:*

$$w_{\text{wave}} = \begin{bmatrix} Y_{\text{wave}} \\ N_{\text{wave}} \\ 0 \end{bmatrix}$$

can be augmented to the state-space model in the example by introducing four new state variables for the WF motion and two new state variable for wave drift. The matrices in the model should be expanded to include the WF model parameters.

Exercise 4.2 *Given the spectral density function $S(\omega) = A\omega^{-5} \exp\left(-B\omega^{-4}\right)$, compute the maximum value $S_{\max}(\omega_o)$ of $S(\omega)$ by requiring that:*

$$\left(\frac{dS(\omega)}{d\omega}\right)_{\omega=\omega_0} = 0$$

Is

$$\left(\frac{d^2S(\omega)}{d\omega^2}\right)_{\omega=\omega_0}$$

positive or negative? (Explain why)

Exercise 4.3 *Plot the frequency of encounter:*

$$\omega_e(U, \omega_0, \beta) = \left|\omega_0 - \frac{\omega_0^2}{g} U \cos\beta\right|$$

as a function of ω_0 for different β-values when $U = 10$ (m/s). What the is most critical direction β for an autopilot controlled ship in transit? (Explain why)

Exercise 4.4 *Use the Matlab GNC toolbox command:*

w_wind = windcoef(gamma_r,V_r,L,B,A_L,A_T,A_SS,S,C,M)

to plot the wind forces and moments as functions of γ_r for $V_r = 10$ (m/s). The wind parameters can be chosen rather arbitrarily or by defining your own ship. What the is most critical direction γ_r for an autopilot controlled ship in transit? (Explain why). Will the conclusion be the same if you consider different values of V_r as well?

Exercise 4.5 *Write a Matlab m-file (or Simulink block) that generates a slowly-varying ocean current with speed limited by* $0.5 \leq V_c(t) \leq 1.0$ *(m/s). Plot the b-frame components:*

$$\begin{aligned} u_c^b &= V_c \cos\beta_c \\ v_c^b &= V_c \sin\beta_c \end{aligned}$$

as a function of time t *for different values of* $\beta_c \in [0, 360^o]$.

Exercise 4.6 *Implement the linear wave model (4.52)–(4.53) in Simulink and plot the wave amplitude* $y(t)$ *as a function of time* t *for different values of* ω_0 *and* K_w *when* $\lambda = 0.1$. *Explain how the parameters affect the time response of the wave amplitude.*

Part II

Guidance, Navigation and Control Fundamentals

Chapter 5

Maritime Guidance Systems

This chapter describes methods for the design of vessel *guidance systems*. Guidance systems for marine vessels are usually used to generate a reference trajectory for time-varying *trajectory tracking* or time-invariant *maneuvering* or *path* control. *Set-point regulation* is a special case where the desired velocity, position, and attitude are chosen to be constant.

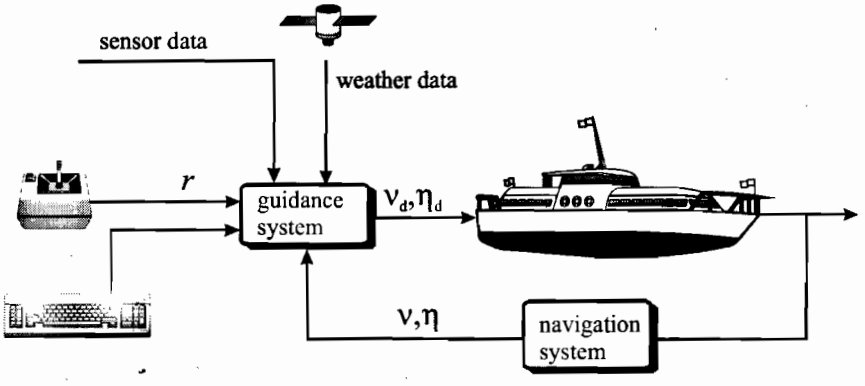


Figure 5.1: Guidance system.

As shown in Figure 5.1, the guidance system can use operator, joy-stick or keyboard inputs, external inputs (weather data e.g. measured wind, wave and current speeds and directions), Earth topological information (digital chart, radar, and sonar data), obstacle and collision avoidance data, and finally the state vector which is available as output from the

navigation/sensor system. The required data are further processed to generate a feasible trajectory for a moving marine vessel using *ad-hoc* techniques or sophisticated methods like interpolation techniques, dynamic optimization, or filtering techniques. By feasible we mean a trajectory limited by the bandwidth of the vessel dynamics.

For a ship or an underwater vehicle, the guidance and control system usually consists of:

- *an attitude control system*
- *a path control system*

In its simplest form the attitude control system is a course autopilot. The main function of the attitude feedback control system is to maintain the vessel in the desired attitude on the ordered path by controlling the vessel in roll, pitch, and yaw. The task of the path controller is to keep the vessel on the predescribed path with some predefined dynamics (e.g. forward speed) by generating orders to the attitude control system. The principles and definitions of *guidance, navigation,* and *control* are further explained in Section 1.3.

5.1 Reference Models

The simplest form of a reference model is to use a low-pass (LP) filter structure:

$$\frac{x_d}{r}(s) = h_{lp}(s) \tag{5.1}$$

where r denotes the command and x_d is the desired state. The choice of filter should reflect the dynamics of the vessel such that a feasible trajectory is constructed. For instance, it is important to take into account physical speed and acceleration limitations of the vessel. The bandwidth of the reference model must also be chosen lower than the bandwidth of the vessel control system in order to obtain satisfactory tracking performance and stability.

One attractive method to generate a smooth reference trajectory $\mathbf{x}_d \in \mathbb{R}^n$ for tracking control is to use a physically motivated model. For marine vessels it is convenient to use reference models motivated by the dynamics of *mass-damper-spring systems* to generate the desired state trajectories, for instance:

$$h_{lp}(s) = \frac{\omega_{n_i}^2}{s^2 + 2\zeta_i\omega_{n_i}s + \omega_{n_i}^2} \tag{5.2}$$

where ζ_i $(i = 1, \ldots, n)$ is the *relative damping ratio* and ω_{n_i} $(i = 1, \ldots, n)$ is the *natural frequency.* This model can be written as a MIMO mass-damper-spring system:

$$\mathbf{M}_d\ddot{\boldsymbol{\eta}}_d + \mathbf{D}_d\dot{\boldsymbol{\eta}}_d + \mathbf{G}_d\boldsymbol{\eta}_d = \mathbf{G}_d\mathbf{r} \tag{5.3}$$

where $\mathbf{M}_d, \mathbf{D}_d$, and $\mathbf{G}_d$ are positive design matrices specifying the desired dynamics of the system. The model (5.3) can also be represented as a linear time invariant (LTI) system:

$$\dot{\mathbf{x}}_d = \mathbf{A}_d\mathbf{x}_d + \mathbf{B}_d\mathbf{r} \tag{5.4a}$$

where $\mathbf{x}_d = [\boldsymbol{\eta}_d^\top, \dot{\boldsymbol{\eta}}_d^\top]^\top \in \mathbb{R}^{2n}$ is the desired state, $\mathbf{r} \in \mathbb{R}^r$ $(r \leq n)$ is a bounded input (command) usually generated by a joy-stick or a keyboard. The state and input matrices are recognized as:

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}_d^{-1}\mathbf{G}_d & -\mathbf{M}_d^{-1}\mathbf{D}_d \end{bmatrix}, \quad \mathbf{B}_d = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_d^{-1}\mathbf{G}_d \end{bmatrix}, \quad \mathbf{C}_d = [\mathbf{I}, \mathbf{0}] \tag{5.5}$$

5.1.1 Velocity Reference Model

The velocity reference model should at least be of order two to obtain smooth reference signals for velocity $\boldsymbol{\nu}_d$ and acceleration $\dot{\boldsymbol{\nu}}_d$. Let $\mathbf{r}^b$ denote the operator input in the *b*-frame. The 2nd-order low-pass filter (5.2) can be used for this purpose. Let:

$$\mathbf{D}_d = \mathbf{M}_d\, 2\boldsymbol{\Delta}\boldsymbol{\Omega}, \quad \mathbf{G}_d = \mathbf{M}_d\, \boldsymbol{\Omega}^2 \tag{5.6}$$

in (5.3) where $\boldsymbol{\Delta} > 0$ and $\boldsymbol{\Omega} > 0$ are diagonal design matrices of *relative damping ratios* and *natural frequencies:*

$$\begin{aligned} \boldsymbol{\Delta} &= \mathrm{diag}\{\zeta_1, \zeta_2, \ldots, \zeta_n\} \\ \boldsymbol{\Omega} &= \mathrm{diag}\{\omega_{n_1}, \omega_{n_2}, \ldots, \omega_{n_n}\} \end{aligned}$$

Next (5.3) can be premultiplied with $\mathbf{M}_d^{-1}$ which results in:

$$\ddot{\boldsymbol{\nu}}_d + 2\boldsymbol{\Delta}\boldsymbol{\Omega}\dot{\boldsymbol{\nu}}_d + \boldsymbol{\Omega}^2\boldsymbol{\nu}_d = \boldsymbol{\Omega}^2\mathbf{r}^b \tag{5.7}$$

where $\boldsymbol{\nu}_d$ is the desired velocity, $\dot{\boldsymbol{\nu}}_d$ is the desired acceleration, and $\ddot{\boldsymbol{\nu}}_d$ is interpreted as the desired "*jerk*." The state space representation is:

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\boldsymbol{\Omega}^2 & -2\boldsymbol{\Delta}\boldsymbol{\Omega} \end{bmatrix}, \quad \mathbf{B}_d = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\Omega}^2 \end{bmatrix} \tag{5.8}$$

Note that a step in the command $\mathbf{r}^b$ will give a step in $\ddot{\boldsymbol{\nu}}_d$ while $\dot{\boldsymbol{\nu}}_d$ and $\boldsymbol{\nu}_d$ will be low-pass filtered and therefore smooth signals in a tracking control system. We also notice that the steady-state velocity is equal to the operator input:

$$\lim_{t\to\infty} \boldsymbol{\nu}_d(t) = \mathbf{r}^b \tag{5.9}$$

5.1.2 Position and Attitude Reference Models

The position and attitude reference model $\boldsymbol{\eta}_d$ is typically chosen of 3rd order for filtering of steps in the *n*-frame input $\mathbf{r}^n$. This suggests that a 1st-order low-pass filter should be cascaded with the mass-damper-spring system. Moreover, consider the transfer function:

$$\frac{\eta_{d_i}}{r_i^n}(s) = \frac{\omega_{n_i}^2}{(1+T_i s)(s^2 + 2\zeta_i\omega_{n_i}s + \omega_{n_i}^2)}, \quad (i = 1, \ldots, n) \tag{5.10}$$

where a 1st-order low-pass filter with time constant $T_i = 1/\omega_{n_i} > 0$ has been added. This can also be written:

$$\frac{\eta_{d_i}}{r_i^n}(s) = \frac{\omega_{n_i}^3}{s^3 + (2\zeta_i + 1)\omega_{n_i}s^2 + (2\zeta_i + 1)\omega_{n_i}^2 s + \omega_{n_i}^3}, \quad (i = 1, \ldots, n) \tag{5.11}$$

or in a vectorial setting as:

$$\boldsymbol{\eta}_d^{(3)} + (2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega}\ddot{\boldsymbol{\eta}}_d + (2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega}^2\dot{\boldsymbol{\eta}}_d + \boldsymbol{\Omega}^3\boldsymbol{\eta}_d = \boldsymbol{\Omega}^3\mathbf{r}^n \tag{5.12}$$

The state space representation is:

$$\mathbf{A}_d = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ -\boldsymbol{\Omega}^3 & -(2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega}^2 & -(2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega} \end{bmatrix}, \quad \mathbf{B}_d = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \boldsymbol{\Omega}^3 \end{bmatrix} \tag{5.13}$$

In the case of n critically damped systems, $\zeta_i = 1$ $(i = 1, \ldots, n)$, we have that $\boldsymbol{\Delta} = \mathbf{I}$. Consequently:

$$\boldsymbol{\eta}_d^{(3)} + 3\boldsymbol{\Omega}\ddot{\boldsymbol{\eta}}_d + 3\boldsymbol{\Omega}^2\dot{\boldsymbol{\eta}}_d + \boldsymbol{\Omega}^3\boldsymbol{\eta}_d = \boldsymbol{\Omega}^3\mathbf{r}^n \tag{5.14}$$

$$\Updownarrow$$

$$(s + \omega_{n_i})^3 \eta_{d_i} = \omega_{n_i}^3 r_i^n \quad (i = 1, \ldots, n) \tag{5.15}$$

These reference models models also satisfies:

$$\lim_{t \to \infty} \boldsymbol{\eta}_d(t) = \mathbf{r}^n \tag{5.16}$$

5.1.3 Saturating Elements

One drawback with a linear reference model is that the time constants in the model often yields a satisfactory response for one operating point of the system while the response for other amplitudes of the operator input r_i results in a completely different behavior. This is due to the exponential convergence of the signals in a linear system. One way to circumvent this problem is to use amplitude gain scheduling so that the the reference model design parameters (ζ_i, ω_i) are scheduled with respect to the magnitude of the input signal r_i.

The performance of the linear reference model can also be improved by including saturation elements for velocity and acceleration. The saturating element is defined as:

$$\text{sat}(x) = \begin{cases} \text{sgn}(x)x_{\max} & \text{if } |x| \geq x_{\max} \\ x & \text{else} \end{cases} \tag{5.17}$$

where the saturation limits:

$$\nu_i \leq \nu_i^{\max}, \quad \dot{\nu}_i \leq \dot{\nu}_i^{\max} \tag{5.18}$$

should reflect the physical limitations of the vessel; see Example 5.1.

These techniques have been used in model reference adaptive control (MRAC) by Ameron-gen (1982, 1984) and adaptive control of underwater vehicles by Fjellstad *et al.* (1992). The position and attitude reference model should therefore be modified as shown in Figure 5.2.

5.1.4 Nonlinear Damping

Nonlinear damping can also be included in the reference model to reduce the velocity for large amplitudes or step inputs r_i. This suggests the modified model:

$$\boldsymbol{\eta}_d^{(3)} + (2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega}\ddot{\boldsymbol{\eta}}_d + (2\boldsymbol{\Delta} + \mathbf{I})\boldsymbol{\Omega}^2\dot{\boldsymbol{\eta}}_d + \mathbf{d}(\dot{\boldsymbol{\eta}}_d) + \boldsymbol{\Omega}^3\boldsymbol{\eta}_d = \boldsymbol{\Omega}^3\mathbf{r}^n \tag{5.19}$$

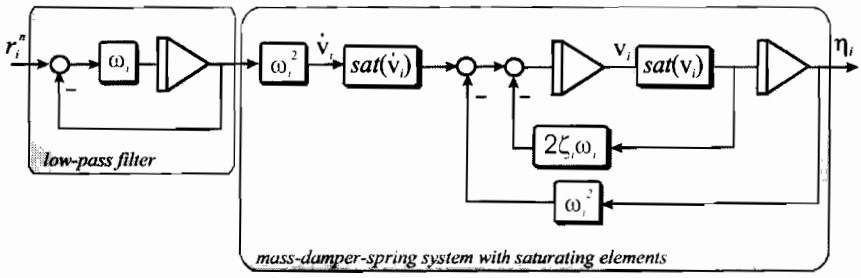


Figure 5.2: Reference model including saturating elements.

where the nonlinear function $\mathbf{d}(\dot{\boldsymbol{\eta}}_d) = \{d_i(\dot{\eta}_{d_i})\}$ could be chosen as:

$$d_i(\dot{\eta}_{d_i}) = \sum_j \delta_{ij} \left|\dot{\eta}_{d_i}\right|^{p_j} \dot{\eta}_{d_i}, \quad (i = 1, \ldots, n) \tag{5.20}$$

where $\delta_{ij} > 0$ are design parameters and $p_j > 0$ are some integers. The effect of nonlinear damping is demonstrated in Example 5.1.

Example 5.1 (Reference Model)
Consider the mass-damper-spring reference model:

$$\dot{x}_d = v_d \tag{5.21}$$

$$\dot{v}_d + 2\zeta\omega_n v_d + \delta\left|v_d\right| v_d + \omega_n^2 x_d = \omega_n^2 r \tag{5.22}$$

where $\zeta = \omega_n = 1$. *Figure 5.3 shows a comparison of responses using* $\delta = 0, \delta = 1$, *and a saturating element,* $v_{\max} = 1$ *for an operator step input* $r = 10$. *The Matlab example file ExRefMod.m in the GNC Toolbox was used to generate the plots.*

5.2 Way-Point Guidance Systems

Systems for way-point guidance are used both for ships and underwater vehicles. These systems consists of a way-point generator with human interface. The selected way-points are stored in a way-point database and used for generation of a trajectory or a path for the moving vessel to follow. Both trajectory and maneuvering control systems can be designed for this purpose. Sophisticated features like weather routing, obstacle avoidance and mission planning can be incorporated in the design of way-point guidance systems. Some of these features will be discussed in the forthcoming section.

5.2.1 Trajectory Tracking and Maneuvering Control

The concepts of *trajectory tracking* and *maneuvering* control should be distinguished when designing way-point guidance systems. The following definitions will be employed (Skjetne *et al.* 2002*d*):

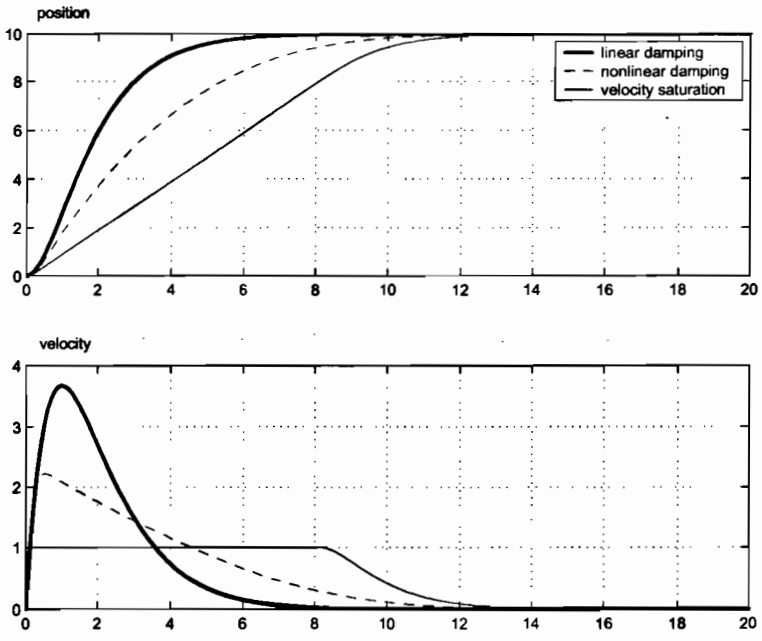


Figure 5.3: Desired position and velocity for a step input $r = 10$.

Definition 5.1 (Tracking Problem)
When the objective is to force the system output $y(t) \in \mathbb{R}^m$ *to track a desired output* $y_d(t) \in \mathbb{R}^m$, *it will be referred to as a* tracking problem *(or in some cases a trajectory tracking problem).*

This definition is consistent with Athans and Falb (1966) and later with Hauser and Hindmann (1995), Ortega *et al.* (1998), and Encarnacao and Pascoal (2001*b*).

Definition 5.2 (Parametrized Path)
A parametrized path is defined as a geometric curve $\boldsymbol{\eta}_d(\theta) \in \mathbb{R}^q$ *with* $q \geq 1$ *parametrized by a continuous path variable* θ.

For surface vessel in 3 DOF it is common to define:

$$\boldsymbol{\eta}_d^{3\text{DOF}}(\theta) = [x_d(\theta), y_d(\theta), \psi_d(\theta)]^\top \tag{5.23}$$

while an extension to 6 DOF path control is:

$$\boldsymbol{\eta}_d^{6\text{DOF}}(\theta) = [x_d(\theta), y_d(\theta), z_d(\theta), \phi_d(\theta), \theta_d(\theta), \psi_d(\theta)]^\top \tag{5.24}$$

Let the derivatives of $\boldsymbol{\eta}(\theta)$ with respect to θ be denoted as $\boldsymbol{\eta}'$, $\boldsymbol{\eta}''$, $\boldsymbol{\eta}^{(3')}, \ldots, \boldsymbol{\eta}^{(i')}$. Hence, the path characterization vector can be defined as:

Definition 5.3 (Path Characterization Vector)
The path characterization vector is defined as:

$$\boldsymbol{\eta}_{cv}(\theta) = \left[\boldsymbol{\eta}'(\theta)^\top, \boldsymbol{\eta}''(\theta)^\top, \ldots, \boldsymbol{\eta}^{(n')}(\theta)^\top\right]^\top$$

For $n = 2$ the path characterization vector will define the desired velocities and accelerations of the vessel as a function of the path variable θ. This is used to define a feasible path–i.e., a path which is possible to follow for a ship or an underwater vehicle. It is important that the path characterization vector reflects physical limitations like maximum velocity or acceleration constraints of the vessel.

Definition 5.4 (Maneuvering Problem)
The maneuvering problem involves solving two tasks:
1) Geometric Task: force the state $\mathbf{y}(t)$ *to converge to a desired path* $\mathbf{y}_d(\theta(t))$*:*

$$\lim_{t\to\infty} [\mathbf{y}(t) - \mathbf{y}_d(\theta(t))] = \mathbf{0} \tag{5.25}$$

for any continuous function $\theta(t)$.
2) Dynamic Task: force the speed $\dot{\theta}$ *to converge to a desired speed* v_s*:*

$$\lim_{t\to\infty} \left[\dot{\theta}(t) - v_s(\theta(t))\right] = 0 \tag{5.26}$$

Definition 5.4 implies that the dynamics $\theta = \theta(t)$ along the path can be specified independently of the error dynamics. A special case of the maneuvering problem is:

$$\dot{\theta}(t) = 1, \quad \theta(0) = 0 \tag{5.27}$$

which is recognized as the tracking problem since the solution of (5.27) is $\theta = t$.

Trajectory Tracking and Maneuvering Control

Methods for trajectory tracking and maneuvering control are described in Chapter 10. These methods are classified according to the number of available actuators. This can be illustrated by considering a marine vessel in *surge, sway* and *yaw*–i.e., 3 DOF horizontal motion.

- **Trajectory Tracking Control:** Tracking of a *time-varying reference trajectory* $\boldsymbol{\eta}_d(t) = [x_d(t), y_d(t), \psi_d(t)]^\top$ is achieved by minimizing the tracking error, $\boldsymbol{\eta}(t) - \boldsymbol{\eta}_d(t)$, where $\boldsymbol{\eta}(t) = [x(t), y(t), \psi(t)]^\top$. The tracking error can be decomposed in a *vessel parallel* (VP) reference frame according to:

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \mathbf{R}^\top(\psi)(\boldsymbol{\eta} - \boldsymbol{\eta}_d) \tag{5.28}$$

where $\mathbf{R}(\psi) \in SO(3)$ is the rotation matrix in yaw. Expanding (5.28), yields

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\psi(x - x_d) + \sin\psi(y - y_d) \\ -\sin\psi(x - x_d) + \cos\psi(y - y_d) \\ \psi - \psi_d \end{bmatrix} \tag{5.29}$$

The physical interpretations of e_i $(i = 1, \ldots, 3)$ are:

$$\begin{aligned} e_1 &= \text{path tangential tracking error} \\ e_2 &= \text{cross-track error (normal to path)} \\ e_3 &= \text{heading error} \end{aligned}$$

Based on this definition the following considerations can be made:

- **Three or more controls:** This is referred to as a *fully actuated* dynamic positioning (DP) system and typical application are crab-wise motions (low speed maneuvering) and station-keeping. DP control algorithms are discussed in Chapter 11.
- **Two controls:** DP in 3 DOF with only two controls is an active area of research. This is an *underactuated* control problem which is impossible to solve using linear theory; see Pettersen and Egeland (1999), Pettersen and Fossen (2000), Pettersen and Nijmeijer (2001), Jiang (2002), Do *et al.* (2002a, 2002b, 2002c), for instance. The benefit is additional safety in case of actuator failure. Moreover, a underactuated control philosophy can serve as a back-up solution in an emergency situation.
- **One control:** 3 DOF position control with only one actuator is not considered to be of practical importance and hardly possible.

- **Maneuvering Control:** Tracking a *path* $(x_d(\theta), y_d(\theta))$ with speed requirements:
  - **Three or more controls:** Regulation of the positions (x, y) to $(x_d(\theta), y_d(\theta))$. Since this only requires two controls it gives additional flexibility in that the heading angle ψ can be controlled to an arbitrary value $\psi_d(\theta)$ during path following.
  - **Two controls:** The conventional actuator configuration in this case is 1) rudder servo (or a transverse thruster) used to regulate the cross-track error to zero and 2) forward thrust (or a transverse thruster) intended for forward speed/acceleration assignments.
  - **One control:** The cross-track error given by (5.29) can be minimized to zero by using only one control, typically a rudder, while the ship moves at constant speed.

Control systems for trajectory tracking and maneuvering control are discussed in Chapter 10.

5.2.2 Way-Point Representation

The route of a ship or an underwater vehicle is usually specified in terms of way-points. Each way-point is defined using Cartesian coordinates (x_k, y_k, z_k) for $i = 1, \ldots, n$. The way-point *database* therefore consists of:

$$\text{wpt.pos} = \{(x_0, y_0, z_0), (x_1, y_1, z_1), \ldots, (x_n, y_n, z_n)\}$$

For surface vessel, only two coordinates (x_k, y_k) are used. Additionally, other way-point properties like speed, heading etc., can be defined, i.e.:

$$\begin{aligned} \text{wpt.speed} &= \{U_0, U_1, \ldots, U_n\} \\ \text{wpt.heading} &= \{\psi_0, \psi_1, \ldots, \psi_n\} \end{aligned}$$

For surface vessels this means that the vessel should pass through way-point (x_i, y_i) at forward speed U_i with heading angle ψ_i. The heading is usually unspecified during cross-tracking, whereas it is more important during a crab-wise maneuver close to offshore installations (dynamic positioning).

The way-point database can be generated using many criteria. These are usually based on:

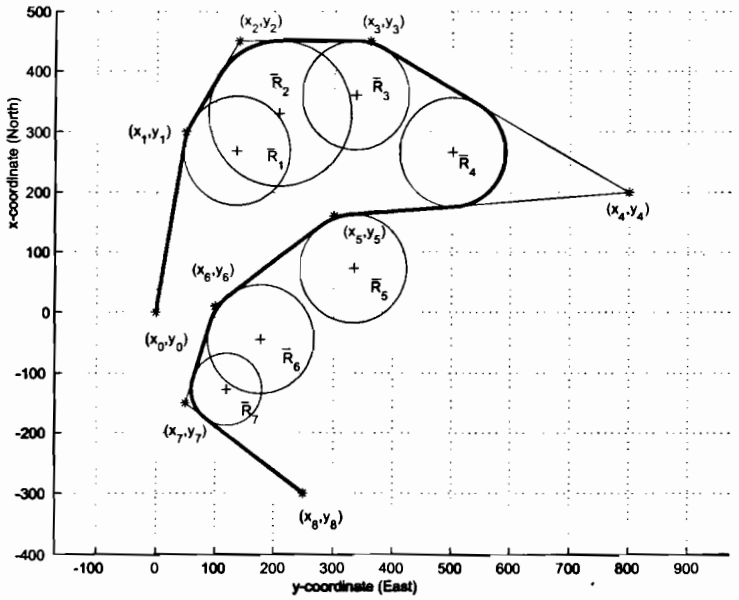


Figure 5.4: Straight-lines and inscribed circles used for way-point guidance.

- **Mission:** the vessel should move from some starting point (x_0, y_0, z_0) to the terminal point (x_n, y_n, z_n) via the way-points (x_i, y_i, z_i).
- **Environmental data:** information about wind, waves, and currents can be used for energy optimal routing (or avoidance of bad weather for safety reasons).
- **Geographical data:** information about shallow waters, islands etc. should be included.
- **Obstacles:** floating constructions and other obstacles must be avoided.
- **Collision avoidance:** avoiding moving vessels close to your own route by introducing safety margins.
- **Feasibility:** each way-point must be feasible, in that it must be possible to maneuver to the next way-point without exceeding maximum speed, turning rate etc.

On-line replanning can be used to update the way-point database in case of time-varying conditions like changing weather, moving vessels (collision avoidance) etc. Optimality with regard to weather is discussed in Section 5.2.5. This is referred to as weather routing.

Path Generation using Straight-Lines and Circular Arcs

In practise it is common to represent the desired path using straight-lines and circle arcs to connect the way-points. This is shown in Figure 5.4 where the inscribed circle between two straight lines describes the desired turn. The radius of the inscribed circle is denoted $\bar{R}_i$ $(i = 1, \ldots, n)$.

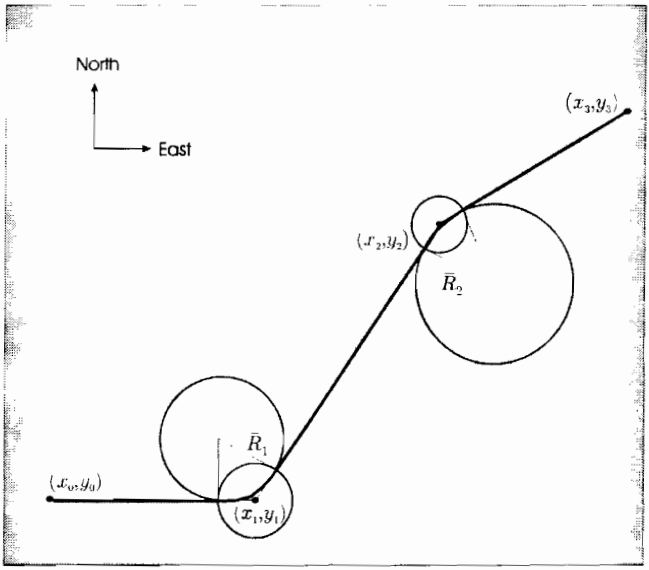


Figure 5.5: Path consisting of straight-lines and circular arcs. The white circle intersects with the inscribed circle at the turning point.

The drawback of this strategy, in comparison to a cubic interpolation strategy, for instance, is that a jump in the desired yaw rate r_d is experienced. This is due to the fact that the desired yaw rate along the straight line is $r_d = 0$ while it is $r_d =$ constant on the circle arc during steady turning. Hence, there will be a jump in the desired yaw rate during transition from the straight-line to the circle arc. This produces a small off-set during cross-tracking. If a smooth reference trajectory, e.g. generated by interpolation, is used, these drawbacks are overcome. However, it is convenient to use straight-lines and circle arcs due to their simplicity. Another consideration is that the human operator can specify a circle with radius R_i around each way-point (white circle in Figure 5.5). These values are stored in the database as:

$$\text{wpt.radius} = \{R_0, R_1, \ldots, R_n\}$$

The point where the circle arc intersects the straight-line represent the turning point of the ship. Hence, the radius of the inscribed circle can be computed from R_i as:

$$\bar{R}_i = R_i \tan \alpha_i, \quad (i = 1, \ldots, n) \tag{5.30}$$

where α_i is defined in Figure 5.6.

5.2.3 Trajectory Generation using a Vessel Simulator

A time-varying reference trajectory for a moving vessel can be generated using a closed loop model of the vessel where the time constants, relative damping ratios, and natural frequencies are chosen to reflect physical limitations of the vessel. For instance, the vessel model can be chosen as:

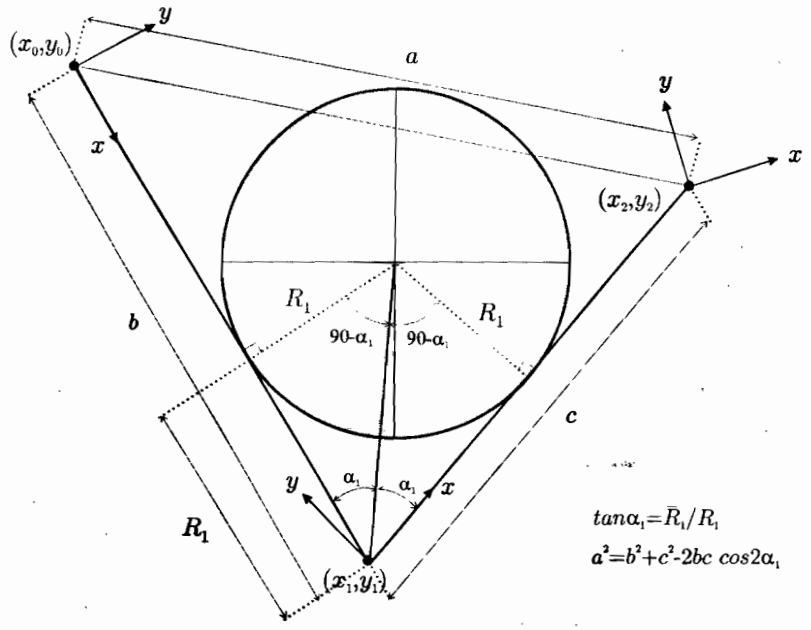


Figure 5.6: Circle with radius $\bar{R}_1$ inscribed between the points (x_0, y_0), (y_1, y_2), and (x_3, y_3).

$$\dot{\eta}_d = \mathbf{J}(\eta_d)\nu_d \tag{5.31}$$

$$\mathbf{M}\dot{\nu}_d + \mathbf{N}\nu_d + \mathbf{g}(\eta_d) = \tau \tag{5.32}$$

where the damping matrix for simplicity is modelled as a diagonal matrix with design parameters:

$$\mathbf{N} = \text{diag}\{n_1, \ldots, n_6\} > 0 \tag{5.33}$$

The system inertia matrix $\mathbf{M}$ is included in the model to guarantee proper scaling of the control inputs τ. Smooth reference trajectories $(\eta_d(t), \nu_d(t))$ are then obtained by simulating the model under closed-loop control, for instance by using a nonlinear PD-controller (see Section 7.1.4):

$$\tau = \mathbf{g}(\eta_d) - \mathbf{J}^\top(\eta_d)\left[\mathbf{K}_p(\eta_d - \eta_{\text{ref}}) + \mathbf{K}_d\dot{\eta}_d\right] \tag{5.34}$$

where the set-point vector η_{ref} is chosen equal to the way-points coordinates (x_k, y_k, z_k). The control law (5.34) is in fact a *guidance controller* since it is applied to the reference model. In addition to this, it is smart to include saturation elements for velocity and acceleration to keep these quantities within their physical limits.

A switching strategy between the set-points (way-points) must also be adopted. One simple way to do this is to use a circle of acceptance; see Section 5.3.

Example 5.2 (Generation of Reference Trajectory using a Vessel Model)
The desired reference trajectories of a ship can be modelled as:

$$\dot{x}_d = U_d \cos\psi_d \tag{5.35}$$
$$\dot{y}_d = U_d \sin\psi_d \tag{5.36}$$

with forward speed dynamics:

$$(m - X_{\dot{u}})\dot{U}_d + \frac{1}{2}\rho C_d A U_d^2 = \tau \tag{5.37}$$

where $U_d > 0$ is the reference speed, ρ is the density of water, C_d is the drag coefficient, A is the projected cross-sectional area of the submerged hull in the x-direction, and $m - X_{\dot{u}}$ is the mass included hydrodynamic added mass. The course dynamics is chosen as:

$$\dot{\psi}_d = r_d \tag{5.38}$$
$$T\dot{r}_d + r_d = K\delta \tag{5.39}$$

where K and T are design parameters. The guidance system has two inputs, thrust τ and rudder angle δ. The guidance controllers can be chosen of PI and PID types:

$$\tau = -K_{p1}(U_d - U_{\text{ref}}) - K_{i1}\int_0^t (U_d - U_{\text{ref}})d\tau \tag{5.40}$$

and

$$\delta = -K_{p2}(\psi_d - \psi_{\text{ref}}) - K_{i2}\int_0^t (\psi_d - \psi_{\text{ref}})d\tau - K_{d2}r_d \tag{5.41}$$

where ψ_{ref} is generated using a LOS algorithm (see Section 5.3):

$$\psi_{\text{ref}} = atan2\,(y_k - y_d(t), x_k - x_d(t)) \tag{5.42}$$

Numerical integration of the ODEs (5.35)–(5.39) with feedback (5.40)–(5.41) yields a smooth reference trajectory $(x_d(t), y_d(t), \psi_d(t))$ with speed assignment $U_d(t)$.

5.2.4 Path and Trajectory Generation using Interpolation

It is attractive to use spline or polynomial interpolation methods to generate a path $(x_d(\theta), y_d(\theta))$ through the predefined way-points. Notice that a trajectory $(x_d(t), y_d(t))$ is obtained by choosing $\dot{\theta} = k$ such that $\theta = kt$ $(k = 1, \ldots, N)$.

Cubic Spline and Hermite Interpolation

In Matlab™ several methods for interpolation are available.

Matlab:
The different methods for interpolation are found by typing:

```
help polyfun
```

Two useful methods path generation are the cubic spline interpolant (`spline.m`) and the piecewise cubic Hermite interpolating polynomial (`pchip.m`).

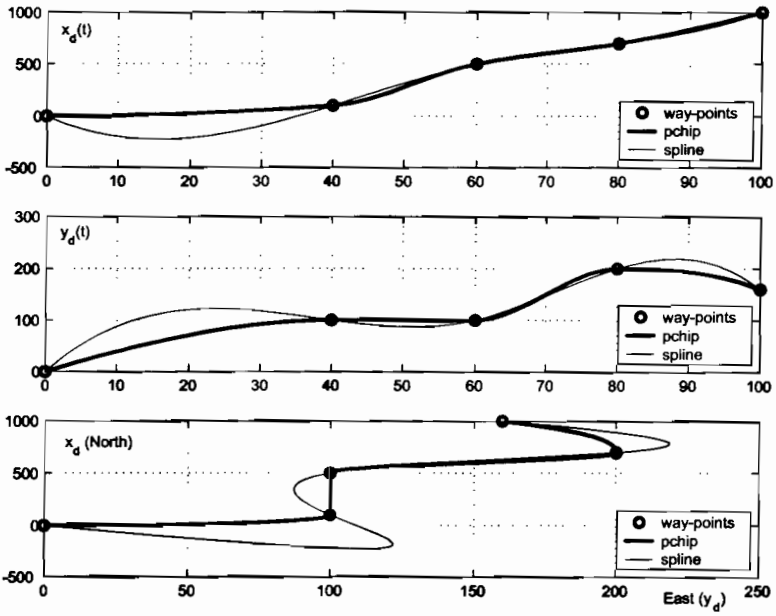


Figure 5.7: Results using cubic Hermite and spline interpolation, see ExSpline.m.

The main difference between cubic spline and Hermite Interpolation are how the slopes at the end points are handled. For simplicity let us consider the problem of trajectory generation. The cubic Hermite interpolant ensures that the first derivatives $(\dot{x}_d(t), \dot{y}_d(t))$ are continuous. In addition, the slopes at each endpoint are chosen in such a way that $(x_d(t), y_d(t))$ are shape preserving and respects monotonicity.

Cubic spline interpolation is usually done by requiring that the 2nd derivative $(\ddot{x}_d(t), \ddot{y}_d(t))$ at the endpoints of the polynomials are equal. This gives a smooth spline. Consequently, the cubic spline will be more accurate than Hermite interpolating polynomial if the data values are of a smooth function. The cubic Hermite interpolant, on the contrary, has less oscillations if the data are non-smooth.

The results of interpolating a set of predefined way-points to a trajectory $(x_d(t), y_d(t))$ using the cubic Hermite interpolant and cubic spline interpolation methods are shown in Figure 5.7. It is seen that a different behavior is obtained due to the conditions on the 1st and 2nd derivatives on the endpoints.

Polynomial Interpolation

Instead of using the Matlab functions `pchip.m` and `spline.m` a cubic spline can be interpolated through a set of way-points by considering the *cubic polynomials:*

$$x_d(\theta) = a_3\theta^3 + a_2\theta^2 + a_1\theta + a_0 \quad (5.43)$$

$$y_d(\theta) = b_3\theta^3 + b_2\theta^2 + b_1\theta + b_0 \quad (5.44)$$

Matlab:
The script `ExSpline.m` generates the plots in Figure 5.7.

```
% ExSpline - Cubic Hermite and spline interpolation of way-points

wpt.pos.x = [0 100 500 700 1000];
wpt.pos.y = [0 100 100 200 160];
wpt.time  = [0 40 60 80 100];

t = 0:1:max(wpt.time);                % time
x_p = pchip(wpt.time,wpt.pos.x,t);  % cubic Hermite interpolation
y_p = pchip(wpt.time,wpt.pos.y,t);
x_s = spline(wpt.time,wpt.pos.x,t); % spline interpolation
y_s = spline(wpt.time,wpt.pos.y,t);

subplot(311), plot(wpt.time,wpt.pos.x,'o',t,[x_p; x_s])
subplot(312), plot(wpt.time,wpt.pos.y,'o',t,[y_p; y_s])
subplot(313), plot(wpt.pos.y,wpt.pos.x,'o',y_p,x_p,y_s,x_s)
```

where $(x_d(\theta), y_d(\theta))$ are the position of the vessel and where θ is a path variable given by:

$$\dot{\theta} = f(\theta, t) \tag{5.45}$$

The partial derivatives of $x_d(\theta)$ and $y_d(\theta)$ with respect to θ are:

$$x'_d(\theta) = \frac{dx_d(\theta)}{d\theta} = 3a_3\theta^2 + 2a_2\theta + a_1 \tag{5.46}$$

$$y'_d(\theta) = \frac{dy_d(\theta)}{d\theta} = 3b_3\theta^2 + 2b_2\theta + b_1 \tag{5.47}$$

Hence, the speed $U_d(t)$ of the vessel can be computed as:

$$\dot{x}_d(t) = \frac{dx_d(\theta)}{d\theta}\dot{\theta}(t) \tag{5.48}$$

$$\dot{y}_d(t) = \frac{dy_d(\theta)}{d\theta}\dot{\theta}(t) \tag{5.49}$$

resulting in:

$$\begin{aligned} U_d(t) &= \sqrt{\dot{x}_d^2(t) + \dot{y}_d^2(t)} \\ &= \sqrt{x'_d(\theta)^2 + y'_d(\theta)^2}\ \dot{\theta}(t) \end{aligned} \tag{5.50}$$

Similarly an expression for the acceleration $\dot{U}_d(t)$ can be found. The unknown parameters $a_0, a_1, a_2, a_3, b_0, b_1, b_2, b_3$ can be computed using the following algorithm.

Cubic Spline Algorithm for Path Generation

The path through the way-points (x_{k-1}, y_{k-1}) and (x_k, y_k) must satisfy:

$$x_d(\theta_{k-1}) = x_{k-1}, \quad x_d(\theta_k) = x_k \tag{5.51}$$
$$y_d(\theta_{k-1}) = y_{k-1}, \quad y_d(\theta_k) = y_k \tag{5.52}$$

where $k = 1, \ldots, n$. In addition, smoothness is obtained by requiring that:

$$\lim_{\theta \to \theta_k^-} x_d'(\theta_k) = \lim_{\theta \to \theta_k^+} x_d'(\theta_k) \tag{5.53}$$
$$\lim_{\theta \to \theta_k^-} x_d''(\theta_k) = \lim_{\theta \to \theta_k^+} x_d''(\theta_k) \tag{5.54}$$

For this problem, it is possible to add only two boundary conditions (velocity or acceleration) for the x- and y-equations, respectively. Moreover:

$$x_d'(\theta_0) = x_0', \quad x_d'(\theta_n) = x_n' \tag{5.55}$$
$$y_d'(\theta_0) = y_0', \quad y_d'(\theta_n) = y_n' \tag{5.56}$$

or

$$x_d''(\theta_0) = x_0'', \quad x_d''(\theta_n) = x_n'' \tag{5.57}$$
$$y_d''(\theta_0) = y_0'', \quad y_d''(\theta_n) = y_n'' \tag{5.58}$$

The polynomial $x_d(\theta_k)$ is given by the parameters $\mathbf{a}_k = [a_{3k}, a_{2k}, a_{1k}, a_{0k}]^\top$, resulting in $4(n-1)$ unknown parameters. The number of constraints are also $4(n-1)$ if only velocity or acceleration constraints are chosen at the end points. The unknown parameters for n way-points are collected into a vector:

$$\mathbf{x} = [\mathbf{a}_k^\top, \ldots, \mathbf{a}_{n-1}^\top]^\top \tag{5.59}$$

Hence, the cubic interpolation problem can be written as a linear equation:

$$\mathbf{y} = \mathbf{A}(\theta_{k-1}, \ldots, \theta_k)\mathbf{x}, \quad k = 1, 2, \ldots, n \tag{5.60}$$

where:

$$\mathbf{y} = [x_{\text{start}}, x_0, x_1, x_1, 0, 0, x_2, x_2, 0, 0, \ldots, x_n, x_{\text{final}}]^\top \tag{5.61}$$

The start and end points can be specified in terms of velocity or acceleration constraints

$x_{\text{start}} \in \{x'_0, x''_0\}$ and $x_{\text{final}} \in \{x'_n, x''_n\}$, respectively. This gives:

$$\mathbf{A}(\theta_{k-1}, \theta_k) = \left[\begin{array}{ccccc} \mathbf{c}_{\text{start}} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \cdots & \mathbf{0}_{1\times4} \\ \mathbf{p}(\theta_0) & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ \hline \mathbf{p}(\theta_1) & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ 0 & \mathbf{p}(\theta_1) & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ -\mathbf{v}(\theta_1) & \mathbf{v}(\theta_1) & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ -\mathbf{a}(\theta_1) & \mathbf{a}(\theta_1) & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ \hline \mathbf{0}_{1\times4} & \mathbf{p}(\theta_2) & \mathbf{0}_{1\times4} & & \mathbf{0}_{1\times4} \\ \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \mathbf{p}(\theta_2) & & \mathbf{0}_{1\times4} \\ \mathbf{0}_{1\times4} & -\mathbf{v}(\theta_2) & \mathbf{v}(\theta_2) & & \mathbf{0}_{1\times4} \\ \mathbf{0}_{1\times4} & -\mathbf{a}(\theta_2) & \mathbf{a}(\theta_2) & & \mathbf{0}_{1\times4} \\ \hline \vdots & & & \ddots & \\ \hline \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & & \mathbf{p}(\theta_n) \\ \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times4} & \cdots & \mathbf{c}_{\text{final}} \end{array}\right] \tag{5.62}$$

where $\mathbf{c}_{\text{start}} \in \{x'_d(\theta_0), x''_d(\theta_0)\}$, $\mathbf{c}_{\text{final}} \in \{x'_d(\theta_n), x''_d(\theta_n)\}$ and:

$$\mathbf{p}(\theta_k) = [\theta_k^3,\ \theta_k^2,\ \theta_k,\ 1] \tag{5.63}$$

$$\mathbf{v}(\theta_k) = \mathbf{p}'(\theta_k) = [3\theta_k^2,\ 2\theta_k,\ 1,\ 0] \tag{5.64}$$

$$\mathbf{a}(\theta_k) = \mathbf{p}''(\theta_k) = [6\theta_k,\ 2,\ 0,\ 0] \tag{5.65}$$

Equation (5.60) can be solved for $\theta_k = 0, 1, \ldots, n$ according to:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{y} \tag{5.66}$$

The formulas for $\mathbf{b}_k = [b_{3k}, b_{2k}, b_{1k}, b_{0k}]^\top$ are obtained in a similar manner.

Matlab:
Formula (5.66) has been implemented in the script `ExPathGen.m` and `pva.m`. The results for the following set of way-points:

```
wpt.pos.x = [0 200 400 700 1000];
wpt.pos.y = [0 200 500 400 1200];
```

where $\theta = 0, \ldots, 4$ are shown in Figures 5.8 and 5.9.

Transformation of Path to Reference Trajectories using Desired Speed Profiles

In Figure 5.9 it is seen that the solution between two successive way-points:

$$x_d(\theta) = a_3\theta^3 + a_2\theta^2 + a_1\theta + a_0 \tag{5.67}$$

$$y_d(\theta) = b_3\theta^3 + b_2\theta^2 + b_1\theta + b_0 \tag{5.68}$$

indeed is a *time-independent* path when $x_d(\theta)$ is plotted against $y_d(\theta)$ for increasing θ-values. The path can be transformed to a *time-varying* trajectory by defining a *speed profile.*

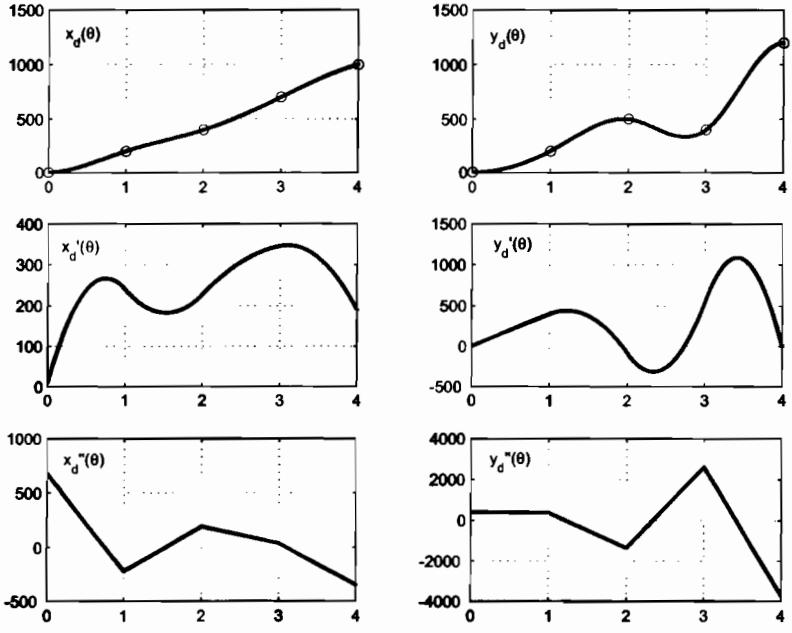


Figure 5.8: The plots shows the cubic polynominals $x_d(\theta)$ and $y_d(\theta)$ and their first and second derivatives.

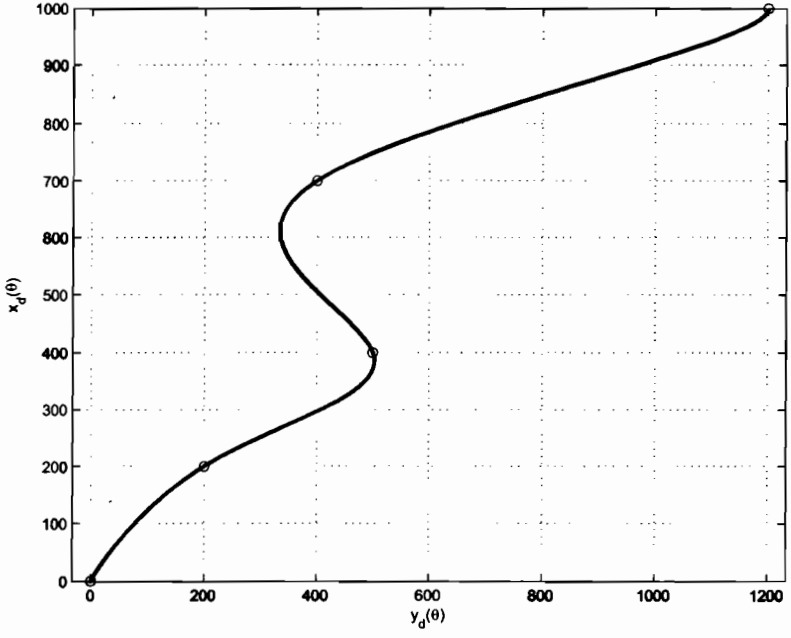


Figure 5.9: xy-plot based on cubic spline optimization.

The speed profile assigns dynamics to $\theta(t)$ such that the desired path transforms to a time-dependent reference trajectory at the same time as the desired speed and acceleration profiles are preserved. From (5.50) it is seen that:

$$\dot{\theta}(t) = \frac{U_d(t)}{\sqrt{x_d'(\theta)^2 + y_d'(\theta)^2}}, \quad \theta(t_k) = k, \quad (k = 0, 1, 2, ..., n) \tag{5.69}$$

where $\theta(t_k) = k$ is the initial condition of the differential equation, and $U_d(t)$ is the desired speed profile. Let U_{ref} be the input to a 1st-order system:

$$T\dot{U}_d(t) + U_d(t) = U_{\text{ref}}, \quad T > 0 \tag{5.70}$$

A smooth transition from the desired speed $U_d(t_k)$ at way-point k to the next way-point $k+1$ where the specification:

$$U_{\text{ref}} = U_d(t_{k+1}) \tag{5.71}$$

can be made. This is illustrated in the following example.

Example 5.3 (Transformation of Path to Reference Trajectories)
Consider the first two way-points in the example file `ExPathGen.m`*:*

$$\begin{aligned}(x_0, y_0) &= (0, 0)\\ (x_1, y_1) &= (200, 200)\end{aligned}$$

The cubic polynomials satisfying (5.66) are:

$$\begin{aligned}x_d(\theta) &= -29.89\,\theta^3 + 135.63\,\theta^2 + 94.25\,\theta\\ y_d(\theta) &= 108.05\,\theta^3 - 2.30\,\theta^2 + 94.25\,\theta\end{aligned}$$

for $\theta \in [0, 1]$. *Let the speed dynamics time constant be* $T = 10$ *(s). Assume that the vessel is initially at rest* ($U_d(t_0) = 0$) *and that the desired speed of way-point no.* 1 *is* $U_{\text{ref}} = U_d(t_1) = 5.0$ *(m/s). The numerical solutions of:*

$$\dot{\theta}(t) = \frac{U_d(t)}{\sqrt{x_d'(\theta)^2 + y_d'(\theta)^2}} \tag{5.72}$$

$$T\dot{U}_d(t) + U_d(t) = U_{\text{ref}} \tag{5.73}$$

for way-points 0 *and* 1 *corresponding to* $\theta_0(t_0) = 0$ *and* $\theta_1(t_1) = 1$ *with* $t_0 = 0$ *and* t_1 *unknown, is shown in Figure 5.10, see* `ExPathGen.m`*. It is seen that the desired speed of* 5.0 *(m/s) is reached in approximately* 67 *(s). Hence, the terminal time must be chosen as* $t_1 \geq 67$ *(s) (corresponding to* $\theta(t_1) = 1$*) in order to satisfy the desired speed dynamics. If* $t_1 < 67$ *(s) there is not enough time to reach the desired speed of way-point no.* 1 *unless the time constant* T *is reduced. The time constant should reflect what is physically possible for the vessel. Notice that the path* $(x_d(\theta), y_d(\theta))$ *has been transformed to a time-varying reference trajectory* $(x_d(t), y_d(t))$ *by assigning a speed profile (5.72) to be solved numerically with the path planner (5.66). This gives design flexibility since the path can be generated off-line using a way-point database while speed is assigned to the path when the dynamics of the actual vessel is considered.*

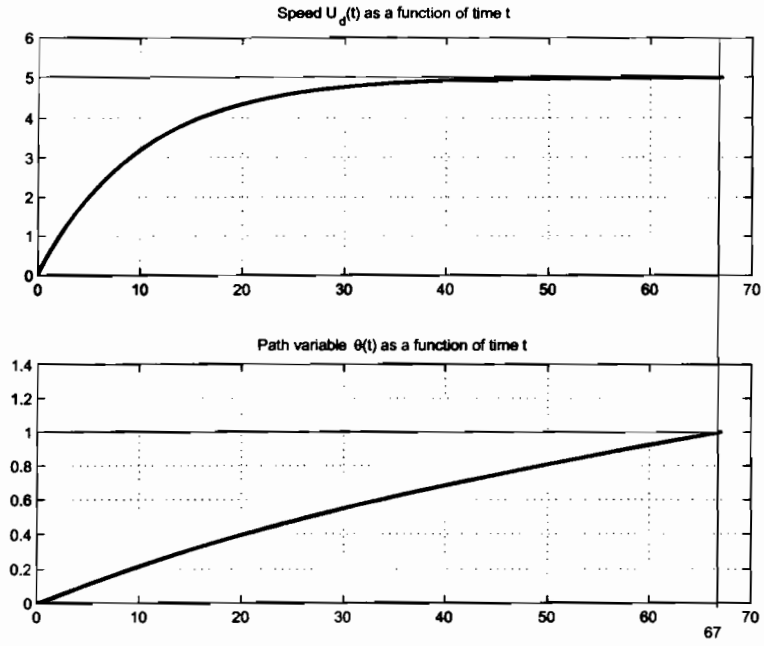


Figure 5.10: Upper plot shows that the speed $U_d(t)$ reaches the desired value of 5.0 (m/s) in approximately 67 (s). The lower plot shows that the path variable $\theta(t)$ is incremented from 0 to 1 during the speed transition.

Nonlinear Constrained Optimization

Another solution to trajectory and path generation is to use nonlinear constrained optimization techniques. These methods allows an object function to be specified where minimum time, energy etc. are design goals. In addition, speed and acceleration constraints of the vessel can be added. The drawback is that nonlinear constraint optimization problems are much harder to solve numerically than the methods described in the previous sections. The Matlab^{TM} optimization toolbox will be used to demonstrate how this can be done.

In general trajectory tracking and path planning problems can be formulated as:

$$J = \min_{\mathbf{x}} \{f(\mathbf{x})\} \tag{5.74}$$

$$\begin{aligned} \text{subject to} \quad & g_k(\mathbf{x}) \leq 0 && (k = 1, \ldots, n_g) \\ & h_j(\mathbf{x}) = 0 && (j = 1, \ldots, n_h) \\ & x_{i,\min} \leq x_i \leq x_{i,\max} && (i = 1, \ldots, n_x) \end{aligned}$$

where $f(\mathbf{x})$ should be minimized with respect to the parameter vector $\mathbf{x}$ with $g_i(\mathbf{x})$ and $h_j(\mathbf{x})$ as nonlinear inequality and equality constraints, respectively. An attractive simplification is to use quadratic programming, that is:

$$J = \min_{\mathbf{x}} \left\{ \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x} + \mathbf{f}^\top \mathbf{x} \right\} \tag{5.75}$$

subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$

$x_{i,\min} \leq x_i \leq x_{i,\max} \quad (i = 1, \ldots, n_x)$

For simplicity, consider two way-points (x_k, y_k) and (x_{k+1}, y_{k+1}) satisfying:

$$x(t_k) = x_k, \qquad y(t_k) = y_k \tag{5.76}$$

$$x(t_{k+1}) = x_{k+1}, \qquad y(t_{k+1}) = y_{k+1} \tag{5.77}$$

Choosing the speed constraints as:

$$\dot{x}_d(t) = U_d(t)\cos\psi_d(t) \tag{5.78}$$

$$\dot{y}_d(t) = U_d(t)\sin\psi_d(t) \tag{5.79}$$

where the angle $\psi_d(t)$ is computed as $\psi_d(t_k) = \text{atan2}(y_{k+1} - y_k, x_{k+1} - x_k)$, that is with direction towards the next way-point. Hence:

$$\dot{x}_d(t_k) = U_k \cos\psi_k \tag{5.80}$$

$$\dot{y}_d(t_k) = U_k \sin\psi_k \tag{5.81}$$

For two way-points this results in:

$$\mathbf{y} = \mathbf{A}(t_k, t_{k+1})\mathbf{x} \tag{5.82}$$

where:

$$\mathbf{y} = [x_k, x_{k+1}, y_k, y_{k+1}, U_k\cos\psi_k, U_k\sin\psi_k, U_{k+1}\cos\psi_{k+1}, U_{k+1}\sin\psi_{k+1}]^\top \tag{5.83}$$

and:

$$\mathbf{A}(t_k, t_{k+1}) = \begin{bmatrix} t_k^3 & t_k^2 & t_k & 1 & 0 & 0 & 0 & 0 \\ t_{k+1}^3 & t_{k+1}^2 & t_{k+1} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & t_k^3 & t_k^2 & t_k & 1 \\ 0 & 0 & 0 & 0 & t_{k+1}^3 & t_{k+1}^2 & t_{k+1} & 1 \\ 3t_k^2 & 2t_k & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3t_k^2 & 2t_k & 1 & 0 \\ 3t_{k+1}^2 & 2t_{k+1} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3t_{k+1}^2 & 2t_{k+1} & 0 & 0 \end{bmatrix} \tag{5.84}$$

The criterion to minimize is:

$$J = \min_{\mathbf{x}} \left\{ [\mathbf{A}(t_k, t_{k+1})\mathbf{x} - \mathbf{y}^\top][\mathbf{A}(t_k, t_{k+1})\mathbf{x} - \mathbf{y}] \right\} \tag{5.85}$$

for given pairs (t_k, t_{k+1}) of time. Expanding this expression, yields:

$$\bar{J} = \frac{1}{2}\left(J - \mathbf{y}^\top\mathbf{y}\right) = \min_{\mathbf{x}} \left\{ \frac{1}{2}\mathbf{x}^\top \mathbf{A}^\top(t_k, t_{k+1})\mathbf{A}(t_k, t_{k+1})\mathbf{x} - \mathbf{y}^\top \mathbf{A}(t_k, t_{k+1})\mathbf{x} \right\} \tag{5.86}$$

implying that

$$\mathbf{H} = \mathbf{A}^{\top}(t_k, t_{k+1})\mathbf{A}(t_k, t_{k+1}) \quad (5.87)$$

$$\mathbf{f} = -\mathbf{y}^{\top}\mathbf{A}(t_k, t_{k+1}) \quad (5.88)$$

In this expression the starting time t_k is given while the arrival time t_{k+1} is unknown. The cubic polynomials (5.43)–(5.44) imply that there are 8 additional unknown parameters to optimize:

$$\mathbf{x} = [a_3, a_2, a_1, a_0, b_3, b_2, b_1, b_0]^{\top} \quad (5.89)$$

giving a total of 9 unknown parameters. In addition linear constraints $\mathbf{Ax} \leq \mathbf{b}$ can be added. The reference trajectory can be found using quadratic programming.

Matlab:
Trajectory generation using the optimization toolbox is demonstrated in the following example:

Example 5.4 (Trajectory Generation using Quadratic Programming)
Consider two way-points:

$$(x_0, y_0) = (10, 10)$$
$$(x_1, y_1) = (200, 100)$$

with speed constraint:

$$U_d(t) \leq 10 \text{ (m/s)}$$

in the GNC toolbox script:

`ExQuadProg`

The desired speed in the way-points are $U_0(t_0) = 0$ (m/s) and $U_1(t_1) = 5$ (m/s) with $t_0 = 0$ (s). The arrival time t_1 is computed in a for–loop by solving the quadratic optimization problem (5.75) for each time $t_1 = t_0 + dt$ where dt is incremented by 1.0 (s) each time. This process is terminated when the first solution $U_d(t) \leq U_{\max}$ is reached (this can be easily changed if other requirements are more important). The optimal solution:

$$x_d(t) = -0.0102\,t^3 + 0.5219\,t^2 - 4.28 \cdot 10^{-12}\,t + 10.0$$
$$y_d(t) = -0.0048\,t^3 + 0.2472\,t^2 - 1.04 \cdot 10^{-12}\,t + 10.0$$

for $t \in [t_0, t_1]$ is obtained after 29 loops ($t_1 = 29$ (s)) using `quadprog.m` in the Matlab™ optimization toolbox. The results are shown in Figure 5.11.

5.2.5 Weather Routing

A weather routing or voyage planning system (VPS) computes the most efficient route using meteorological and oceanographic data, information about the ship's hull and propulsion system and shipping economics to ensure that the vessel reaches port on time. The data from

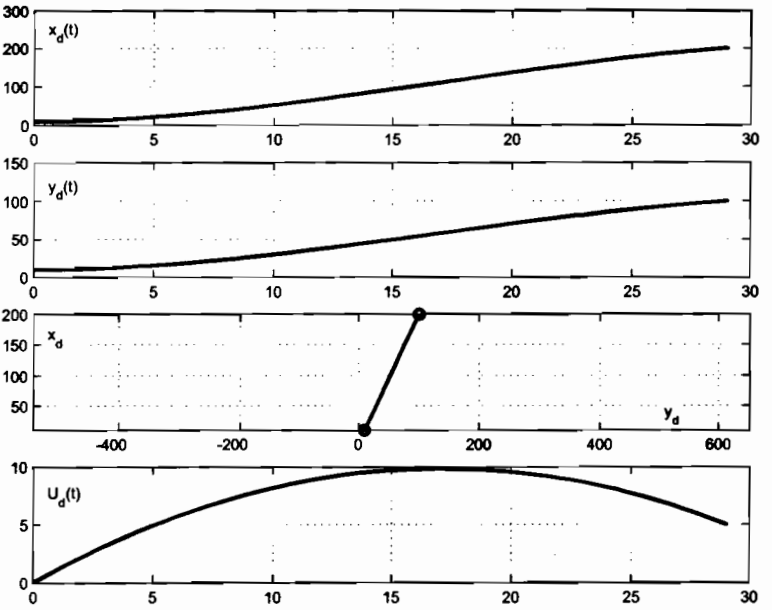


Figure 5.11: The two upper plots shows the cubic polynominals $x_d(t)$ and $y_d(t)$. In the third plot $y_d(t)$ is plotted against $x_d(t)$ while the lower plot is speed $U_d(t)$.

this analysis can be way-points with optimal speed and heading information. The routing software of a modern weather routing system includes features like:

- Surface analysis and forecast models
- Sea state and wind wave models
- Upper air models
- Formation description of low pressure systems
- Hurricanes and tropical weather models
- Ocean current models
- Vessel performance models
- Cargo condition, trim, draft, deck load etc.
- Link to Internet sources for weather data
- Interface to satellite system transmitting weather data
- Optimization of routes based on a fixed estimated time of arrival (ETA)
- Routing of vessels around hazardous weather conditions

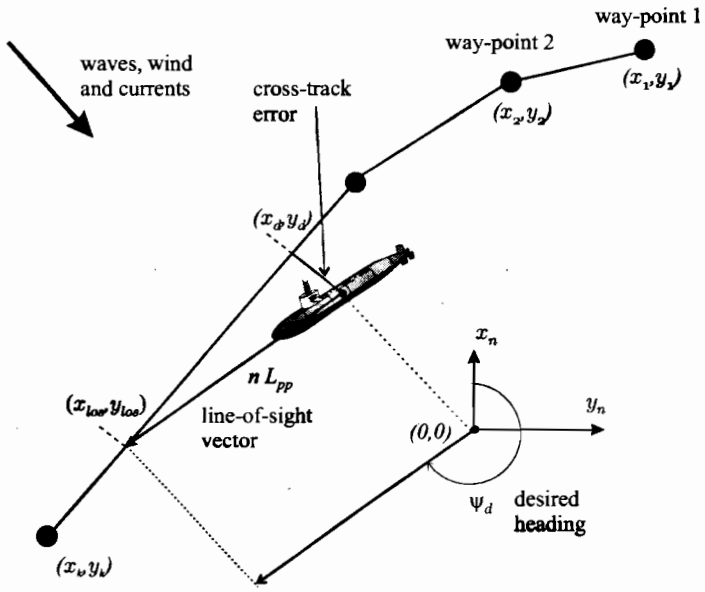


Figure 5.12: Definition of LOS vector.

The optimal route is computed using numerical optimization off-line. This can be done by computer onboard the ship or by a company onshore transmitting the results to the vessel electronically on a 24-hours basis. Several companies offers continuous voyage monitoring with status reports and performance evaluations. This allows for replanning during changing weather conditions. Global weather information is available from several forecast centers like the European Center for Medium Range Weather Forecasting, the U.S. National Weather Service, the U.S. Navy Fleet Numerical Oceanographic Command, the Japanese Meteorological Agency and others.

Some useful references on weather routing of ships are Calvert (1989), Hagiwara (1989), Padadakis and Perakis (1990), Lo (1991), Barbier *et al.* (1994), Lo and McCord (1995), McCord and Smith (1995), and Lo and McCord (1998).

5.3 Line-of-Sight Guidance

An attractive method for path control is *line-of-sight guidance*. A line-of-sight vector from the vessel to the next way-point or a point on the path between two way-points can be computed for heading control. If the vessel is equipped with a course autopilot the angle between the line-of sight vector and the predescribed path can be used as set-point for the course autopilot. This will force the vessel to track the path. Solutions in 2-D and 3-D for surface vessels and underwater vehicles are presented below.

5.3.1 2-Dimensional LOS Guidance System for Surface Vessel

In many applications the *line-of-sight (LOS) vector* is taken as a the vector from the body-fixed origin (x, y) to the next way-point (x_k, y_k). This suggests that the set-point to the course autopilot should be chosen as:

$$\psi_d(t) = \text{atan2}\,(y_k - y(t), x_k - x(t)) \tag{5.90}$$

where (x, y) is the vessel position measurement usually measured with a satellite navigation system. The four quadrant inverse tangent function $\text{atan2}(y, x)$ is used to ensure that:

$$-\pi \leq \text{atan2}(y, x) \leq \pi \tag{5.91}$$

Matlab:
This LOS guidance system can be implemented in Matlab™ as:

```
psi_d = atan2((yk-y),(xk-x))
```

The drawback with a LOS vector pointing to the next way-point is that a way-point located far away from the vessel will result in large cross-track errors if there are transverse wind, current and wave disturbances. Therefore, the LOS vector can be modified as shown in Figure 5.12 where it is defined as the vector from the vessel coordinate origin (x, y) to the intersecting point on the path $(x_{\text{los}}, y_{\text{los}})$ a distance n ship lengths L_{pp} ahead of the vessel. In this case the desired yaw angle can be computed as:

$$\psi_d(t) = \text{atan2}\,(y_{\text{los}} - y(t), x_{\text{los}} - x(t)) \tag{5.92}$$

where the LOS coordinates $(x_{\text{los}}, y_{\text{los}})$ are given by:

$$(y_{\text{los}} - y(t))^2 + (x_{\text{los}} - x(t))^2 = (nL_{pp})^2 \tag{5.93}$$

$$\left(\frac{y_{\text{los}} - y_{k-1}}{x_{\text{los}} - x_{k-1}}\right) = \left(\frac{y_k - y_{k-1}}{x_k - x_{k-1}}\right) = \text{constant} \tag{5.94}$$

The first equation is recognized as the theorem of *Pythagoras* while the second equation states that the slope of the path between the way-points (x_{k-1}, y_{k-1}) and (x_k, y_k) is constant. Hence, the pair $(x_{\text{los}}, y_{\text{los}})$ can be solved from (5.93)–(5.94).

LOS guidance have been applied to surface ships by McGookin *et al.* (2000*b*) and Fossen *et al.* (2003).

Circle of Acceptance

When moving along the path a switching mechanism for selecting the next way-point is needed. Way-point (x_{k+1}, y_{k+1}) can be selected on a basis of whether the vessel lies within a *circle of acceptance* with radius R_0 around way point (x_k, y_k). Moreover if the vehicle positions $(x(t), y(t))$ at time t satisfy:

$$[x_k - x(t)]^2 + [y_k - y(t)]^2 \leq R_0^2 \tag{5.95}$$

the next way point (x_{k+1}, y_{k+1}) should be selected–i.e., k should be incremented to $k = k + 1$. A guideline could be to choose R_0 equal to two ship lengths, that is $R_0 = 2L_{pp}$.

5.3.2 3-Dimensional LOS Guidance System for Underwater Vehicles

It is straightforward to generalize the concepts of LOS guidance to 3-D maneuvering (Healey and Lienard 1993). In this case, the desired yaw angle is chosen as (5.90) or (5.92) while the circle of acceptance is replaced by a *sphere of acceptance:*

$$[x_k - x(t)]^2 + [y_k - y(t)]^2 + [z_k - z(t)]^2 \leq R_0^2 \tag{5.96}$$

taking into account the effect of depth changes, that is variations in the z-coordinate. In 3-D navigation the way-points are defined as (x_k, y_k, z_k) for $i = 1, \ldots, n$ while the position $(x(t), y(t), z(t))$ is assumed to be measured.

5.4 Exercises

Exercise 5.1 *Given the following way-points:*

$$\begin{aligned} wpt.pos.x &= [0, 500, 800, 700, 1000] \\ wpt.pos.y &= [0, 400, 650, 900, 1000] \\ wpt.speed &= [0, 4, 6, 6, 4] \end{aligned}$$

a) Write a Matlab m-file or a Simulink block for path generation based on the data wpt.pos.x and wpt.pos.y using your favorite method. Plot the data and path in an xy-plot.

b) Write a Matlab m-file or a Simulink block transforming the path of problem a) to a time-varying reference trajectory using the speed data wpt.speed as input. Plot the reference trajectories as functions of time.

Exercise 5.2 *Write a Matlab m-file or a Simulink block for LOS guidance using the 2-dimensional method. Simulate the program and plot the results.*

Chapter 6

Estimator Based Navigation Systems

Wave filtering is one of the most important issues to take into account when designing ship control systems (Fossen 1994). It is important that only the slowly-varying disturbances are counteracted by the steering and propulsion systems; the oscillatory motion due to the waves (1st-order wave-induced disturbances) should be prevented from entering the feedback loop. This is done by using *wave filtering* techniques (Balchen *et al.* 1976). A wave filter is usually a model-based observer which separates the position and heading measurements into a *low-frequency* (LF) and a *wave-frequency* (WF) position and heading part; see Figure 6.1.

Definition 6.1 (Wave Filtering)
Wave filtering can be defined as the reconstruction of the LF motion components from noisy measurements of position, heading and in some cases velocity and acceleration by means of a state observer or a filter.

Remark: *If a state observer is applied, estimates of the WF motion components (1st-order wave-induced disturbances) can also be computed.*

Wave filtering is crucial in ship motion control systems since the WF part of the motion should **not** be compensated for by the control system unless wave-induced vibration damping is an issue. This is the case for high-speed craft. If the WF part of the motion enters the feedback loop, this will cause unnecessary usage of the actuators (thrust modulation), and reduce the tracking performance which, again, results in increased fuel consumption.

In this section, model-based wave filtering and observer design using linear wave response models are discussed. This is one of the most important features of a high precision ship control system. The best commercial autopilot and DP systems all have some kind of wave filtering in order to reduce wear and tear on the steering machine, as well as thrust modulation.

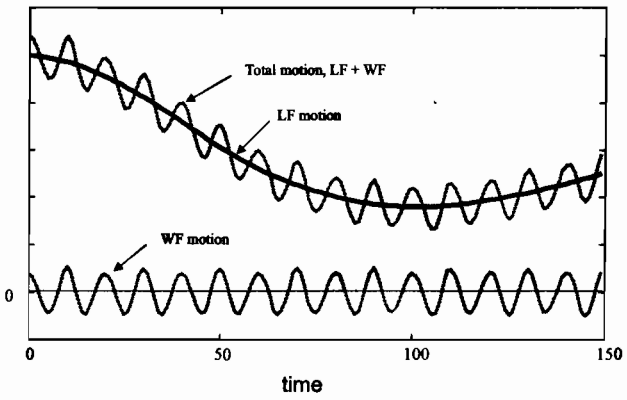


Figure 6.1: The plot shows how the total motion of a marine vessel can be separated into LF and WF motion components.

6.1 Observers for Heading Autopilots

This section shows how observers and wave filters for heading autopilots can be designed. The main sensors in the navigation system are:

- Magnetic and/or gyroscopic compasses measuring ψ
- Yaw rate gyro measuring r

In many commercial systems only the compass is used for feedback control since the yaw rate can be estimated in a state estimator. Techniques for model-based wave filtering using stand-alone compass solutions and integrated compass/yaw gyro solutions are discussed linear wave response models; see Section 4.2.

6.1.1 Magnetic and Gyroscopic Compasses

A compass is the primary device for direction-finding on the surface of the Earth. Compasses may operate on magnetic or gyroscopic principles or by determining the direction of the Sun or a star. We will restrict our discussion to the magnetic and gyroscopic compasses, since these are the primary devices onboard commercial ships and rigs.

Magnetic Compass

The magnetic compass is an old Chinese invention, which probably dates back to 100 AD. Knowledge of the compass as a directional device came to Western Europe sometime in the 12th century and it is today a standard unit in all commercial and navy ships.

A magnetic compass is in fact an extremely simple device (as opposed to a gyroscopic compass). It consists of a small, lightweight magnet balanced on a nearly frictionless pivot point. The magnet is generally called a needle. The magnetic field inside the Earth has its south end at the North Pole and opposite. Hence, the North end of the compass needle points towards the North Pole (opposite magnets attract). The magnetic field of the Earth

is, however, not perfectly aligned along the Earth's rotational axis. It is skewed slightly off center. This skew or bias is called the *declination* and it must be compensated for. It is therefore common to indicate what the declination is on navigational maps. Sensitivity to magnetic variations and declination cause problems in ship navigation. These problems were overcome after the introduction of the gyroscopic compass.

Gyroscopic Compass

The first recorded construction of the gyroscope is usually credited to *C. A. Bohnenberger* in 1810 while the first electrically driven gyroscope was demonstrated in 1890 by *G. M. Hopkins* (see Allensworth 1999, Bennet 1979). A gyroscope is a disk mounted on a base in such a way that the disk can spin freely on its x- and y-axes; that is, the disk will remain in a fixed position in whatever directions the base is moved. A properly mounted gyroscope will always turn to match its plane of rotation with that of the Earth, just as a magnetic compass turns to match the Earth's magnetic field.

The large variations in the magnetic character of ships caused by electrical machinery, weapon systems etc. made the construction of accurate declination or deviation tables for the magnetic compass very difficult. In parallel works, *Dr. H. Anschütz* of Germany and *Elmer Sperry* of the USA worked on a practical application of *Hopkins'* gyroscope. In 1908 Anschütz patented the first North seeking gyrocompass, while Elmer Sperry was granted a patent for his ballistic compass which includes vertical damping three years later.

In 1910, when the Anschütz gyro compass appeared, the problem with magnetic variations in ship navigation was eliminated. However, this compass proved to be quite unsatisfactory during rolling of the ship, since it produced an "intercardinal rolling error". Therefore in 1912 Anschütz redesigned the compass to overcome this defect. One year later, the Sperry compass entered the marked and it became a serious competitor with the Anschütz. Today gyroscopic compasses are produced by a large number of companies for both commercial and navy ships.

6.1.2 Low-Pass and Notch Filtering of Wave Frequency Motions

For wave periods in the interval 5 (s)$< T_0 <$ 20 (s), the dominating wave frequency (modal frequency) f_0 of a wave spectrum will be in the range (see Section 4.2):

$$0.05 < f_0 < 0.2 \qquad \text{(Hz)} \tag{6.1}$$

The circular frequency $\omega_0 = 2\pi f_0$, corresponding to periods $T_0 > 5$ (s) is:

$$\omega_0 < 1.3 \quad \text{(rad/s)} \tag{6.2}$$

Waves within the frequency band (6.1) can be accurately described by 1st- and 2nd-order wave theory. The 1st-order wave disturbances (WF motions) produce large *oscillations* about a *mean* wave force which can be computed from 2nd-order wave theory; see Figure 6.1. The mean wave (drift) force is slowly varying and it is usually compensated for by using *integral action* in the control law, while *wave filtering* must be performed to remove 1st-order components from the feedback loop.

For instance, 1st-order wave disturbances around $f_0 = 0.1$ (Hz) can be close to or outside the control bandwidth of the vessel depending of the vessel considered. For a large oil tanker,

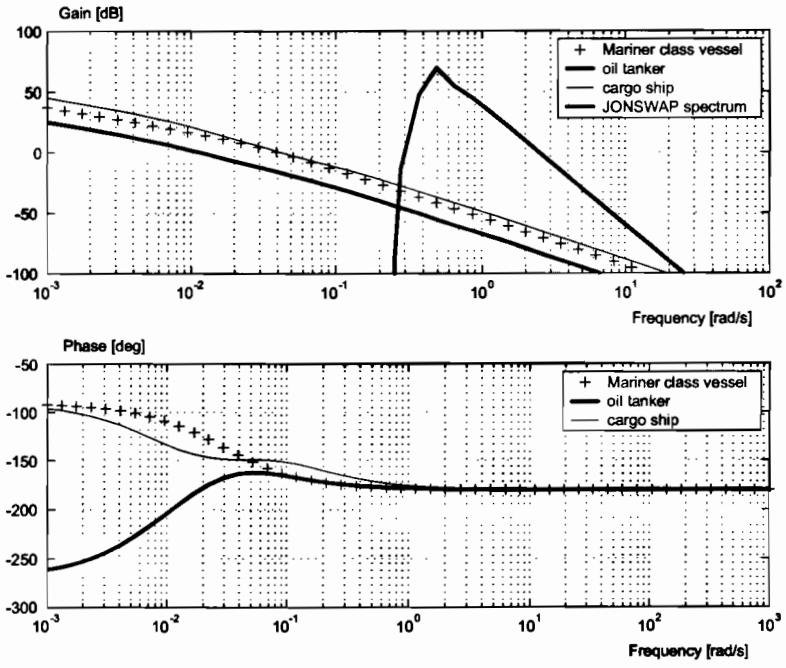


Figure 6.2: Bode plots showing $\frac{\psi}{\delta}(s)$ for three different vessels and the JONSWAP spectrum for $\omega_0 = 0.5$ (rad/s) and $H_s = 5$ (m).

the crossover frequency can be as low as a 0.01 (rad/s) as shown in Figure 6.2, while smaller vessels like cargo ships and the Mariner class vessel, are close to 0.05 (rad/s).

A feedback control system will typically move the bandwidth of these vessels up to 0.1 (rad/s) which still is below the wave spectrum shown in Figure 6.2. However, the wave disturbances will be inside the bandwidth of the servos and actuators of the vessels. Hence, the wave disturbances must be filtered out before feedback is applied in order to avoid unnecessary control action. In other words, we do not want the rudder and thruster actuators of the ship to compensate for the 1st-order wave frequency motion. This is usually referred to as *wave filtering*.

Low-Pass Filtering

For sea states where the WF motion is much higher than the bandwidth ω_b of the controller, that is:

$$\omega_b \ll \omega_e, \quad \left(\omega_e = \left|\omega_0 - \omega_0^2 (U/g)\, \cos\beta\right|\right) \tag{6.3}$$

where ω_e is the frequency of encounter, see (4.59), a low-pass filter can be used to filter out the 1st-order wave disturbances. This is typically the case for large vessels such as oil tankers. In the autopilot case, the design objective can be understood by considering the *measurement equation*:

$$y(s) = \underbrace{h_{\text{ship}}(s)\delta(s)}_{\psi(s)} + \underbrace{h_{\text{wave}}(s)w(s)}_{\psi_w(s)} \tag{6.4}$$

where $y(s)$ is the compass measurement, $w(s)$ is a zero-mean Gaussian white noise process, and $\delta(s)$ is the rudder input. The signal $\psi(s)$ represent the LF motion, while $\psi_w(s)$ is the WF motion. Linear theory suggests that, see (4.43) and (3.264):

$$h_{\text{wave}}(s) = \frac{K_w s}{s^2 + 2\lambda\omega_0 s + \omega_0^2} \tag{6.5}$$

$$h_{\text{ship}}(s) = \frac{K(1 + T_3 s)}{s(1 + T_1 s)(1 + T_2 s)} \tag{6.6}$$

Feedback directly from y will therefore include the WF motion. For a large tanker, proper *wave filtering* can be obtained by using a low-pass filter to produce an estimate of $\psi(s)$ such that:

$$\hat{\psi}(s) = h_{lp}(s)y(s) \tag{6.7}$$

Consequently, the feedback control law δ should be a function of $\hat{\psi}$ and not y in order to avoid 1st-order wave-induced rudder motions.

For instance, a first order low-pass filter with time constant T_f can be designed according to:

$$h_{lp}(s) = \frac{1}{1 + T_f s} \qquad \omega_b < \frac{1}{T_f} < \omega_e \qquad \text{(rad/s)} \tag{6.8}$$

This filter will suppress disturbances over the frequency $1/T_f$. This criterion is obviously hard to satisfy for smaller vessels since ω_b can be close to or even larger than ω_e.

Higher order low-pass filters can be designed by using a *Butterworth filter*, for instance. The n-th order Butterworth filter:

$$h_{lp}(s) = \frac{1}{p(s)} \tag{6.9}$$

is found by solving the Butterworth polynomial:

$$p(s)p(-s) = 1 + (s/j\omega_f)^{2n} \tag{6.10}$$

for $p(s)$. Here n denotes the order of the filter while ω_f is the cut-off frequency. For $n = 1, \ldots, 4$ the solutions are:

$$(n = 1) \qquad h_{lp}(s) = \frac{1}{1 + s/\omega_f}$$

$$(n = 2) \qquad h_{lp}(s) = \frac{\omega_f^2}{s^2 + 2\zeta\omega_f s + \omega_f^2}; \qquad \zeta = \sin(45^o)$$

$$(n = 3) \qquad h_{lp}(s) = \frac{\omega_f^2}{s^2 + 2\zeta\omega_f s + \omega_f^2} \cdot \frac{1}{1 + s/\omega_f}; \qquad \zeta = \sin(30^o)$$

$$(n = 4) \qquad h_{lp}(s) = \prod_{i=1}^{2} \frac{\omega_f^2}{s^2 + 2\zeta_i\omega_f s + \omega_f^2}; \qquad \zeta_1 = \sin(22.5^o), \quad \zeta_2 = \sin(67.5^o)$$

A higher-order low-pass filter implies better disturbance suppression of the price of additional phase lag; see Figure 6.3.

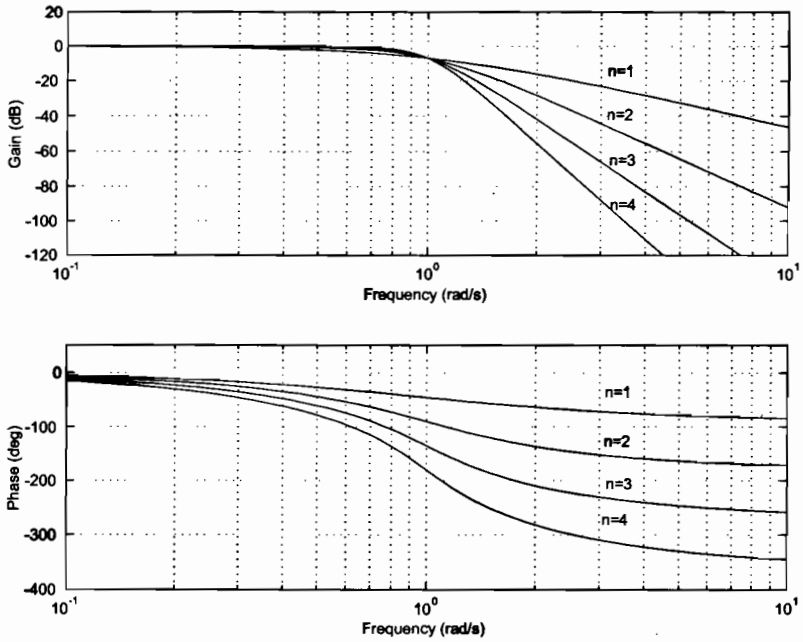


Figure 6.3: Bode plot showing the Butterworth filter for $n = 1, \ldots, 4$ with cut-off frequency $\omega_f = 1.0$ (rad/s).

Low-pass and Notch Filtering

For smaller vessels the bandwidth of the controller ω_b can be close to or within the range $\omega_{\min} < \omega_e < \omega_{\max}$ of the wave spectrum. This problem can be handled by using a low-pass filter in cascade with a notch filter:

$$\hat{\psi}(s) = h_{lp}(s)h_n(s)y(s) \tag{6.11}$$

where:

$$h_n(s) = \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{(s + \omega_n)^2} \tag{6.12}$$

Here $0 < \zeta < 1$ is a design parameter used to control the magnitude of the notch while the notch frequency ω_n should be chosen equal to the peak frequency ω_0 of the spectrum for a vessel at zero speed (dynamic positioning). The low-pass and notch filters are shown in Figure 6.4 for different values of ζ.

For a vessel moving at forward speed U the optimal notch frequency will be:

$$\omega_n = \omega_e \tag{6.13}$$

This frequency can be computed on-line by using a frequency tracker or adaptive filtering techniques.

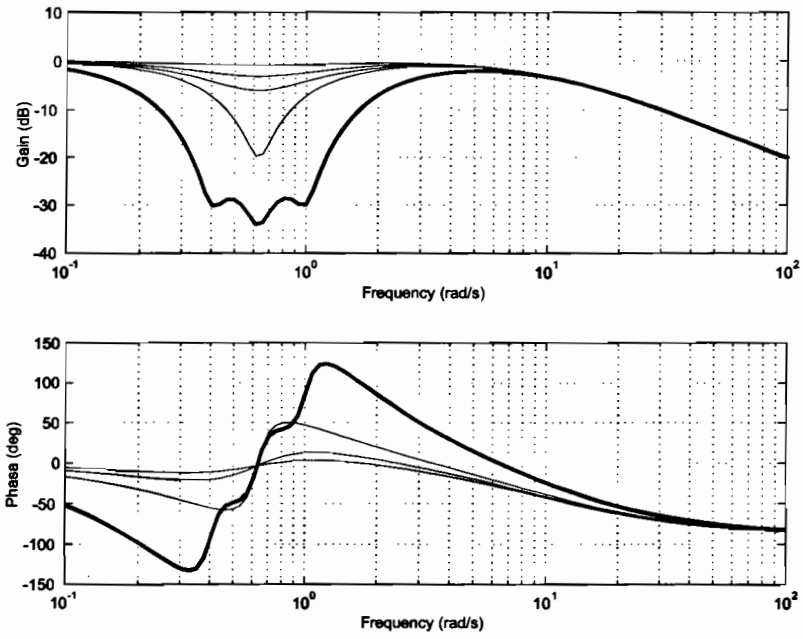


Figure 6.4: Bode plot showing the notch filter for $\zeta \in \{0.1, 0.5, 0.9\}$ and $\omega_0 = 0.63$ (rad/s) in cascade with a low-pass filter with time constant $T_f = 0.1$ (s). The thick line represents three cascaded notch filters at $\omega_1 = 0.4$ (rad/s), $\omega_2 = 0.63$ (rad/s), and $\omega_3 = 1.0$ (rad/s).

Cascaded Notch Filter

Since the estimate of ω_n can be poor and one single-notch filter only covers a small part of the actual frequency range of the wave spectrum, an alternative filter structure consisting of three cascaded notch filters with fixed center frequencies has been suggested; see page 921 of Grimble and Johnson (1989):

$$h_{cn}(s) = \prod_{i=1}^{3} \frac{s^2 + 2\zeta\omega_i s + \omega_i^2}{(s + \omega_i)^2} \tag{6.14}$$

The center frequencies of the notch filters are typically chosen as $\omega_1 = 0.4$ (rad/s), $\omega_2 = 0.63$ (rad/s), and $\omega_3 = 1.0$ (rad/s). This is shown in Figure 6.4. Notice that additional phase lag is introduced when using a cascaded notch filter.

6.1.3 Fixed Gain Observers using only Compass Measurements

An alternative to conventional filtering of wave disturbances is to apply a state estimator (observer). A state estimator can be designed to separate the LF components of the motion from the noisy measurements by using a model of the ship and the wave disturbances. In fact, a model-based wave filter is well suited to separate the LF and WF motions from each

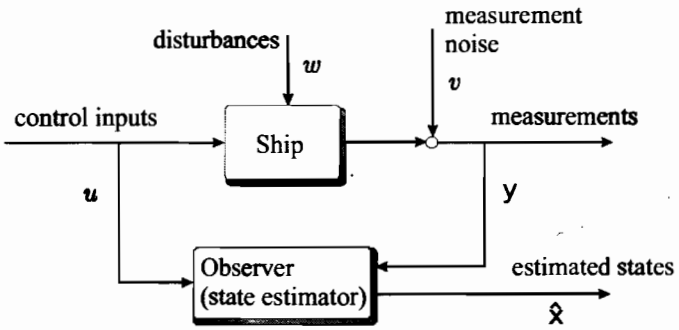


Figure 6.5: Block diagramt showing the system model and the observer.

other, even for vessels where the control bandwidth is close to or higher than the encounter frequency.

Let the system state-space model be written as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{E}\mathbf{w} \tag{6.15}$$

$$y = \mathbf{h}^\top \mathbf{x} + v \tag{6.16}$$

An observer copying this dynamics is (neglecting the zero mean white noise terms w and v):

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \boldsymbol{\gamma} \tag{6.17}$$

$$\hat{y} = \mathbf{h}^\top \hat{\mathbf{x}} \tag{6.18}$$

where $\boldsymbol{\gamma} = \boldsymbol{\gamma}(y, \hat{y})$ is an *injection term* to be constructed such that $\hat{\mathbf{x}} \rightarrow \mathbf{x}$ as $t \rightarrow \infty$. The ultimate goal of the observer is therefore that the unmeasured state vector $\hat{\mathbf{x}}$ should be reconstructed from the measurements u and y; see Figure 6.5.

Luenberger Observer

Assume that $\mathbf{w} = \mathbf{v} = \mathbf{0}$. Defining the estimation error as $\dot{\tilde{\mathbf{x}}} = \mathbf{x} - \hat{\mathbf{x}}$ implies that (6.15) and (6.17) can be written:

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} - \boldsymbol{\gamma} \tag{6.19}$$

A fixed-gain (Luenberger) observer is found by choosing the injection term $\boldsymbol{\gamma}$ as:

$$\boldsymbol{\gamma} = \mathbf{k}\varepsilon, \qquad \varepsilon = y - \hat{y} = \mathbf{h}^\top \tilde{\mathbf{x}} \tag{6.20}$$

where

$$\mathbf{k}^\top = [K_1, ..., K_n] = \text{constant} \tag{6.21}$$

is the observer gain vector. Hence, the error dynamics become:

$$\dot{\tilde{\mathbf{x}}} = (\mathbf{A} - \mathbf{k}\mathbf{h}^\top)\tilde{\mathbf{x}} \tag{6.22}$$

Asymptotical convergence of $\tilde{\mathbf{x}}$ to zero can be obtained for a constant $\mathbf{k}$ if the system $(\mathbf{A}, \mathbf{h}^\top)$ is observable. The following definition is needed to check if the system is observable or not:

Definition 6.2 (Observability)
Consider the linear time-invariant system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{6.23}$$

$$\mathbf{y} = \mathbf{H}\mathbf{x} \tag{6.24}$$

The state and output matrix $(\mathbf{A}, \mathbf{H})$ *must satisfy the* observability condition *to ensure that the state* $\mathbf{x}$ *can be reconstructed from the output* $\mathbf{y}$ *and the input* $\mathbf{u}$. *The* observability condition *requires that the matrix (Gelb* et al. *1988):*

$$\mathcal{O} = [\mathbf{H}^\top \mid \mathbf{A}^\top\mathbf{H}^\top \mid \ldots \mid (\mathbf{A}^\top)^{n-1}\mathbf{H}^\top] \tag{6.25}$$

must be of full column rank such that (at least) a left inverse exists.

Matlab:
If the observability matrix $\mathcal{O}$ is non-singular, the poles of the error dynamics can be placed in the left half-plane by using the Matlab^{TM} function:

```
k = place(A',h,p)'
```

where `p=[p_1,...,p_n]` is a vector describing the desired locations of the observer error poles (must be distinct). Notice that both `k` and `A` are transposed, since the dual problem of the regulator problem is solved.

Examples 6.1–6.2 show how the Luenberger observer can be used in ship control when only compass measurements are available. Emphasis is placed on wave filtering and estimation of the yaw rate.

Example 6.1 (Nomoto Ship Model Exposed to Wind, Waves and Currents)
Let a 1st-order Nomoto model (without loss of generality) be used to describe the LF motion of the ship:

$$\dot{\psi} = r \tag{6.26}$$

$$\dot{r} = -\frac{1}{T}r + \frac{K}{T}(\delta - b) + w_r \tag{6.27}$$

$$\dot{b} = -\frac{1}{T_b}b + w_b \tag{6.28}$$

where the rudder off-set b is modelled as a 1st-order Markov process with $T_b \gg T$. *In the limiting case, that is* $T_b \to \infty$, *this reduces to a Wiener process* ($\dot{b} = w_b$). *The rudder bias*

model is needed to counteract slowly-varying moments on the ship due to wave drift forces, LF wind, and ocean currents. Consequently, the bias term b ensures that $\delta = b$ gives $r = 0$ and $\psi =$ constant in steady-state. The linear wave model (4.52)–(4.53) can be used to model the wave response:

$$\dot{\xi}_w = \psi_w \tag{6.29}$$

$$\dot{\psi}_w = -\omega_0^2 \xi_w - 2\lambda\omega_0 \psi_w + K_w w_w \tag{6.30}$$

The process noise terms, w_r, w_b, and w_w are modelled as zero-mean Gaussian white noise processes. By combining the ship and wave models, the compass measurement equation can be expressed by the sum:

$$y = \psi + \psi_w + v \tag{6.31}$$

where v represents zero-mean Gaussian measurement noise. Notice that the yaw rate r nor the wave states ξ_w and ψ_w are measured. The resulting state-space model for $u=\delta$, $\mathbf{x} = [\xi_w, \psi_w, \psi, r, b]^\top$ and $\mathbf{w} = [w_w, w_r, w_b]^\top$ becomes:

$$\mathbf{A} = \left[\begin{array}{cc|ccc} 0 & 1 & 0 & 0 & 0 \\ -\omega_0^2 & -2\lambda\omega_0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{1}{T} & -\frac{K}{T} \\ 0 & 0 & 0 & 0 & -\frac{1}{T_b} \end{array}\right], \quad \mathbf{b} = \left[\begin{array}{c} 0 \\ 0 \\ \hline 0 \\ \frac{K}{T} \\ 0 \end{array}\right] \tag{6.32}$$

$$\mathbf{E} = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \underbrace{2\lambda\omega_0\sigma}_{K_w} & 0 & 0 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right], \quad \mathbf{h}^\top = [0,1,1,0,0] \tag{6.33}$$

Matlab:
The following example shows how the the Luenberger observer gains of a ship autopilot system can be computed in Matlab™.

Example 6.2 (Luenberger Observer Gains)
It is straightforward to see that the autopilot model with wave frequency, wind, and current model (6.32)–(6.33) is observable from the input δ to the compass measurement y. Let $K = 1, T = 50, \lambda = 0.1, \omega_0 = 1$, and $T_b = 1000$, then:

```
K = 1; T=50; lambda = 0.1; wo =1; Tb = 1000;

A = [ 0 1 0 0 0
      -wo*wo -2*lambda*wo 0 0 0
       0 0 0 1 0
       0 0 0 -1/T -K/T
       0 0 0 0 -1/Tb       ]

h = [0,1,1,0,0]'

n = rank(obsv(A,h'))
```

results in n=5 *corresponding to* $\text{rank}(\mathcal{O}) = 5$. *Hence, the system is observable according to Definition 6.2, implying that all states* r, b, ψ_w, ξ_w *can be reconstructed from a single measurement* $y = \psi + \psi + v$. *The Luenberger filter gains can now be computed by using:*

```
k = place(A',h,[p1,p2,p3,p4,p5])'
```

where `p1,p2,p3,p4,p5` *are the desired closed loop poles of the error dynamics.*

Passivity Based Pole Placement

The observer error dynamics can be reformulated as two subsystems for yaw angle/rudder bias, and yaw rate. Fossen and Strand (1999*b*) have shown that these systems forms a *passive interconnection* (Lozano *et al.* 2000) if the observer gains are chosen according to (see also Section 6.2.4):

$$\mathbf{k} = \begin{bmatrix} -2\omega_0(1-\lambda)/\omega_c \\ 2\omega_0(1-\lambda) \\ \omega_c \\ K_4 \\ K_5 \end{bmatrix} \tag{6.34}$$

where $\omega_c > \omega_0$ is the filter cut-off frequency and the remaining gains must satisfy:

$$0 < 1/T_b < K_5/K_4 < \omega_0 < \omega_c \tag{6.35}$$

The design problem is now reduced to choosing K_4 and K_5 such that the ratio K_5/K_4 satisfies the passive gain constraint (6.35).

Matlab:
The passive wave filter can be simulated using the Simulink model:

```
wavefilter1.mdl
```

in the GNC toolbox; see Figure 6.6 and Example 6.3.

A more detailed analysis of the passive observer is done in Section 6.2.4 which discusses applications to ship positioning in 3 DOF. In fact, the 1 DOF heading autopilot can viewed as a special case of the 3 DOF DP observer.

Example 6.3 (Passive Wave Filtering)
Consider the Mariner class cargo ship with $K = 0.185\ s^{-1}$ *and* $T = T_1 + T_2 - T_3 = 107.3$ *s (Chislett and Strøm-Tejsen 1965*a*). The bias time constant is chosen to be rather large, for instance* $T_b = 100$ *s. The wave response model is modelled by a linear approximation*

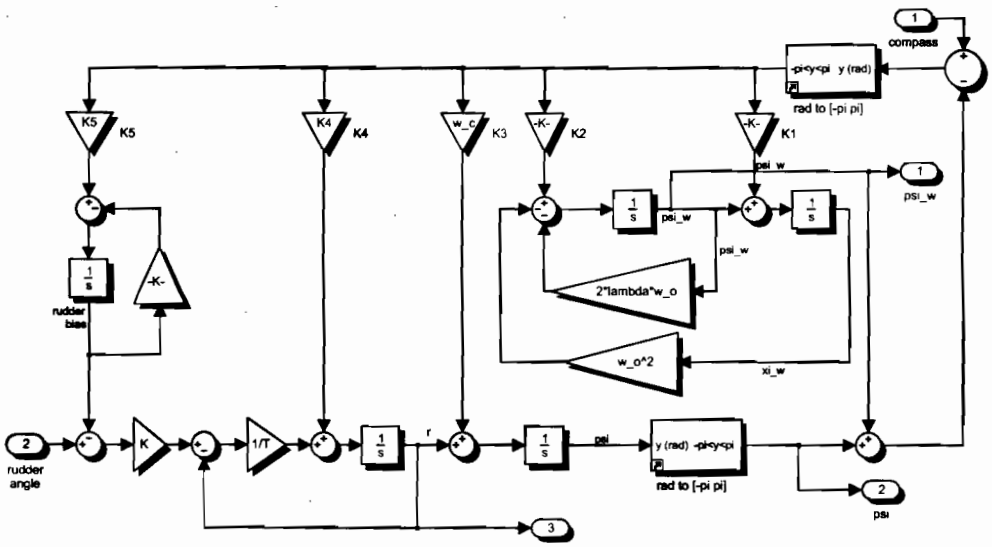


Figure 6.6: Simulink block diagram showing wavefilter1.mdl for generation of ψ, ψ_w, and r from the compass measurements y.

to the JONSWAP spectrum with $\lambda = 0.1$ *and* $\omega_0 = 1.2$ *rad/s; see Section 4.2.2. Hence, (6.32)–(6.33) become:*

$$\mathbf{A} = \left[\begin{array}{cc|ccc} 0 & 1 & 0 & 0 & 0 \\ -1.96 & -0.26 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -0.0093 & -0.0017 \\ 0 & 0 & 0 & 0 & -0.001 \end{array}\right], \mathbf{b} = \left[\begin{array}{c} 0 \\ 0 \\ \hline 0 \\ 0.0017 \\ 0 \end{array}\right] \tag{6.36}$$

$$\mathbf{E} = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ 0.26\,\sigma & 0 & 0 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right], \quad \mathbf{h}^{\mathsf{T}} = [0, 1, 1, 0, 0] \tag{6.37}$$

where $\sigma > 0$ *reflects the sea state. Using passivity as a tool for filter design with cut-off frequency* $\omega_c = 1.1\omega_0$*, yields:*

$$\mathbf{k} = \left[\begin{array}{c} K_1 \\ K_2 \\ \hline K_3 \\ K_4 \\ K_5 \end{array}\right] = \left[\begin{array}{c} -2\omega_0(1-\lambda)/\omega_c \\ 2\omega_0(1-\lambda) \\ \hline \omega_c \\ K_4 \\ K_5 \end{array}\right] = \left[\begin{array}{c} -1.64 \\ 1.80\,\omega_0 \\ \hline 1.10\,\omega_0 \\ K_4 \\ K_5 \end{array}\right] \tag{6.38}$$

This clearly shows that the gains should be adjusted with varying ω_0*. Choosing* $K_4 = 0.1$ *and* $K_5 = 0.01$ *such that* $K_5/K_4 = 0.1$*, yields the transfer functions shown in Figure 6.7.*

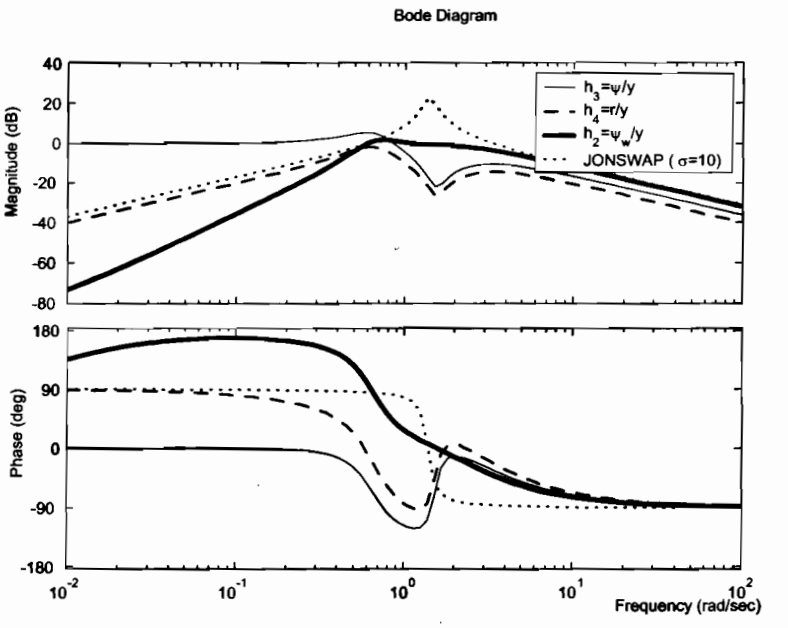


Figure 6.7: Bode plot showing the wave filter transfer functions and the JONSWAP spectrum.

Notice that the notch effect at ω_0 is more than -20 (dB) for $h_3(s)$ and $h_4(s)$ representing the state estimates $\hat{\psi}$ and $\hat{r}$. We also see that high-frequency motion components above ω_c is low-pass filtered. Finally, the transfer function $h_2(s)$ representing reconstruction of the WF motion $\hat{\psi}_w$ filters out signals on the outside of the wave response spectrum, while signals close to ω_0 pass through the filter with unity gain, that is 0 (dB). The poles of the error dynamics are:

$$\begin{aligned} p_1 &= -1.3125 + 0.9793i \\ p_2 &= -1.3125 - 0.9793i \\ p_3 &= -1.2620 \\ p_4 &= -0.0825 \\ p_5 &= -0.0098 \end{aligned}$$

The time-series for $\sigma = 6.25$ are shown in Figure 6.8 where the function wavefilter1.mdl *in the GNC toolbox has been used; see Exercise 6.1 and Section 6.2.4 for a detailed discussion on passivity based wave filtering in surge, sway, and yaw.*

Wave Filter Frequency Analysis

Consider the state estimator:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k}(y - \mathbf{h}^\top \hat{\mathbf{x}}) \tag{6.39}$$

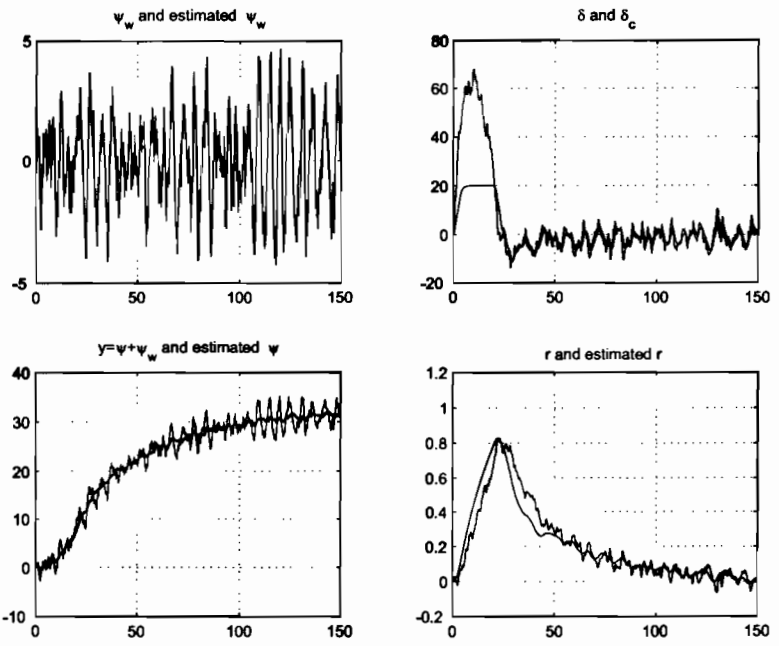


Figure 6.8: Time-series showing the performance of the passive wave filter.

It is then straightforward to show that:

$$\hat{\mathbf{x}}(s) = (s\mathbf{I} - \mathbf{A} + \mathbf{k}\mathbf{h}^\top)^{-1}(\mathbf{k}y(s) + \mathbf{b}u(s)) \tag{6.40}$$

Assume that $u(s) = 0$ (no feedback) such that:

$$\mathbf{h}(s) = [h_1, h_2, h_3, h_4, h_5]^\top = (s\mathbf{I} - \mathbf{A} + \mathbf{k}\mathbf{h}^\top)^{-1}\mathbf{k} \tag{6.41}$$

The states of interest are:

$$\hat{\psi}_w(s) = h_2(s)y(s) \tag{6.42}$$

$$\hat{\psi}(s) = h_3(s)y(s) \tag{6.43}$$

$$\hat{r}(s) = h_4(s)y(s) \tag{6.44}$$

where $h_3(s)$ represents a notch filter with a low-pass filter in cascade:

$$h_3(s) = h_{\text{notch}}(s)\, h_{\text{low pass}}(s) \tag{6.45}$$

The filter $h_4(s)$ also possesses notch filtering in cascade with a second filter representing a limited differentiator for generation of $\hat{r}(s)$ from $y(s)$. Notice that $h_2(s)$ is close to 1 (0 dB) in a band around the wave spectrum while lower and higher frequencies are suppressed in order to reconstruct $\psi_w(s)$ from $y(s)$. This can be seen from the Bode plot in Figure 6.7. These results has also been theoretically verified by Grimble (1978). In this work Grimble showed that the *stationary Kalman filter* for the ship positioning problem will be approximately equivalent to a notch filter in cascade with a second filter, typically a low-pass filter.

The stationary Kalman filter will have the same structure as the fixed gain observer discussed in this section.

When including the feedback term $u(s)$ in the analysis, it is well known that application of an observer is superior to notch and low-pass filtering in cascade, since the observer uses the input $u(s)$ for prediction in addition to filtering the measured output $y(s)$. In fact, this input signal reduces the problems associated with additional phase lag in the filtered signal which is the main problem with most standard filters (low-pass, high-pass, notch etc.). Simulation results verifying these observations have been documented by Grimble (1978).

6.1.4 Kalman Filter Based Wave Filter Design using only Compass Measurements

An alternative solution to the pole-placement technique is to apply a *Kalman filter* (KF) to compute the gain vector $\mathbf{k}$. Kalman filtering (or *optimal state estimation* in sense of minimum variance) allows the user to estimate the state $\mathbf{x}$ of a dynamic system from a noise-contaminated input-output pair $(\mathbf{u}, \mathbf{y})$. The interested reader is advised to consult Brown and Hwang (1998) or Gelb *et al.* (1988) for details on Kalman filter design. Applications specific to the field of guidance and control can be found in Lin (1992).

KF-based wave filtering has been discussed by numerous authors. The interested reader is advised to consult the following references for details; Balchen *et al.* (1976), Balchen *et al.* (1980a, 1980b), Grimble *et al.* (1980a, 1980b), Fung and Grimble (1981), Fung and Grimble (1983), Fotakis *et al.* (1982), Sælid and Jenssen (1983), Sælid *et al.* (1983), Reid *et al.* (1984), Holzhüter and Strauch (1987), Holzhüter (1992), Sørensen *et al.* (1995), Sørensen *et al.* (1996), Fossen *et al.* (2000) and Sørensen *et al.* (2000).

Consider the linear continuous-time system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{6.46}$$

where the process noise $\mathbf{w}$ is assumed to be a zero mean Gaussian white noise process with covariance matrix $\mathbf{Q} = \mathbf{Q}^\top > 0$. In the one-dimensional case $\mathbf{Q}$ corresponds to the squared *standard deviation* σ^2. Furthermore, let the measurement equation (sensor system) be represented by:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{6.47}$$

where the measurement noise $\mathbf{v}$ is assumed to be a zero mean Gaussian white noise process with covariance matrix $\mathbf{R} = \mathbf{R}^\top > 0$.

If the system (6.46)–(6.47) is *observable* (see Definition 6.2), the state vector $\mathbf{x} \in \mathbb{R}^n$ can be reconstructed recursively through the measurement vector $\mathbf{y} \in \mathbb{R}^m$ and the control input vector $\mathbf{u} \in \mathbb{R}^p$; see Figure 6.5. The continuous time KF algorithms are given in Table 6.1.

In the linear case it is computationally advantageous to use the steady-state solution of the KF. This filter will in fact have the same structure as the fixed-gain observers of Section 6.1.3. The only difference is the method for computation of the filter gain matrix.

Continuous-Time Steady-State Kalman Filter

An attractive simplification of the continuous-time Kalman filter is the steady-state solution obtained for the linear time-invariant (LTI) system:

Table 6.1: Continuous-Time Kalman Filter.

Design matrices	$\mathbf{Q}(t)=\mathbf{Q}^\top(t)>0,\ \mathbf{R}(t)=\mathbf{R}^\top(t)>0$ (usually constant)
Initial conditions	$\hat{\mathbf{x}}(0)=\mathbf{x}_0$ $\mathbf{P}(0)=E[(\mathbf{x}(0)-\hat{\mathbf{x}}(0))(\mathbf{x}(0)-\hat{\mathbf{x}}(0))^\top]=\mathbf{P}_0$
Kalman gain matrix	$\mathbf{K}(t)=\mathbf{P}(t)\mathbf{H}^\top(t)\mathbf{R}^{-1}(t)$
State estimate propagation	$\dot{\hat{\mathbf{x}}}(t)=\mathbf{A}(t)\hat{\mathbf{x}}(t)+\mathbf{B}(t)\mathbf{u}(t)+\mathbf{K}(t)[\mathbf{y}(t)-\mathbf{H}(t)\hat{\mathbf{x}}(t)]$
Error covariance propagation	$\dot{\mathbf{P}}(t)=\mathbf{A}(t)\mathbf{P}(t)+\mathbf{P}(t)\mathbf{A}^\top(t)+\mathbf{E}(t)\mathbf{Q}(t)\mathbf{E}^\top(t)$ $-\mathbf{P}(t)\mathbf{H}^\top(t)\mathbf{R}^{-1}(t)\mathbf{H}(t)\mathbf{P}(t),\ \mathbf{P}(t)=\mathbf{P}^\top(t)>0$

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{u}+\mathbf{E}\mathbf{w} \tag{6.48}$$
$$\mathbf{y} = \mathbf{H}\mathbf{x}+\mathbf{v} \tag{6.49}$$

where $\mathbf{w}$ and $\mathbf{v}$ are zero mean Gaussian white noise processes. The steady-state Kalman filter is given by:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}}+\mathbf{B}\mathbf{u}+\mathbf{K}_\infty(\mathbf{y}-\mathbf{H}\hat{\mathbf{x}}) \tag{6.50}$$
$$\mathbf{K}_\infty = \mathbf{P}_\infty\mathbf{H}^\top\mathbf{R}^{-1} \tag{6.51}$$

where $\mathbf{P}_\infty=\mathbf{P}_\infty^\top>0$ is the positive definite solution of the algebraic matrix Riccati equation:

$$\mathbf{A}\mathbf{P}_\infty+\mathbf{P}_\infty\mathbf{A}^\top+\mathbf{E}\mathbf{Q}\mathbf{E}^\top-\mathbf{P}_\infty\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}\mathbf{P}_\infty=\mathbf{0} \tag{6.52}$$

Matlab:
The following example shows how the Kalman filter gains can be computed in Matlab™ for a ship exposed to waves.

Example 6.4 (Continuous Time Steady-State KF for Ship Autopilots)
For the ship-wave system (6.32)–(6.33), the SISO continuous-time state estimator

takes the form:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k}_\infty(y - \mathbf{h}^\top \mathbf{x}) \tag{6.53}$$

where the Kalman filter gain is:

$$\mathbf{k}_\infty = \frac{1}{r}\mathbf{P}_\infty \mathbf{h} \tag{6.54}$$

The covariance matrix $\mathbf{P}_\infty = \mathbf{P}_\infty^\top > 0$ *is given by the ARE:*

$$\mathbf{A}\mathbf{P}_\infty + \mathbf{P}_\infty \mathbf{A}^\top + \mathbf{E}\mathbf{Q}\mathbf{E}^\top - \frac{1}{r}\mathbf{P}_\infty \mathbf{h}\mathbf{h}^\top \mathbf{P}_\infty = \mathbf{0} \tag{6.55}$$

The KF gain $\mathbf{k}_\infty$ *is computed in Matlab*$^{\text{TM}}$ *as:*

```
R = r
Q = diag(q11,q22,q33)

[k,P] = lqe(A,E,h,Q,R)
```

where the tuning of the filter is done by choosing the four design parameters `r, q11, q22, and q33`. *The first of these,* `r`, *represents the compass covariance which can be computed by logging a time-series* `psi`$= \psi(t)$ *of the compass at constant heading. Hence, the Matlab*$^{\text{TM}}$ *command:*

```
r=cov(psi)
```

gives a good estimate of the measurement noise. The disadvantage with the KF approach is that information about the process noise w_1, w_2 *and* w_3 *represented by the weights* `q11, q22` *and* `q33` *are necessary. These three quantities are usually found by trial and error. The variance of the process and measurement noise will vary with each sea state implying that several sets of KF gains must be computed.*

Scaling Procedure for Continuous-Time Steady-State Wave Filter Design

Since the gain and time constants are speed- and thus time-dependent the steady-state solution of the Kalman filter ($\mathbf{P}_\infty$ and $\mathbf{k}_\infty$) varies with the sea state. This suggests that the system matrices should be scaled with respect to forward speed U and ship length L. The model parameters can be made non-dimensional by defining the time and gain constants as $T' = T(U/L)$ and $K' = K(L/U)$, respectively; the wave frequency is scaled according to $\omega_0' = \omega_0(L/U)$. Furthermore, we introduce the time scaling $t' = t(U/L)$ and:

$$\begin{aligned}
\dot{r}' &= \dot{r}(L/U)^2 & r' &= r(L/U) & w_r' &= w_r(L/U)^2 \\
\dot{\psi}' &= \dot{\psi}(L/U) & \psi' &= \psi & w_0' &= w_0(L/U) \\
\dot{\delta}_0' &= \dot{\delta}_0(L/U) & \delta_0' &= \delta_0 & w_w' &= w_w(L/U) \\
\dot{\psi}_w' &= \dot{\psi}_w(L/U) & \psi_w' &= \psi_w & \delta' &= \delta \\
\dot{\xi}_w' &= \dot{\xi}_w & \xi_w' &= \xi_w(U/L) & v' &= v
\end{aligned}$$

Hence, the scaled ship-wave model can be written in vector form as:

$$\dot{\mathbf{x}}' = \mathbf{A}'\mathbf{x}' + \mathbf{b}'u' + \mathbf{E}'\mathbf{w}' \tag{6.56}$$

with speed-invariant quantities:

$$\mathbf{A}' = \left[\begin{array}{cc|ccc} 0 & 1 & 0 & 0 & 0 \\ -(\omega_0')^2 & -2\lambda\omega_0' & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{1}{T'} & -\frac{K'}{T'} \\ 0 & 0 & 0 & 0 & -\frac{1}{T_\delta'} \end{array}\right], \quad \mathbf{b}' = \left[\begin{array}{c} 0 \\ 0 \\ \hline 0 \\ \frac{K'}{T'} \\ 0 \end{array}\right] \tag{6.57}$$

$$\mathbf{E}' = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ 2\lambda\omega_0'\sigma' & 0 & 0 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right], \qquad \mathbf{h}'=[0,1,1,0,0]^\top \tag{6.58}$$

Discrete-Time Kalman Filter Design

The discrete-time Kalman filter is defined in terms of the discretized system model:

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Delta}\mathbf{u}(k) + \mathbf{\Gamma}\mathbf{w}(k) \tag{6.59}$$

$$\mathbf{y}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{v}(k) \tag{6.60}$$

where

$$\mathbf{\Phi} = \exp(\mathbf{A}h) \approx \mathbf{I} + \mathbf{A}h + \frac{1}{2}(\mathbf{A}h)^2 + \ldots + \frac{1}{N!}(\mathbf{A}h)^N \tag{6.61}$$

$$\mathbf{\Delta} = \mathbf{A}^{-1}(\mathbf{\Phi} - \mathbf{I})\mathbf{B}, \qquad \mathbf{\Gamma} = \mathbf{A}^{-1}(\mathbf{\Phi} - \mathbf{I})\mathbf{E} \tag{6.62}$$

and h is the sampling time.

Matlab:
The discretized system matrices can be computed as:

```
[PHI,DELTA]=c2d(A,B,h)
[PHI,GAMMA]=c2d(A,E,h)
```

where `PHI`=$\mathbf{\Phi}$, `DELTA`=$\mathbf{\Delta}$ and `GAMMA`=$\mathbf{\Gamma}$. Notice that Euler integration implies choosing $N = 1$, that is $\mathbf{\Phi}(k) = \mathbf{I} + \mathbf{A}h$; see Appendix B.1.

The linear discrete-time Kalman filter algorithm is given in Table 6.2. This algorithm, however, requires that the state estimation error covariance matrix $\hat{\mathbf{P}}(k) = \hat{\mathbf{P}}(k)^\top > 0$ is computed on-line. Since the matrix is symmetrical, the number of differential equations will be $n(n+1)/2$ for $\mathbf{P}(k) \in \mathbb{R}^{n\times n}$. In addition there are n state estimates corresponding to $\hat{\mathbf{x}}(k)$.

Table 6.2: Discrete-Time Kalman Filter.

Design matrices	$\mathbf{Q}(k)=\mathbf{Q}^\top(k)>0$, $\mathbf{R}(k)=\mathbf{R}^\top(k)>0$ (usually constant)
Initial conditions	$\overline{\mathbf{x}}(0)=\mathbf{x}_0$ $\overline{\mathbf{P}}(0)=E[(\mathbf{x}(0)-\hat{\mathbf{x}}(0))(\mathbf{x}(0)-\hat{\mathbf{x}}(0))^\top]=\mathbf{P}_0$
Kalman gain matrix	$\mathbf{K}(k)=\overline{\mathbf{P}}(k)\mathbf{H}^\top(k)\,[\mathbf{H}(k)\overline{\mathbf{P}}(k)\mathbf{H}^\top(k)+\mathbf{R}(k)]^{-1}$
State estimate update	$\hat{\mathbf{x}}(k)=\overline{\mathbf{x}}(k)+\mathbf{K}(k)\,[\mathbf{y}(k)-\mathbf{H}(k)\overline{\mathbf{x}}(k)]$
Error covariance update	$\hat{\mathbf{P}}(k)=[\mathbf{I}-\mathbf{K}(k)\mathbf{H}(k)]\,\overline{\mathbf{P}}(k)\,[\mathbf{I}-\mathbf{K}(k)\mathbf{H}(k)]^\top$ $+\mathbf{K}(k)\mathbf{R}(k)\mathbf{K}^\top(k)$, $\quad\hat{\mathbf{P}}(k)=\hat{\mathbf{P}}(k)^\top>0$
State estimate propagation	$\overline{\mathbf{x}}(k+1)=\mathbf{\Phi}(k)\hat{\mathbf{x}}(k)+\mathbf{\Delta}(k)\mathbf{u}(k)$
Error covariance propagation	$\overline{\mathbf{P}}(k+1)=\mathbf{\Phi}(k)\hat{\mathbf{P}}(k)\mathbf{\Phi}^\top(k)+\mathbf{\Gamma}(k)\mathbf{Q}(k)\mathbf{\Gamma}^\top(k)$

The main problem in the realization of the state estimator is that the parameters K, T, ω_0, and λ are unknown. Satisfactory values for the non-dimensional ship parameters K' and T' can usually be found from maneuvering trials or by parameter estimation. Holzhüter (1992) claims that the damping coefficient in the wave model can be chosen rather arbitrarily as long as it is low (typically $\lambda = 0.01$–0.1), whereas the wave frequency ω_0 can be treated as a tunable or gain scheduled parameter. In some cases it can be advantageous to estimate ω_0 on-line by applying a frequency tracker or adaptive control theory (Strand and Fossen 1999).

6.1.5 Observer and Wave Filter Design using both Compass and Rate Measurements

In this section the designs of the previous sections are modified to include a rate gyro in addition to the compass. This is advantageous since the gyro can be integrated with the compass in an optimal manner resulting in less variance and better accuracy of the state estimates. One simple way to do this is to treat the gyro measurements as an input to the system model.

The system model becomes:

$$\dot{\psi} = u_{\text{gyro}} + b \tag{6.63}$$

$$\dot{b} = w_b \tag{6.64}$$

where b denotes the gyro bias, w_b is Gaussian white noise and u_{gyro} is the *rate gyro measurement*. The WF model is similar to (6.29)–(6.30). This model will give proper wave filtering of the state ψ. However, the estimate of r is not wave filtered, since this signal is taken directly from the gyro measurement u_{gyro}. This can be solved by filtering u_{gyro} with a notch

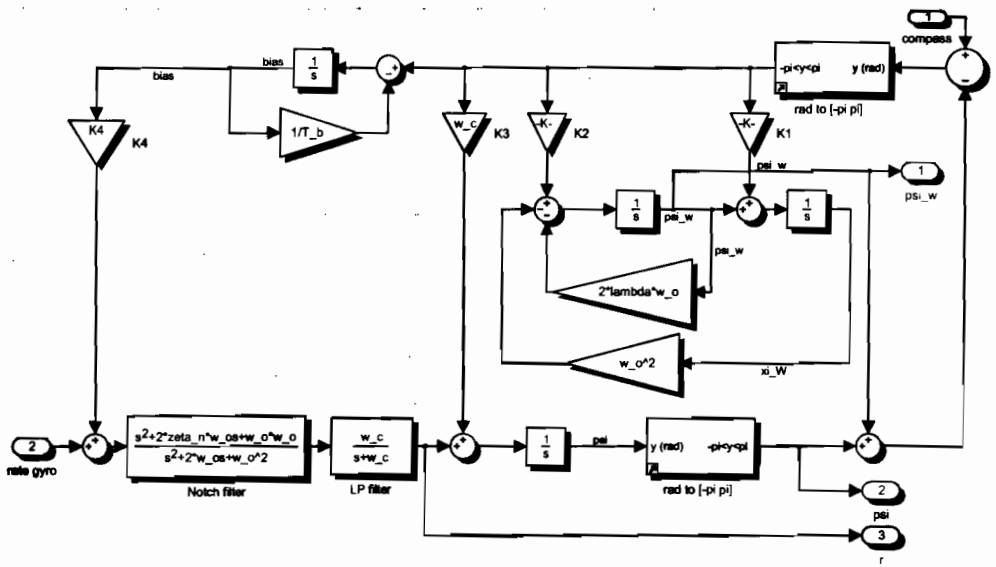


Figure 6.9: Simulink block diagram showing wavefilter2.mdl for filtering of compass ψ and gyro rate r measurements.

filter $h_{\text{notch}}(s)$ and a low-pass filter $h_{\text{lp}}(s)$ to the cost of some phase lag (see Figure 6.9):

$$u_f = h_{\text{notch}}(s)\, h_{\text{lp}}(s)\, u_{\text{gyro}} \tag{6.65}$$

The resulting model becomes:

$$\dot{\hat{\xi}}_w = \hat{\psi}_w + K_1\,\varepsilon \tag{6.66}$$

$$\dot{\hat{\psi}}_w = -\omega_0^2\,\hat{\xi}_w - 2\lambda\omega_0\,\hat{\psi}_w + K_2\,\varepsilon \tag{6.67}$$

$$\dot{\hat{\psi}} = u_f + \hat{b} + K_3\,\varepsilon \tag{6.68}$$

$$\dot{\hat{b}} = -\frac{1}{T_b}\hat{b} + K_4\,\varepsilon \tag{6.69}$$

where $\varepsilon = y - \hat{\psi} - \hat{\psi}_w$ and $T_b \gg 0$. Notice that the gyro bias must be estimated on-line since it will vary with temperature and possible scale factor/misalignment errors when mounted onboard the ship. This is a slowly-varying process so the gain K_4 can be chosen quite small reflecting a large bias time constant. If passivity-based pole placement (6.34) is used, K_1, K_2, and K_3 become:

$$K_1 = -2\omega_0(1-\lambda)/\omega_c, \quad K_2 = 2\omega_0(1-\lambda), \quad K_3 = \omega_c \tag{6.70}$$

Alternatively, the KF algorithm can be used to compute the gains.

Matlab:
The observer with compass and rate measurements can be simulated using the Simulink model:

```
wavefilter2.mdl
```

in the GNC toolbox as shown in Figure 6.9.

The performance of the wave filter is demonstrated in Example 6.5 while the design of the autopilot is discussed in Example 8.10 in Section 8.4.2.

Example 6.5 (Wave Filter Design for a Car Carrier: Experimental Results)
The wave filter has been tested on a scale model of MV Autoprestige of the United European Car Carriers (UECC); see Figure 8.16 in Example 8.10. This was done by using the Matlab/Simulink Real-Time Workshop (http://www.mathworks.com) and RT-Lab from Opal (http://www.opal-rt.com). The C-code was automatically generated under Matlab and downloaded to the target PC onboard the model ship using a wireless Ethernet connection and the QNX real-time operating system (http://www.qnx.com).

The maneuvering test were performed in the Ocean Basin at MARINTEK in Trondheim April 2001. The wave filter block diagram is shown in Figure 6.9. The experimental results and performance of the wave filter during a course-keeping maneuver are shown in Figure 6.10. It is seen that the WF motion components are quite well removed from the estimate of ψ resulting in good course-keeping capabilities. We also notice that the estimate of ψ_w is quite good, while r could be slightly improved by changing the observer gains. In the experiment the significant wave height was $H_s = 1.3$ (m) in full scale, while the frequency of encounter was $\omega_e = 1.07$ (rad/s). The cruise speed of the model was $U = 2.3$ (m/s).

Other techniques for the integration of compass and rate measurements are described in Lindegaard (2003).

6.2 Observers for Dynamic Positioning Systems

In this section both the *Kalman filter* and a *nonlinear passive observer* are presented for DP applications. Both observers include wave filtering, bias state estimation, reconstruction of the LF motion components, and estimates of the non-measured body velocities. Positioning feedback systems are described more closely in Chapter 11. Before observer design is discussed, a general introduction to navigation systems is given.

6.2.1 Navigation Systems

For conventional ships, only position and heading measurements are available. Several position measurement systems are commercially available, such as local *hydroaccustic positioning reference* (HPR) systems and *global navigation satellite systems* (GNSS).

The two commercial available GNSS are the U.S. system *Navstar GPS* (see Hofmann-Wellenhof *et al.* 1994, Parkinson and Spilker 1995), and the Russian *Global Orbiting Navigation Satellite System* (GLONASS); see Kayton and Fried (1997) and Leick (1995), for instance.

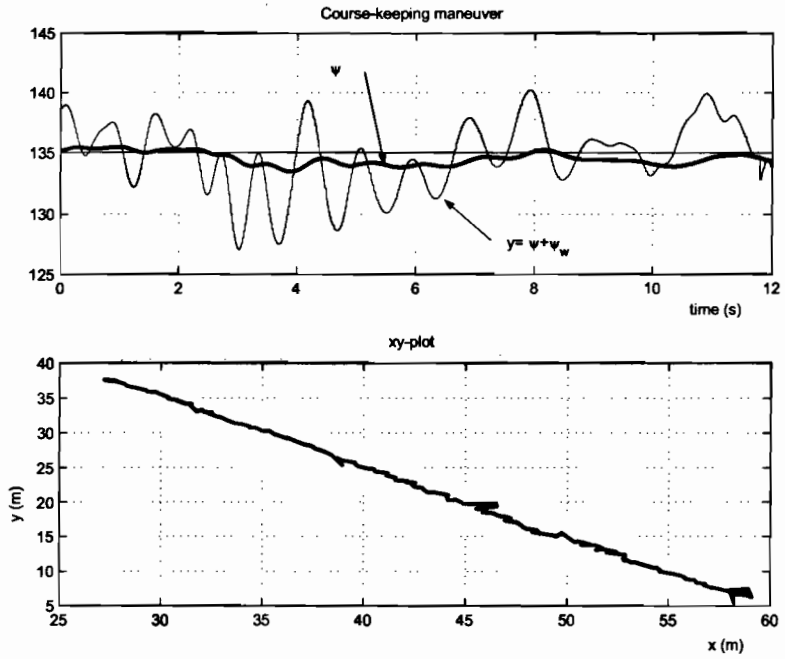


Figure 6.10: The upper plot shows reconstruction of ψ (thick line) from the compass measurement $y = \psi + \psi_w$ (thin line) and gyro measurement r using a wave filter. The lower plot show the xy-coordinates during course-keeping ($\psi_d = 135$ deg) in the Ocean Basin at Marintek. Courtesy UECC and MARINTEK.

In addition to this, a European satellite navigation system, *Galileo*, is currently under construction. *Galileo* is an initiative of the European Union, in collaboration with the European Space Agency (ESA) and European Industry, to launch a European financed global satellite navigation system under civilian control. ESA will begin launching Galileo satellites in 2004 with a full constellation ready to begin operational service in 2007.

Navstar GPS

Navstar GPS is a dual-use satellite based system that provides positioning and timing data to users worldwide. The foundation for GPS was laid in the late sixties with studies made by the U.S. Air Force and Navy, and the Joint Program Office that managed the development of Navstar GPS was formed in 1973. The first operational satellite was launched in 1978, with the last one in 1994. The number of satellites remained relatively low until 1989/1990, when a number of Block II satellites were launched. Initial operational capability was declared at the end of 1993, and full operational capability was declared by the end of 1994.

GPS was showcased in the 1991 Gulf War, and that brought attention to the capability that this system could provide. Since then GPS has become an invaluable resource in numerous applications in the air, on land, and at sea. Although developed for military purposes, GPS has been great success in the civilian community. The removal of *Selective Availability* (SA) on the 1st of May 2000, has made high accuracy navigation available at low cost. SA was a

combination of methods used by the U.S. Department of Defense for deliberately degrading GPS accuracy for non–U.S. military users.

Differential and Augmented GPS

Since the accuracy of the GPS satellite navigation system was degraded for civilian users until the year 2000, *differential* global positioning system (DGPS) was developed for accurate navigation. The main idea of a *differential* GPS system is that a fixed receiver located, for example *on shore* with a known position, is used to calculate the GPS position errors. The position errors are then transmitted to the GPS-receiver on board the ship and used as corrections to the actual ship position. In a DGPS-system the horizontal positioning errors are squeezed down to less than 1 (m), which is the typical accuracy of a ship positioning system today (Hofmann-Wellenhof *et al.* 1994).

Another form for DGPS is local-area differential GPS (LADGPS) where the user's GPS receiver also receives real-time pseudorange and carrier phase corrections from a local reference receiver located within line of sight. This results in a highly accurate position estimate in a local region.

Similarly in a wide-area DGPS (WADGPS) the user's GPS receiver receives corrections determined from a network of reference stations distributed over a wide geographical area.

Carrier Differential GPS

A GPS receiver in lock is able to track the phase shift of the carrier, and output the fractional phase measurement at each epoch. However, the overall phase measurement contains an unknown number of carrier cycles. This is called the integer ambiguity (N). This ambiguity exists because the receiver merely begins counting carrier cycles from the time a satellite signal is placed in active track. The precision of the phase measurement is about 0.01 cycles ($\approx 0.01 * 19$ cm $= 1.9$ mm), and if N is determined, it allows for highly accurate position measurements.

Ambiguity resolution is a very active research area, and there exist several receivers (known as *real time kinematic* (RTK) receivers) on the market today that utilize carrier measurements to achieve position accuracy in the order of a few centimeters. These position measurements are, however, not as robust as GPS and DGPS. The baseline length (distance to the reference receiver) is also restricted to about 50 km, due to atmospheric errors causing problems for the ambiguity search algorithms. New wide area augmentation systems that implements WADGPS have the potential to increase this range significantly, due to improved atmospheric modeling. In fact, 10 cm level accuracies have been reported for up to 750 km baselines.

Satellite Overlay and Inmarsat Civil Navigation

Inmarsat is an 80-nation consortium also known as the International Mobile Satellite Organization. Inmarsat was founded in 1979 to provide maritime mobile services using four geostationary satellites providing global coverage. These satellites broadcasts data to many ground station networks worldwide. This includes the service of the Inmarsat civil navigation geostationary satellite overlay which complements the GPS and GLONASS satellite systems resulting in improved accuracy, reliability and coverage.

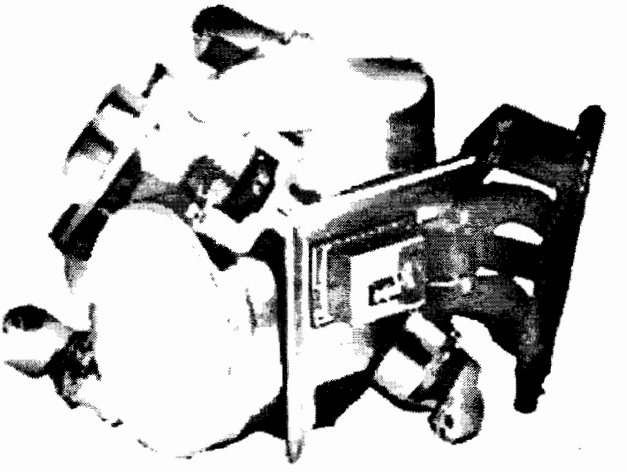


Figure 6.11: The Litton LN-100 Inertial Sensor Assembly. Courtesy to Litton Guidance and Control Systems.

6.2.2 Inertial Measurement Systems

Today inertial measurement technology is available for commercial users thanks to a significant reduction in price the last decade. As a consequence of this, low cost inertial sensors can be integrated with satellite navigation system using a conventional Kalman filter or a nonlinear state observer. Methods for this with focus on marine applications are presented in Vik and Fossen (2002) while GPS/INS integration techniques in more general are discussed by Farrell and Barth (1998), Titterton and Weston (1997), and Grewal *et al.* (2001) to mention some.

Figure 6.11 shows an ISA (*Inertial Sensor Assembly*), which is a cluster of three gyros and three accelerometers that measure angular velocity and linear acceleration, respectively.

An IMU (*Inertial Measurement Unit*) consists of an ISA, hardware to interface the ISA, and low level software that performs down-sampling, temperature calibration, and vibration (sculling and coning) compensation. Figure 6.12 shows the Litton LN-200 IMU.

An inertial navigation system (INS) consists of a measurement part (IMU), and software (state observer) that computes position, velocity and attitude from the measurements. When integrating the angular velocity (gyro) and linear acceleration (accelerometers) drift must be prevented. This is obtained by using a GNSS as reference for position and the resulting system is known a *strapdown inertial navigation system.*

The key components of the IMU are described below:

Gyroscopes

The classic gyro is a spinning wheel that utilizes conservation of momentum to detect rotation, and belongs naturally in a gimballed system. For strapdown applications, optical gyros like ring laser gyros (RLG) and fiber optic gyros (FOG) have been used for some time, and are also expected to be the standard for high accuracy strapdown inertial systems for foreseeable

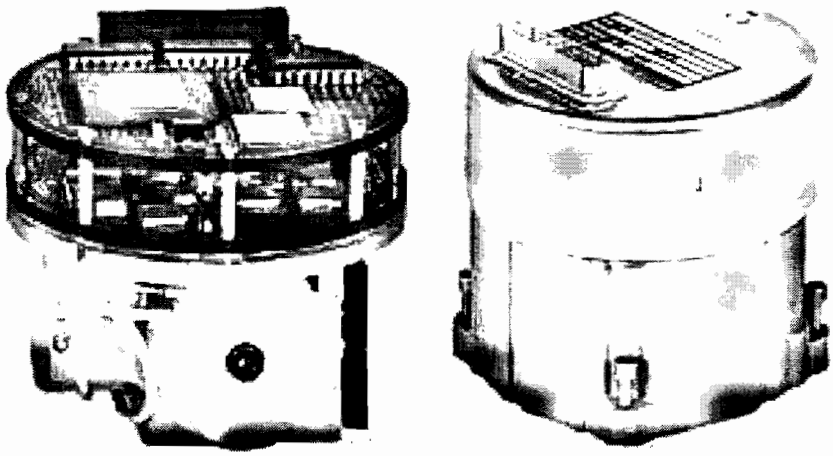


Figure 6.12: The Litton LN-200 Inertial Measurement Unit. The ISA can be seen at the bottom of the figure to the left. The picture to the right shows the IMU with enclosure. The Litton LN-200 IMU. Courtesy to Litton Guidance and Control Systems.

Table 6.3: Performance characteristics for different types of gyros.

Parameter	RLG	FOG	MEMS
Input range (deg/sec)	> 1000	> 1000	> 1000
Bias (deg/hr)	0.001-10	0.01-50	10-3600
Scale-factor error (%)	0.0001-0.01	0.0002-0.5	0.5-2
Bandwidth (Hz)	500	>200	>100

future. For low and medium cost applications, gyros based on micro-electric-mechanical-systems (MEMS) are expected to be dominant (Barbour and Schmidt 1998).

Table 6.3 shows the range of performance available for optical and MEMS based gyros.

Accelerometers

There are several different types of accelerometer. Two of these are *mechanical* and *vibratory accelerometers.*

The mechanical accelerometer can be a *pendulum,* which in its simplest form is based on Newton's second law of motion:

$$F = ma$$

A force F acting on a body of mass m causes the body to accelerate with respect to inertial space. When the case of the instrument is subjected to an acceleration along its sensitive axis, the proof mass tends to resist the change in movement due to its own inertia. As a result, the mass is displaced with respect to the case. Under steady state conditions the force acting on the mass will be balanced by the tension of the spring. The extension of the spring then

Table 6.4: Performance characteristics for different types of accelerometers.

Parameter	Closed loop pendulum	Vibrating quartz	Silicon
Input range (g)	±100	±200	±100
Bias (mg)	0.1-10	0.1-1	<25
Scale-factor error (%)	0.1	0.01	0.5-2
Bandwidth (Hz)	400	100	400
Threshold (μg)	10	<10	1-10

provides a measure of the force, which is proportional to the acceleration.

The *vibratory accelerometers* are usually based on measurement of frequency shifts due to increased or decreased tension in a string. The operation is similar to that of a violin. When a violin string is tightened, the frequency goes up. Similarly, when the accelerometer proof mass attached to a quartz beam is loaded, the frequency of the quartz beam increases. The difference in frequency is measured, and is proportional to the applied acceleration. In addition to quartz technology, vibrating beam accelerometers using silicon are also being developed.

Table 6.4 summarizes key performance parameters for different types of accelerometers. It should be noted that closed loop precision accelerometers can have substantially higher performance, with biases of a few micro-g and scale-factor errors of 0.00001 %. These accelerometers are, however, not normally designed to measure accelerations of ±100 g.

6.2.3 Kalman Filter for Velocity and Wave Frequency Motion

Dynamic positioning (DP) systems have been commercially available for marine vessels since the 1960's. The first DP systems were designed using conventional PID controllers in cascade with low pass and/or notch filters to suppress the wave-induced motion components. From the middle of the 1970's more advanced filtering techniques were available thanks to the Kalman filter (Kalman 1960). This motivated Balchen and coauthors to define wave filtering in terms of linear optimal estimation theory; see Balchen *et al.* (1976, 1980a, 1980b). A similar design technique has been proposed by Grimble *et al.* (1979, 1980a).

Introduction

Filtering and state estimation are important features of a DP system. In many cases, measurements of the vessel velocities are not available. Hence, estimates of the velocities must be computed from noisy position and heading measurements through a state observer. Unfortunately, the position and heading measurements are corrupted with colored noise due to wind, waves, and ocean currents as well as sensor noise. Only the slowly-varying disturbances should be counteracted by the propulsion system, whereas the oscillatory motion due to the waves (1st-order wave disturbances) should not enter the feedback loop. This is done by using so-called wave filtering techniques, which separates the position and heading measurements into a LF and WF position and heading estimate; see Definition 6.1.

In many DP systems the wave filtering and state estimation problems are solved by using

the linear or extended *Kalman filter*. The major drawback of this approach is that the kinematic equations of motions must be linearized about a set of predefined constant yaw angles, typically 36 operating points in steps of 10 degrees, to cover the whole heading envelope. For each of these linearized models, optimal Kalman filter gains are computed such that the gains can be modified on-line by using gain-scheduling techniques. An alternative approach to the linearization procedure is to use *vessel parallel coordinates* to avoid look-up tables (see Section 3.3.2).

This chapter presents a nonlinear passive observer covering the whole state space using one set of gains (see Section 6.2.4).

Vessel Kinematics and Dynamics

The following DP model is considered (Fossen and Strand 1999*b*):

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{6.71}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} = \boldsymbol{\tau} + \mathbf{R}^{\top}(\psi)\mathbf{b} + \mathbf{w}_3 \tag{6.72}$$

where $\boldsymbol{\eta} = [n, e, \psi]^{\top}$, $\boldsymbol{\nu} = [u, v, r]^{\top}$, $\mathbf{b} \in \mathbb{R}^3$ is a vector of bias terms and $\mathbf{w}_3 \in \mathbb{R}^3$ is a vector of zero-mean Gaussian white noise processes. The different quantities in this model are described more closely in Section 3.5.1

1st-Order Wave Response Model

As shown in Section 4.2.2 three linear wave response models in surge, sway, and yaw can be written as:

$$\dot{\boldsymbol{\xi}} = \mathbf{A}_w\boldsymbol{\xi} + \mathbf{E}_w\mathbf{w}_1 \tag{6.73}$$

$$\boldsymbol{\eta}_w = \mathbf{C}_w\boldsymbol{\xi} \tag{6.74}$$

where $\boldsymbol{\xi} \in \mathbb{R}^6$ is the state vector, $\mathbf{w}_1 \in \mathbb{R}^3$ is a vector of zero-mean Gaussian white noise, and $\mathbf{A}_w \in \mathbb{R}^{6\times6}$, $\mathbf{E}_w \in \mathbb{R}^{6\times3}$ and $\mathbf{C}_w \in \mathbb{R}^{3\times6}$ are constant matrices of appropriate dimensions.

Bias Modeling (Slowly-Varying Environmental Disturbances)

It is assumed that the bias forces in surge and sway, and the yaw moment vary slowly. This can be modelled as a *Wiener process* (random walk):

$$\dot{\mathbf{b}} = \mathbf{w}_2 \tag{6.75}$$

where $\mathbf{w}_2 \in \mathbb{R}^3$ is a vector of zero-mean Gaussian white noise. An alternative model is the *1st-order Markov* model:

$$\dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b} + \mathbf{w}_2 \tag{6.76}$$

where $\mathbf{T} = \text{diag}\{T_1, T_2, T_3\} \in \mathbb{R}^{3\times3}$ is a user specified diagonal matrix of positive bias time constants. These models can be used to describe slowly-varying environmental forces and moments due to:

- 2nd-order mean and slowly-varying wave loads
- ocean currents
- wind

Measurement Model

The position and yaw angle measurements are generated by using the principle of linear superposition, that is the 1st-order wave-induced motion component $\boldsymbol{\eta}_w = [n_w, e_w, \psi_w]^\top$ is added to the LF motion components of the vessel given by $\boldsymbol{\eta} = [n, e, \psi]^\top$; see Figure 6.1. Hence, the position and heading measurement equation can be written:

$$\mathbf{y} = \boldsymbol{\eta} + \boldsymbol{\eta}_w + \mathbf{v} \tag{6.77}$$

This equation reflects that the velocity vector $\boldsymbol{\nu}$ and bias $\mathbf{b}$ are treated as unknown states to be estimated on-line from $\mathbf{u}$ and $\mathbf{y}$.

Resulting DP Observer Model

The resulting DP observer model is:

$$\dot{\boldsymbol{\xi}} = \mathbf{A}_w \boldsymbol{\xi} + \mathbf{E}_w \mathbf{w}_1 \tag{6.78}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{6.79}$$

$$\dot{\mathbf{b}} = \mathbf{w}_2 \qquad (\text{alternatively } \dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b} + \mathbf{w}_2) \tag{6.80}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} = -\mathbf{D}\boldsymbol{\nu} + \mathbf{R}^\top(\psi)\mathbf{b} + \boldsymbol{\tau} + \mathbf{w}_3 \tag{6.81}$$

$$\mathbf{y} = \boldsymbol{\eta} + \mathbf{C}_w \boldsymbol{\xi} + \mathbf{v} \tag{6.82}$$

Discrete-Time Extended Kalman Filter (EKF) Design

When designing the EKF it is convenient to write the DP observer model (6.78)–(6.82) in the following form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{6.83}$$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{6.84}$$

where $\mathbf{x} = [\boldsymbol{\xi}^\top, \boldsymbol{\eta}^\top, \mathbf{b}^\top, \boldsymbol{\nu}^\top]^\top \in \mathbb{R}^{15}$ is the state vector, $\mathbf{u} = \boldsymbol{\tau} \in \mathbb{R}^3$ is the control vector, $\mathbf{w} = [\mathbf{w}_1^\top, \mathbf{w}_2^\top, \mathbf{w}_3^\top]^\top \in \mathbb{R}^9$ represents the process white noise vector, and:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{A}_w \boldsymbol{\xi} \\ \mathbf{R}(\psi)\boldsymbol{\nu} \\ \mathbf{0}_{3\times 1} \\ \mathbf{M}^{-1}(-\mathbf{D}\boldsymbol{\nu} + \mathbf{R}^\top(\psi)\mathbf{b}) \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0}_{6\times 3} \\ \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} \\ \mathbf{M}^{-1} \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{C}_w & \mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \mathbf{E}_w \\ \mathbf{0}_{3\times 3} \\ \mathbf{I}_{3\times 3} \\ \mathbf{M}^{-1} \end{bmatrix} \tag{6.85}$$

Table 6.5: Discrete-Time Extended Kalman Filter (EKF) .

Design matrices	$\mathbf{Q}(k)=\mathbf{Q}^\top(k)>0, \mathbf{R}(k)=\mathbf{R}^\top(k)>0$ (usually constant)
Initial conditions	$\overline{\mathbf{x}}(0)=\mathbf{x}_0$ $\overline{\mathbf{P}}(0)=E[(\mathbf{x}(0)-\hat{\mathbf{x}}(0))(\mathbf{x}(0)-\hat{\mathbf{x}}(0))^\top]=\mathbf{P}_0$
Kalman gain matrix State estimate update Error covariance update	$\mathbf{K}(k)=\overline{\mathbf{P}}(k)\mathbf{H}^\top(k)\,[\mathbf{H}(k)\overline{\mathbf{P}}(k)\mathbf{H}^\top(k)+\mathbf{R}(k)]^{-1}$ $\hat{\mathbf{x}}(k)=\overline{\mathbf{x}}(k)+\mathbf{K}(k)\,[\mathbf{y}(k)-\mathbf{H}(k)\overline{\mathbf{x}}(k)]$ $\hat{\mathbf{P}}(k)=[\mathbf{I}-\mathbf{K}(k)\mathbf{H}(k)]\,\overline{\mathbf{P}}(k)\,[\mathbf{I}-\mathbf{K}(k)\mathbf{H}(k)]^\top$ $+\mathbf{K}(k)\mathbf{R}(k)\mathbf{K}^\top(k), \quad \hat{\mathbf{P}}(k)=\hat{\mathbf{P}}(k)^\top>0$
State estimate propagation Error covariance propagation	$\overline{\mathbf{x}}(k+1)=\mathbf{F}(\hat{\mathbf{x}}(k),\mathbf{u}(k))$ $\overline{\mathbf{P}}(k+1)=\mathbf{\Phi}(k)\hat{\mathbf{P}}(k)\mathbf{\Phi}^\top(k)+\mathbf{\Gamma}(k)\mathbf{Q}(k)\mathbf{\Gamma}^\top(k)$

The state vector of this system can be estimated using the discrete-time EKF algorithm of Table 6.5.

The discrete-time quantities $\mathbf{F}(\hat{\mathbf{x}}(k),\mathbf{u}(k))$, $\mathbf{\Phi}(k)$and $\mathbf{\Gamma}(k)$ in Table 6.5 can be found by using *forward Euler* integration for instance. Moreover:

$$\mathbf{F}(\hat{\mathbf{x}}(k),\mathbf{u}(k)) = \hat{\mathbf{x}}(k)+h[\mathbf{f}(\hat{\mathbf{x}}(k))+\mathbf{B}\mathbf{u}(k)] \tag{6.86}$$

$$\mathbf{\Phi}(k) = \mathbf{I}+h\left.\frac{\partial \mathbf{f}(\mathbf{x}(k),\mathbf{u}(k))}{\partial \mathbf{x}(k)}\right|_{\mathbf{x}(k)=\hat{\mathbf{x}}(k)} \tag{6.87}$$

$$\mathbf{\Gamma}(k) = h\mathbf{E} \tag{6.88}$$

where $h>0$ is the sampling time. The EKF has been used in most industrial ship control systems. It should, however, be noted that the there are no proof of global asymptotic stability when the system is linearized. In particular, it is difficult to obtain asymptotic convergence of the bias estimates when using the EKF algorithm in DP and PM. In Section 6.2.4, it is demonstrated how a nonlinear observer can be designed to meet the requirement of global asymptotical/exponential stability through a passivation design.

Linear Vessel Parallel Kalman Filter Design

Since the only nonlinear term in (6.71)–(6.72) is the rotation matrix $\mathbf{R}(\psi)$ it is attractive to use vessel parallel coordinates $\boldsymbol{\eta}_p$ instead of $\boldsymbol{\eta}$ when designing the observer (see Section 3.3.2):

$$\boldsymbol{\eta}_p = \mathbf{R}^\top(\psi)\boldsymbol{\eta} \tag{6.89}$$

such that

$$\begin{aligned}\dot{\boldsymbol{\eta}}_p &= \mathbf{R}^\top(\psi)\dot{\boldsymbol{\eta}} + \dot{\mathbf{R}}^\top(\psi)\boldsymbol{\eta} \\ &= \mathbf{R}^\top(\psi)\mathbf{R}(\psi)\boldsymbol{\nu} + \dot{\mathbf{R}}^\top(\psi)\mathbf{R}(\psi)\boldsymbol{\eta}_p \\ &= \boldsymbol{\nu} + \dot{\mathbf{R}}^\top(\psi)\boldsymbol{\eta} \end{aligned} \tag{6.90}$$

The following assumption is needed to linearize $\mathbf{f}(\mathbf{x})$.

Assumption K1: Constant heading–i.e., $r = 0$ such that $\dot{\mathbf{R}}(\psi) = 0$. This is a good approximation for low-speed applications.

Hence, the use of vessel parallel coordinates implies that the kinematics (6.90) becomes linear:

$$\dot{\boldsymbol{\eta}}_p \approx \boldsymbol{\nu} \tag{6.91}$$

Application of Assumption K1 to the bias model (6.75) suggests the following vessel parallel bias formulation (Wiener process):

$$\dot{\mathbf{b}}_p = \mathbf{w}_2 \tag{6.92}$$

$$\mathbf{b} = \mathbf{R}(\psi)\mathbf{b}_p \tag{6.93}$$

The linear model with $\mathbf{x} = [\boldsymbol{\xi}^\top, \boldsymbol{\eta}_p^\top, \mathbf{b}_p^\top, \boldsymbol{\nu}^\top]^\top$ finally takes the form:

$$\begin{aligned}\mathbf{f}(\mathbf{x}) &\approx \mathbf{A}\mathbf{x} \\ &= \begin{bmatrix} \mathbf{A}_w & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} & \mathbf{0}_{6\times3} \\ \mathbf{0}_{3\times6} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times6} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times6} & \mathbf{0}_{3\times3} & \mathbf{M}^{-1} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix} \mathbf{x} \end{aligned} \tag{6.94}$$

which is quite attractive for Kalman filter design since the A-matrix it is independent of the yaw angle ψ.

The continuous time filter equations for this system is given by (see Table 6.1 in Section 6.1.3):

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \underbrace{\mathbf{P}\mathbf{H}^\top\mathbf{R}^{-1}}_{\mathbf{K}}(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}) \tag{6.95}$$

$$\dot{\mathbf{P}} = \mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^\top + \mathbf{E}\mathbf{Q}\mathbf{E}^\top - \mathbf{P}\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}\mathbf{P} \tag{6.96}$$

Notice that the covariance matrices $\mathbf{Q} = \mathbf{Q}^\top \in \mathbb{R}^{9\times9}$ and $\mathbf{R} = \mathbf{R}^\top \in \mathbb{R}^{3\times3}$ must be specified by the user. The measurement covariance matrix can be chosen as $\mathbf{R} = \text{diag}\{r_1, r_2, r_3\}$ where the elements r_1 and r_2 are the covariance of the GPS position measurements and r_3 is the compass covariance. The matrix $\mathbf{Q}$ can also be chosen to be diagonal with positive tunable parameters. These are usually found by trial and error.

Since the GPS measurement frequency can be as low as 1–10 Hz it is advantageous to implement the discrete-time version of the KF using the *predictor-corrector* representation; see Table 6.2.

Notice that Assumption K1 implies that only local exponential stability can be proven for the vessel-paralell representation. If global exponential stability is important, the passive observer of the next section should be applied.

6.2.4 Passive Nonlinear Observer for Velocity and Wave Frequency Motion

The drawback of the Kalman filter is that it is difficult and time-consuming to tune the state estimator (stochastic system with 15 states and 120 covariance equations). The main reason for this, is that the numerous covariance tuning parameters may be difficult to relate to physical quantities. This results in an *ad hoc* tuning procedure for the process covariance matrix $\mathbf{Q}$ while the measurement covariance matrix $\mathbf{R}$ usually is well defined in terms of sensor specifications.

Introduction

In the 1990s, vectorial observer backstepping was presented as an alternative design methodology for DP state estimation (Fossen and Grøvlen 1998). The motivation for this was to avoid linearization of the yaw kinematics in order to obtain a global stability result. Another motivating factor was to reduce the relatively time-consuming process of tuning the Kalman filter covariance matrices on-line. In fact, vectorial observer backstepping resulted in a uniformly globally exponentially stable (UGES) output feedback control system, which could be directly applied to station-keeping of ships and rigs. The work of Fossen and Grøvlen (1998) is, however, based on a simplified model of the environmental disturbances, since it is assumed that the WF motion and bias states can be neglected in the design. Aarset *et al.* (1998) have shown that these results can be extended to the general case by including a dynamic model for wave filtering and bias state estimation. It is also possible to extend this result to ships that are course-unstable (open-loop unstable in sway and yaw) thanks to the results by Robertson and Johansson (1998), and Lindegaard and Fossen (2001*b*).

A drawback with observer backstepping and also Kalman filter based design techniques, is that a relatively large number of parameters must be determined through experimental testing of the vessel. This motivated the research of a nonlinear passivity-based observer, since passivity arguments simplify the tuning procedure significantly (Fossen and Strand 1999*b*). Hence, the time needed for sea-trials and tuning can be drastically reduced. The nonlinear passive observer, as opposed to a linearized or extended Kalman-filter, guarantees global convergence of all estimation errors (including the bias terms) to zero. Hence, only one set of observer gains are needed to cover the whole state space. In addition, the number of observer tuning parameters are significantly reduced and the wave filter parameters are directly coupled to the dominating wave frequency. Passivity implies that the phase of the error dynamics is limited by 90 degrees, which gives excellent stability properties. In fact the closed-loop system is UGES if (6.75) is used and UGAS for (6.76). Passivity theory also showed to be a new tool with respect to accurate tuning of the observer. The proposed nonlinear observer opens for new controller designs more in line with the actual structure of the physical system e.g. by using a nonlinear *separation principle* (Loria *et al.* 2000).

For extensions to adaptive wave filtering, see Strand and Fossen (1999), while extensions to position mooring systems are found in Strand (1999).

Assumptions

When designing the passive observer, the Kalman filter low-speed assumption, Assumption K1, is removed. The following assumptions are, however, necessary:

Assumption P1: $\mathbf{w} = \mathbf{0}$ and $\mathbf{v} = \mathbf{0}$. The zero-mean Gaussian white noise terms are omitted in the analysis of the observer. If they are included in the Lyapunov function analysis the error dynamics will be uniformly ultimated bounded (UUB) instead of uniform global asymptotical/exponential stable (UGAS/UGES).

Assumption P2: $\mathbf{R}(y_3) = \mathbf{R}(\psi)$ implying that $y_3 = \psi + \psi_w \approx \psi$. This is a good assumption since the magnitude of the wave-induced yaw disturbance ψ_w will normally be less than 5 degrees in extreme weather situations (sea state codes 5–9), and less than 1 degree during normal operation of the ship (sea state codes 0–4).

The following model properties of the inertia and damping matrices will be exploited in the passivation design:

$$\mathbf{M} = \mathbf{M}^\top > 0, \quad \dot{\mathbf{M}} = \mathbf{0}, \quad \mathbf{D} > 0$$

System Model for Nonlinear Passive Observer

The application of Assumptions P1–P2 to (6.71)–(6.77), gives the following DP observer model:

$$\dot{\boldsymbol{\xi}} = \mathbf{A}_w \boldsymbol{\xi} \tag{6.97}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(y_3)\boldsymbol{\nu} \tag{6.98}$$

$$\dot{\mathbf{b}} = \mathbf{T}^{-1}\mathbf{b} \qquad (\text{alternatively } \dot{\mathbf{b}} = \mathbf{0}) \tag{6.99}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} = -\mathbf{D}\boldsymbol{\nu} + \mathbf{R}^\top(y_3)\mathbf{b} + \boldsymbol{\tau} \tag{6.100}$$

$$\mathbf{y} = \boldsymbol{\eta} + \mathbf{C}_w \boldsymbol{\xi} \tag{6.101}$$

where the alternative bias model formulation (6.75) is given in the brackets. For notational simplicity (6.97), (6.98) and (6.101) are written in state-space form:

$$\dot{\boldsymbol{\eta}}_0 = \mathbf{A}_0\boldsymbol{\eta}_0 + \mathbf{B}_0\mathbf{R}(y_3)\boldsymbol{\nu} \tag{6.102}$$

$$\mathbf{y} = \mathbf{C}_0\boldsymbol{\eta}_0 \tag{6.103}$$

where $\boldsymbol{\eta}_0 = [\boldsymbol{\xi}^\top, \boldsymbol{\eta}^\top]^\top$ and:

$$\mathbf{A}_0 = \begin{bmatrix} \mathbf{A}_w & \mathbf{0}_{6\times 3} \\ \mathbf{0}_{3\times 6} & \mathbf{0}_{3\times 3} \end{bmatrix}, \quad \mathbf{B}_0 = \begin{bmatrix} \mathbf{0}_{6\times 3} \\ \mathbf{I}_{3\times 3} \end{bmatrix}, \quad \mathbf{C}_0 = \begin{bmatrix} \mathbf{C}_w & \mathbf{I}_{3\times 3} \end{bmatrix} \tag{6.104}$$

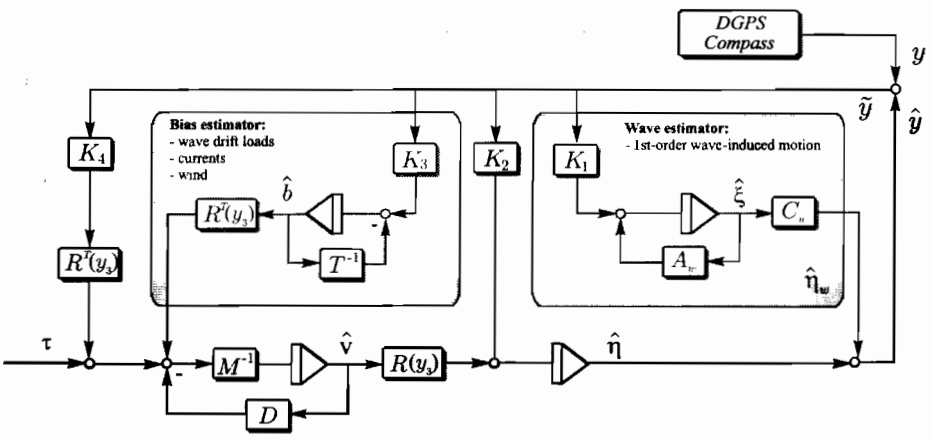


Figure 6.13: Block diagram showing the nonlinear passive DP observer.

Observer Equations

The observer equations can be chosen to copy the dynamics (6.97)–(6.101) resulting in 15 ODEs with no covariance updates as shown in Figure 6.13. Moreover:

$$\dot{\hat{\boldsymbol{\xi}}} = \mathbf{A}_w \hat{\boldsymbol{\xi}} + \mathbf{K}_1(\boldsymbol{\omega}_o)\tilde{\mathbf{y}} \tag{6.105}$$

$$\dot{\hat{\boldsymbol{\eta}}} = \mathbf{R}(y_3)\hat{\boldsymbol{\nu}} + \mathbf{K}_2\tilde{\mathbf{y}} \tag{6.106}$$

$$\dot{\hat{\mathbf{b}}} = -\mathbf{T}^{-1}\hat{\mathbf{b}} + \mathbf{K}_3\tilde{\mathbf{y}} \qquad (\text{alternatively } \dot{\hat{\mathbf{b}}} = \mathbf{K}_3\tilde{\mathbf{y}}) \tag{6.107}$$

$$\mathbf{M}\dot{\hat{\boldsymbol{\nu}}} = -\mathbf{D}\hat{\boldsymbol{\nu}} + \mathbf{R}^\top(y_3)\hat{\mathbf{b}} + \boldsymbol{\tau} + \mathbf{R}^\top(y_3)\mathbf{K}_4\tilde{\mathbf{y}} \tag{6.108}$$

$$\hat{\mathbf{y}} = \hat{\boldsymbol{\eta}} + \mathbf{C}_w\hat{\boldsymbol{\xi}} \tag{6.109}$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}$ is the estimation error and $\mathbf{K}_1(\boldsymbol{\omega}_o) \in \mathbb{R}^{6\times 3}$ and $\mathbf{K}_{2,3,4} \in \mathbb{R}^{3\times 3}$ are observer gain matrices to be interpreted later. Notice that $\mathbf{K}_1(\boldsymbol{\omega}_o)$ is a function of the wave peak frequencies $\boldsymbol{\omega}_o = [\omega_{o1}, \omega_{o2}, \omega_{o3}]^\top$ in surge, sway, and yaw.

The main difference in performance of the two bias state estimators (6.107) is that the first model includes low-pass filtering ($\mathbf{T} > 0$) instead of pure integration of the white noise term $\mathbf{K}_3\tilde{\mathbf{y}}$. This results in *exponential stability* while application of the model $\dot{\hat{\mathbf{b}}} = \mathbf{K}_3\tilde{\mathbf{y}}$ only results in *asymptotic stability* (Vik and Fossen 2002).

Observer Estimation Errors

Similarly as (6.102) and (6.103), the system (6.105), (6.106) and (6.109) is written in state-space form:

$$\dot{\hat{\boldsymbol{\eta}}}_0 = \mathbf{A}_0\hat{\boldsymbol{\eta}}_0 + \mathbf{B}_0\mathbf{R}(y_3)\hat{\boldsymbol{\nu}} + \mathbf{K}_0(\boldsymbol{\omega}_o)\tilde{\mathbf{y}} \tag{6.110}$$

$$\hat{\mathbf{y}} = \mathbf{C}_0\hat{\boldsymbol{\eta}}_0 \tag{6.111}$$

where $\hat{\boldsymbol{\eta}}_0 = [\hat{\boldsymbol{\xi}}^\top, \hat{\boldsymbol{\eta}}^\top]^\top$ and:

$$\mathbf{K}_0(\boldsymbol{\omega}_o) = \begin{bmatrix} \mathbf{K}_1(\boldsymbol{\omega}_o) \\ \mathbf{K}_2 \end{bmatrix} \tag{6.112}$$

The estimation errors are defined as $\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu} - \hat{\boldsymbol{\nu}}$, $\tilde{\mathbf{b}} = \mathbf{b} - \hat{\mathbf{b}}$ and $\tilde{\boldsymbol{\eta}}_0 = \boldsymbol{\eta}_0 - \hat{\boldsymbol{\eta}}_0$. Hence, the error dynamics can be written:

$$\dot{\tilde{\boldsymbol{\eta}}}_0 = [\mathbf{A}_0 - \mathbf{K}_0(\boldsymbol{\omega}_o)\mathbf{C}_0]\tilde{\boldsymbol{\eta}}_0 + \mathbf{B}_0\mathbf{R}(y_3)\tilde{\boldsymbol{\nu}} \tag{6.113}$$

$$\dot{\tilde{\mathbf{b}}} = -\mathbf{T}^{-1}\tilde{\mathbf{b}} - \mathbf{K}_3\tilde{\mathbf{y}} \qquad \text{(alternatively } \dot{\tilde{\mathbf{b}}} = -\mathbf{K}_3\tilde{\mathbf{y}}) \tag{6.114}$$

$$\mathbf{M}\dot{\tilde{\boldsymbol{\nu}}} = -\mathbf{D}\tilde{\boldsymbol{\nu}} + \mathbf{R}^\top(y_3)\tilde{\mathbf{b}} - \mathbf{R}^\top(y_3)\mathbf{K}_4\tilde{\mathbf{y}}. \tag{6.115}$$

In the Lyapunov analysis of the error dynamics (6.113)–(6.115), it is possible to prove UGES for $\mathbf{T} > 0$ (Fossen and Strand 1999*b*) since $\dot{V}(\mathbf{x},t) < 0$ (negative definite). If the bias model $\dot{\mathbf{b}} = \mathbf{0}$ is applied, that is $\mathbf{T} \to \infty$, the Lyapunov analysis results in $\dot{V}(\mathbf{x},t) \leq 0$ (negative semi-definite). Since the error dynamics is nonautonomous (recall that $y_3 = y_3(t)$ is time-varying), Krasovskii-LaSalle's theorem cannot be applied to prove global asymptotic stability; see Appendix A. However, it is possible to prove UGAS by using Matrosov's theorem. Technicalities with respect to this are omitted in this section, but the interested reader is advised to consult (Vik and Fossen 2002) for details regarding the limiting case $\mathbf{T} \to \infty$. The analysis for $\mathbf{T} > 0$, however, is given below.

The dynamics of the velocity estimation error (6.115) is rewritten as:

$$\mathbf{M}\dot{\tilde{\boldsymbol{\nu}}} = -\mathbf{D}\tilde{\boldsymbol{\nu}} - \mathbf{R}^\top(y_3)\tilde{\mathbf{z}} \tag{6.116}$$

where

$$\tilde{\mathbf{z}} = \mathbf{K}_4\tilde{\mathbf{y}} - \tilde{\mathbf{b}}. \tag{6.117}$$

By defining a new state vector:

$$\tilde{\mathbf{x}} = \begin{bmatrix} \tilde{\boldsymbol{\eta}}_0 \\ \tilde{\mathbf{b}} \end{bmatrix} \tag{6.118}$$

Equations (6.113), (6.114) and (6.117) can be written in compact form as:

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} + \mathbf{B}\mathbf{R}(y_3)\tilde{\boldsymbol{\nu}} \tag{6.119}$$

$$\tilde{\mathbf{z}} = \mathbf{C}\tilde{\mathbf{x}} \tag{6.120}$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_0 - \mathbf{K}_0(\boldsymbol{\omega}_o)\mathbf{C}_0 & \mathbf{0}_{9\times 9} \\ -\mathbf{K}_3\mathbf{C}_0 & -\mathbf{T}^{-1} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{B}_0 \\ \mathbf{0}_{3\times 3} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{K}_4\mathbf{C}_0 & -\mathbf{I}_{3\times 3} \end{bmatrix} \tag{6.121}$$

In Figure 6.14 the error signals $\boldsymbol{\varepsilon}_z$ and $\boldsymbol{\varepsilon}_\nu$ are defined according to:

$$\boldsymbol{\varepsilon}_z = -\mathbf{R}^\top(y_3)\tilde{\mathbf{z}}, \quad \boldsymbol{\varepsilon}_\nu = \mathbf{R}(y_3)\tilde{\boldsymbol{\nu}}. \tag{6.122}$$

Thus, the observer error system can be viewed as two linear blocks $\mathcal{H}_1$ and $\mathcal{H}_2$, interconnected through the bounded transformation matrix $\mathbf{R}(y_3)$, that is:

$$\mathcal{H}_1 \; : \; \left\{ \mathbf{M}\dot{\tilde{\boldsymbol{\nu}}} = -\mathbf{D}\tilde{\boldsymbol{\nu}} + \boldsymbol{\varepsilon}_z \right. \tag{6.123}$$

$$\mathcal{H}_2 \; : \; \begin{cases} \dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} + \mathbf{B}\boldsymbol{\varepsilon}_\nu \\ \tilde{\mathbf{z}} = \mathbf{C}\tilde{\mathbf{x}} \end{cases} \tag{6.124}$$

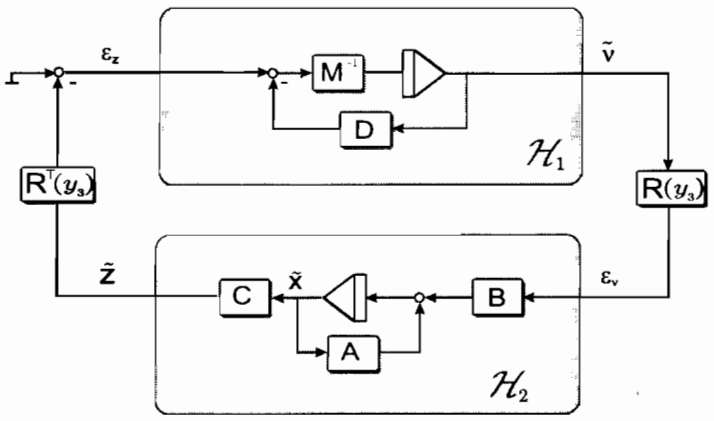


Figure 6.14: Block diagram showing the dynamics of the position/bias and velocity estimation errors.

Stability Analysis for the Passive Observer

Based on the physical properties of the ship dynamics, the following statement can be made:

Proposition 1 (Strictly Passive Velocity Error Dynamics)
The mapping $\mathcal{H}_1$ is state strictly passive.

Proof. *Let:*

$$S_1 = \frac{1}{2}\tilde{\nu}^\top \mathbf{M}\tilde{\nu} \tag{6.125}$$

be a positive definite storage function. Time differentiation of S_1 along the trajectories of $\tilde{\nu}$ yields:

$$\dot{S}_1 = -\frac{1}{2}\tilde{\nu}^\top \left(\mathbf{D} + \mathbf{D}^\top\right)\tilde{\nu} - \tilde{\mathbf{z}}^\top \mathbf{R}(y_3)\tilde{\nu} \tag{6.126}$$

Using the fact that $\boldsymbol{\varepsilon}_z = -\mathbf{R}^\top(y_3)\tilde{\mathbf{z}}$, yields:

$$\boldsymbol{\varepsilon}_z^\top \tilde{\boldsymbol{\nu}} = \dot{S}_1 + \frac{1}{2}\tilde{\boldsymbol{\nu}}^\top \left(\mathbf{D} + \mathbf{D}^\top\right)\tilde{\boldsymbol{\nu}} \tag{6.127}$$

Hence:

$$\int_{t_0}^{t} \varepsilon_z^\top(\tau)\tilde{\boldsymbol{\nu}}(\tau)\, d\tau \geq \alpha\tilde{\boldsymbol{\nu}}^\top\tilde{\boldsymbol{\nu}} + \beta \tag{6.128}$$

where $\alpha = \frac{1}{2}\lambda_{\min}(\mathbf{M})$ is a positive constant and:

$$\beta = \frac{1}{2}\int_{t_0}^{t} \tilde{\boldsymbol{\nu}}^\top \left(\mathbf{D} + \mathbf{D}^\top\right)\tilde{\boldsymbol{\nu}} d\tau \geq 0 \tag{6.129}$$

is the dissipated energy due to hydrodynamic damping. Thus, (6.128) proves that $\boldsymbol{\varepsilon}_z \mapsto \tilde{\boldsymbol{\nu}}$ or the block H_1 is state strictly passive.

For definitions on passivity see Sepulchre *et al.* (1997), Ortega *et al.* (1998) or Lozano *et al.* (2000), for instance.

In order to show that the interconnected system in Figure 6.14 is passive, one of the blocks must be passive while the other block must be strictly passive (Lozano *et al.* 2000). Since the mapping $\varepsilon_z \mapsto \tilde{\nu}$ is strictly passive (block $\mathcal{H}_1$), post-multiplication with the bounded transformation matrix $\mathbf{R}(y_3)$ and pre-multiplication by it's transpose will not affect the passivity properties. Hence, it only remains to show that the the mapping $\varepsilon_\nu \mapsto \tilde{\mathbf{z}}$ (block $\mathcal{H}_2$) is *passive*. This can be done by applying the *Kalman-Yakubovich-Popov (KYP) Lemma.*

Lemma 6.1 (Kalman-Yakubovich-Popov)
Let $\mathbf{Z}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ *be an* $m \times m$ *transfer function matrix, where* $\mathbf{A}$ *is Hurwitz,* $(\mathbf{A}, \mathbf{B})$ *is controllable, and* $(\mathbf{A}, \mathbf{C})$ *is observable. Then* $\mathbf{Z}(s)$ *is strictly positive real (SPR) if and only if there exist positive definite matrices* $\mathbf{P} = \mathbf{P}^\top$ *and* $\mathbf{Q} = \mathbf{Q}^\top$ *such that:*

$$\mathbf{P}\mathbf{A} + \mathbf{A}^\top\mathbf{P} = -\mathbf{Q} \tag{6.130}$$

$$\mathbf{B}^\top\mathbf{P} = \mathbf{C} \tag{6.131}$$

Proof. *See Yakubovich (1973) or Khalil (2002).*

Theorem 6.1 (Passive Observer Error Dynamics)
The interconnected system (6.123) and (6.124) is passive if the observer gain matrices $\mathbf{K}_i$ *(*$i = 1, .., 4$*) are chosen such that (6.124) satisfies the KYP-Lemma.*

Proof. *Since it is established that* H_1 *is strictly passive and* H_2*, which is given by the matrices (A, B, C) can be made SPR by choosing the gain matrices* K_i *(*$i = 1, \ldots, 4$*) according to the KYP lemma, the interconnected system (6.123) and (6.124) is passive (Fossen and Strand 1999b).*

Determination of the Observer Gains

In practice it is easy to find a set of gain matrices $\mathbf{K}_i$ $(i = 1, \ldots, 4)$ satisfying the KYP lemma. Notice that the mapping $\varepsilon_\nu \mapsto \tilde{\mathbf{z}}$ (block $\mathcal{H}_2$) describes three decoupled systems in surge, sway, and yaw. This suggests that the observer gain matrices should have a diagonal structure:

$$\mathbf{K}_1(\omega_o) = \begin{bmatrix} \text{diag}\{K_{11}(\omega_{o1}), K_{12}(\omega_{o2}), K_{13}(\omega_{o3})\} \\ \text{diag}\{K_{14}(\omega_{o1}), K_{15}(\omega_{o3}), K_{16}(\omega_{o3})\} \end{bmatrix} \tag{6.132}$$

$$\mathbf{K}_2 = \text{diag}\{K_{21}, K_{22}, K_{23}\} \tag{6.133}$$

$$\mathbf{K}_3 = \text{diag}\{K_{31}, K_{32}, K_{33}\} \tag{6.134}$$

$$\mathbf{K}_4 = \text{diag}\{K_{41}, K_{42}, K_{43}\} \tag{6.135}$$

Consequently, three decoupled transfer functions can be defined:

$$\mathbf{H}(s) = \text{diag}\{h_1(s), h_2(s), h_3(s)\} \tag{6.136}$$

given by:

$$\tilde{\mathbf{z}}(s) = \mathbf{H}(s)\varepsilon_\nu(s) = \mathbf{H}_0(s)\mathbf{H}_B(s)\varepsilon_\nu(s)$$

where:

$$\begin{aligned}\mathbf{H_0}(s) &= \mathbf{C_0}[s\mathbf{I} + \mathbf{A_0} - \mathbf{K_0}(\boldsymbol{\omega_0})\mathbf{C_0}]^{-1}\mathbf{B_0} \\ \mathbf{H}_B(s) &= \mathbf{K_4} + (s\mathbf{I} + \mathbf{T}^{-1})^{-1}\mathbf{K_3}\end{aligned}$$

The diagonal structure of $\mathbf{H}(s)$ is shown in Figure 6.15. The transfer functions $h_{oi}(s)$ $(i = 1,\ldots,3)$ of $\mathbf{H_0}(s)$ and $h_{Bi}(s)$ $(i = 1,\ldots,3)$ of $\mathbf{H}_B(s)$ of becomes:

$$h_{oi}(s) = \frac{s^2 + 2\lambda_i\omega_{oi}s + \omega_{oi}^2}{s^3 + (K_{1(i+3)} + K_{2i} + 2\lambda_i\omega_{oi})s^2 + (\omega_{oi}^2 + 2\lambda_i\omega_{oi}K_{2i} - K_{1i}\omega_{oi}^2)s + \omega_{oi}^2K_{2i}} \tag{6.137}$$

$$h_{Bi}(s) = K_{4i}\frac{s + \left(\frac{1}{T_i} + \frac{K_{3i}}{K_{4i}}\right)}{s + \frac{1}{T_i}} \overset{T_i \gg 1}{\approx} K_{4i}\frac{s + \frac{K_{3i}}{K_{4i}}}{s + \frac{1}{T_i}} \tag{6.138}$$

where ω_{oi} is the wave frequency, T_i is defined in (6.76), and λ_i is the wave spectrum damping ratio. In order to obtain the desired notch effect (wave filtering) of the observer, the desired shape of $h_{oi}(s)$ is specified as:

$$h_{di}(s) = \frac{s^2 + 2\lambda_i\omega_{oi}s + \omega_{oi}^2}{(s^2 + 2\zeta_{ni}\omega_{oi}s + \omega_{oi}^2)(s + \omega_{ci})} \tag{6.139}$$

where $\zeta_{ni} > \lambda_i$ determines the notch and $\omega_{ci} > \omega_{oi}$ is the filter cut-off frequency. Typically $\zeta_{ni} = 1.0$ and $\lambda_i = 0.1$. Equating (6.137) and (6.139) yields the following formulas for the filter gains in $\mathbf{K_1}(\boldsymbol{\omega}_o)$ and $\mathbf{K_2}$:

$$K_{1i}(\omega_{oi}) = -2(\zeta_{ni} - \lambda_i)\frac{\omega_{ci}}{\omega_{oi}} \tag{6.140}$$

$$K_{1(i+3)}(\omega_{oi}) = 2\omega_{oi}(\zeta_{ni} - \lambda_i) \tag{6.141}$$

$$K_{2i} = \omega_{ci} \tag{6.142}$$

Notice that the filter gains can be gain-scheduled with respect to the dominating wave frequencies ω_{oi} if desired. In Figure 6.15 the transfer function $h_i(s) = h_{Bi}(s)h_{oi}(s)$ is illustrated when all filter gains are properly selected. It is important that the 3 decoupled transfer functions $h_i(s)$ all have phase greater than $-90°$ in order to meet the SPR requirement. It turns out that the KYP lemma and therefore the SPR requirement can easily be satisfied if the following tuning rules for T_i, K_{3i} and K_{4i} are applied:

$$1/T_i \ll K_{3i}/K_{4i} < \omega_{oi} < \omega_{ci}, \qquad (i = 1,\ldots,3) \tag{6.143}$$

Here ω_{oi} $(i = 1,\ldots,3)$ are the dominating wave frequencies and $T_i \gg 1$ $(i = 1,\ldots,3)$ are the bias time constants used to specify the limited integral effect in the bias estimator.

Uniform Global Exponential Stability

The passivity analysis mainly serves as a tool to determine the observer gains. In order to ensure that all estimation errors converge exponentially to zero the following theorem is applied.

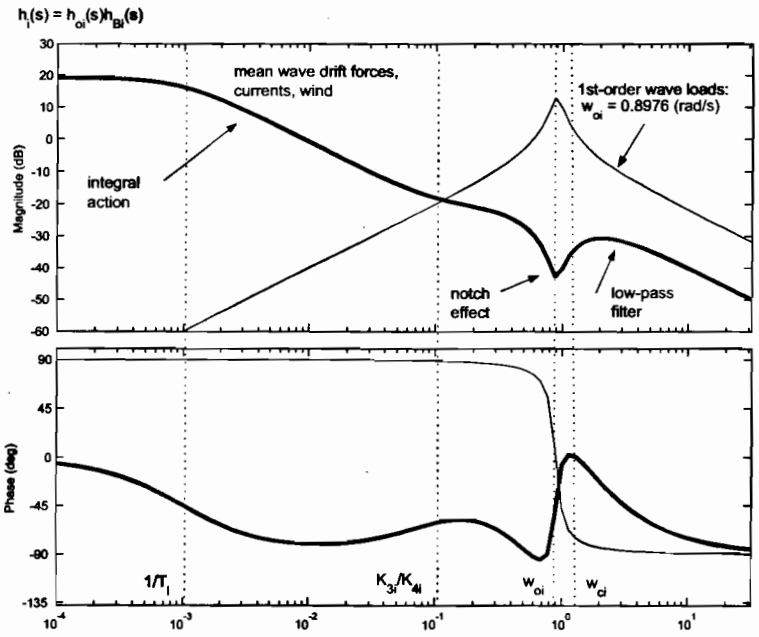


Figure 6.15: Bode plot showing the transfer function $h_i(s)$ in surge ($i = 1$) when $1/T_i \ll K_{3i}/K_{4i} < \omega_{oi} < \omega_{ci}$, see ExPassiveObs.m

Theorem 6.2 (Uniformly Globally Exponentially Stable Observer Error Dynamics) *Under Assumptions P1–P2 the nonlinear observer given by (6.105)–(6.109) is uniformly globally exponentially stable.*

Proof. *Consider the following Lyapunov function candidate:*

$$V = \tilde{\boldsymbol{\nu}}^\top \mathbf{M} \tilde{\boldsymbol{\nu}} + \tilde{\mathbf{x}}^\top \mathbf{P} \tilde{\mathbf{x}}. \tag{6.144}$$

Differentiation of V along the trajectories of $\tilde{\boldsymbol{\nu}}$ and $\tilde{\mathbf{x}}$ and application of Assumptions P1–P2, yields:

$$\begin{aligned} \dot{V} &= -\tilde{\boldsymbol{\nu}}^\top \left(\mathbf{D} + \mathbf{D}^\top \right) \tilde{\boldsymbol{\nu}} + \tilde{\mathbf{x}}^\top \left(\mathbf{P}\mathbf{A} + \mathbf{A}^\top \mathbf{P} \right) \tilde{\mathbf{x}} \\ &\quad + 2\tilde{\boldsymbol{\nu}}^\top \mathbf{R}^\top(y_3) \mathbf{B}^\top \mathbf{P} \tilde{\mathbf{x}} - 2\tilde{\boldsymbol{\nu}}^\top \mathbf{R}^\top(y_3) \tilde{\mathbf{z}} \end{aligned} \tag{6.145}$$

Application of the KYP Lemma, that is $\mathbf{B}^\top \mathbf{P} \tilde{\mathbf{x}} = \mathbf{C} \tilde{\mathbf{x}} = \tilde{\mathbf{z}}$, to (6.145), yields:

$$\dot{V} = -\tilde{\boldsymbol{\nu}}^\top \left(\mathbf{D} + \mathbf{D}^\top \right) \tilde{\boldsymbol{\nu}} - \tilde{\mathbf{x}}^\top \mathbf{Q} \tilde{\mathbf{x}} < 0, \quad \forall \tilde{\mathbf{x}} \neq \mathbf{0}, \tilde{\boldsymbol{\nu}} \neq \mathbf{0} \tag{6.146}$$

Hence, $\tilde{\boldsymbol{\nu}}$ and $\tilde{\mathbf{x}} = [\tilde{\boldsymbol{\xi}}^\top, \tilde{\boldsymbol{\eta}}^\top, \tilde{\mathbf{b}}^\top]^\top$ converge exponentially to zero, q.e.d.

The requirement that $\mathbf{D} + \mathbf{D}^\top > 0$ (open-loop stable ship) might be relaxed by using the kinematic transformation of Lindegaard and Fossen (2001a, 2001b). These papers also show how pole-placement and LMI techniques can be used instead of passivity analysis to compute the observer gains. In addition velocity and partial acceleration feedback are included in the observer.

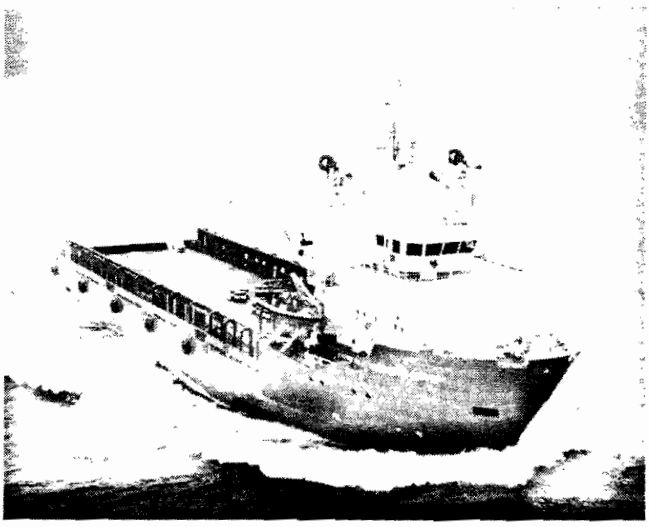


Figure 6.16: The supply vessel *Northern Clipper*. Length $L = 76.2$ (m) and mass $m = 4.591 \cdot 10^6$ (kg).

Computer Simulations and Experimental Results

A combination of computer simulations and full-scale experiments have been used to evaluate the performance and robustness of the nonlinear passive observer.

Example 6.6 (Passive Nonlinear DP Observer)
The case studies are based on the following models of the ship-bias-wave system (Fossen and Strand 1999b):

$$\mathbf{M} = \begin{bmatrix} 5.3122 \cdot 10^6 & 0 & 0 \\ 0 & 8.2831 \cdot 10^6 & 0 \\ 0 & 0 & 3.7454 \cdot 10^9 \end{bmatrix} \tag{6.147}$$

$$\mathbf{D} = \begin{bmatrix} 5.0242 \cdot 10^4 & 0 & 0 \\ 0 & 2.7229 \cdot 10^5 & -4.3933 \cdot 10^6 \\ 0 & -4.3933 \cdot 10^6 & 4.1894 \cdot 10^8 \end{bmatrix} \tag{6.148}$$

with the coordinate system located in the center of gravity. A picture of the actual ship is shown in Figure 6.16. In the experiments the bias time constants were chosen as:

$$\mathbf{T} = diag\{1000, 1000, 1000\} \tag{6.149}$$

The wave model parameters were chosen as $\lambda_i = 0.1$ *and* $\omega_{oi} = 0.8976$ *(rad/s) corresponding to a wave period of* 7.0 *(s) in surge, sway and yaw. The notch filter parameters were chosen as* $\zeta_{ni} = 1.0$ *and* $\omega_{ci} = 1.2255\omega_{oi} = 1.1$*(rad/s). From (6.140)–(6.142) we get, see the GNC toolbox script* `ExPassiveObs.m`*:*

$$\mathbf{K}_1 = \begin{bmatrix} -diag\{2.2059, 2.2059, 2.2059\} \\ diag\{1.6157, 1.6157, 1.6157\} \end{bmatrix} \tag{6.150}$$

$$\mathbf{K}_2 = diag\{1.1, 1.1, 1.1\} \tag{6.151}$$

The loop transfer function $h_{oi}(s) = h_{Bi}(s)h_{oi}(s)$ *for:*

$$\mathbf{K}_4 = diag\{0.1, 0.1, 0.01\}, \qquad \mathbf{K}_3 = 0.1\mathbf{K}_4 \tag{6.152}$$

is shown in Figure 6.15.

Both the simulation study and the full-scale experiment were performed with a measurement frequency of 1 (Hz). The simulation study was performed with non-zero noise terms $\mathbf{v}$ and $\mathbf{w}$ even though these terms were assumed to be zero in the Lyapunov analysis. This was done to demonstrate the excellent performance of the observer in the presence of stochastic noise.

The results of the computer simulations are shown in Figures 6.17–6.18. The plots show that all state estimates converge to their true values. In Figure 6.19–6.20 full scale experimental results with the same observer are reported. Again, excellent convergence and performance in surge, sway, and yaw are observed. In the full scale experiment it was not possible to verify that the velocity estimates converged to their true values, see the lower plots in Figure 6.20. The main reason for this was that only GPS position measurements were available. However, simulation studies indicate that the velocity estimates converge to their true values as well.

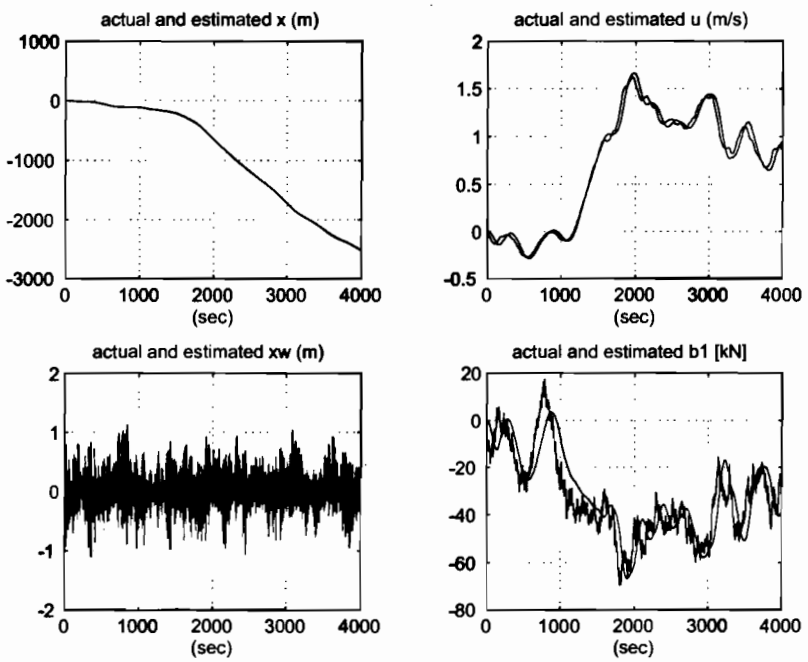


Figure 6.17: Simulation study: LF and WF position, velocity, bias and their estimates in surge.

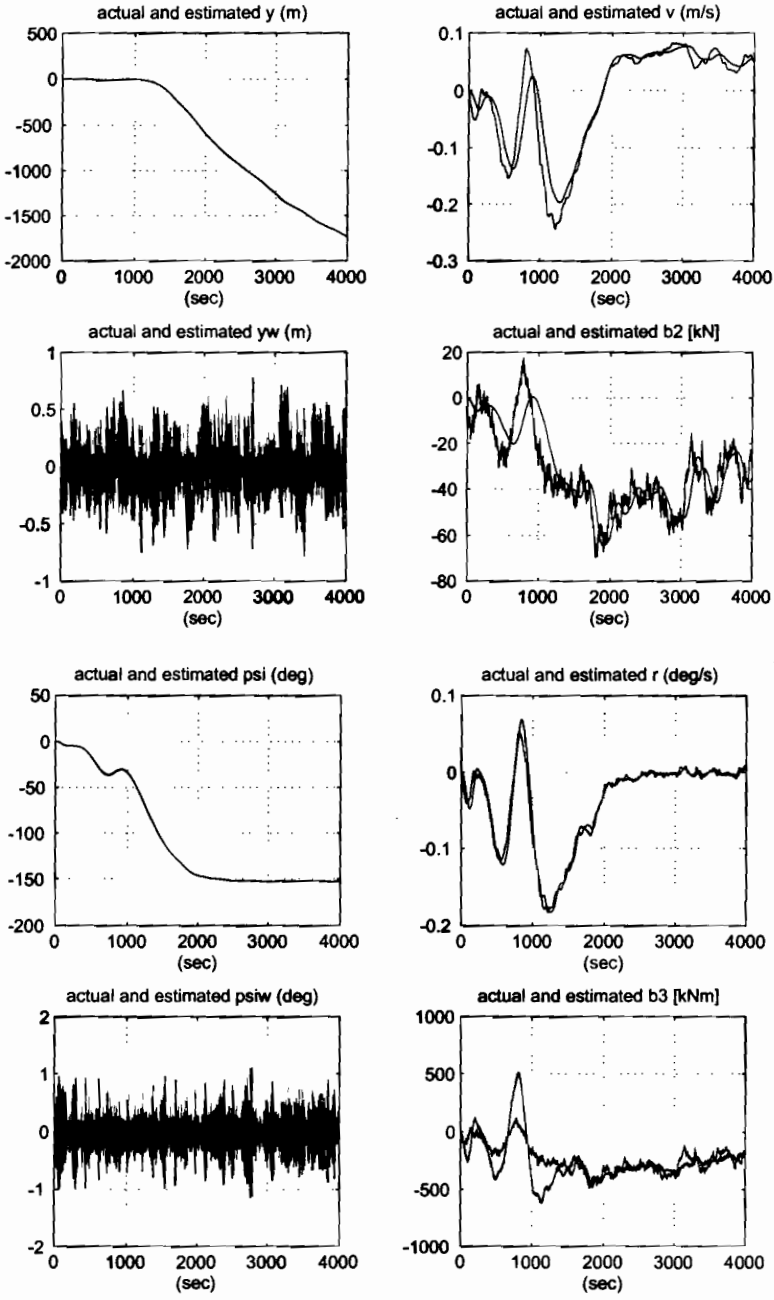


Figure 6.18: Simulation study: LF and WF position, velocity, bias and their estimates in sway and yaw.

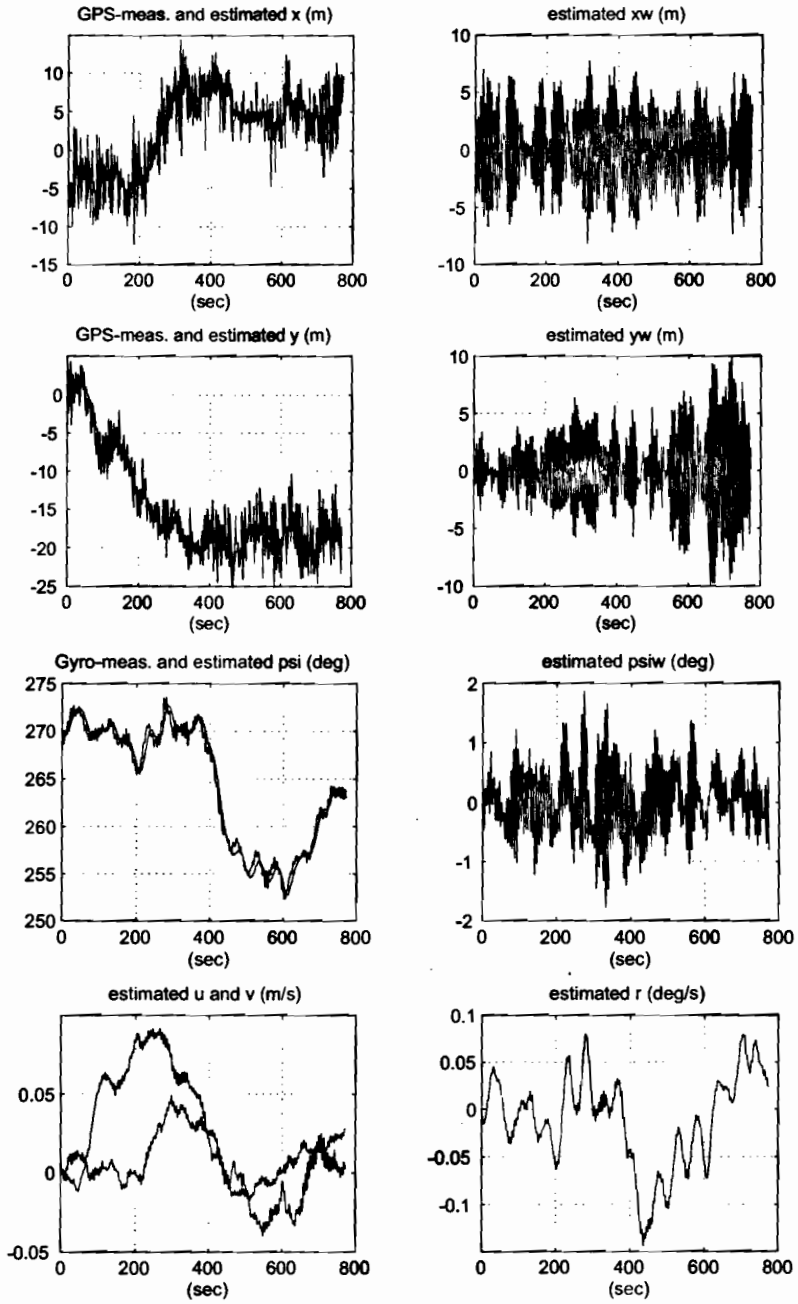


Figure 6.19: Experimental data: Three upper plots - actual position (LF+WF) with estimates of the LF- and WF- positions in surge, sway and yaw. Lower plots - Estimates of the LF velocities.

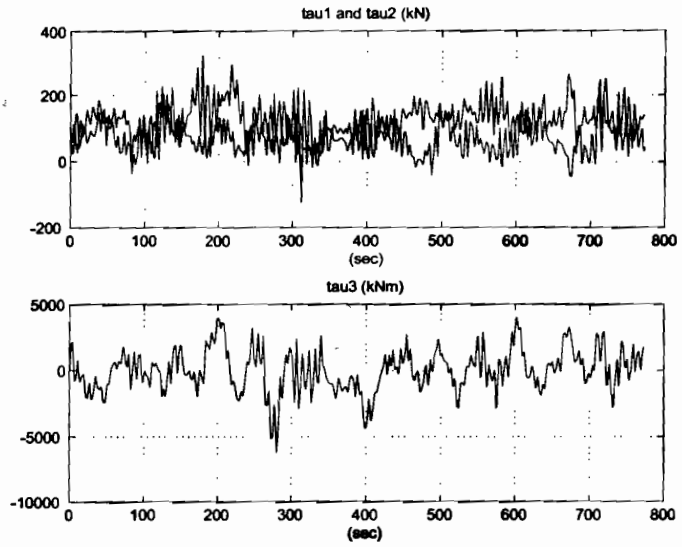


Figure 6.20: Experimental data: Control inputs in surge, sway and yaw.

6.3 6 DOF Integration Filter for IMU and Satellite Navigation Systems

An IMU, see Section 6.2.2, can be integrated with a satellite navigation system in a state observer to obtain estimates of position and velocity in 6 DOF. A stand-alone IMU solution, where acceleration measurements are integrated twice and gyro outputs are integrated once to obtain positions and attitude, respectively, will drift due to sensors biases, misalignments, temperature variations etc. The kinematic equations (strapdown equations) which are integrated numerically in conjuncture with an IMU is defined as an inertial navigation system (INS). The INS drift can be removed by GPS/INS integration in a state observer. The position and velocity accuracy will mainly depend on the GPS quality while acceleration and attitude depend on quality of the accelerometers, gyros, and the compass. If a low-cost IMU is used, the position/attitude estimates will drift rapidly during GPS shortages while a more expensive unit will have better stand-alone capabilities. Construction of integrated GPS/INS navigation systems, their performance, and stand-alone capabilities are described more closely in Farrell and Barth (1998), Titterton and Weston (1997), and Grewal *et al.* (2001) to mention some.

The goal of this section is to present a low-cost IMU/GPS integration technique for ship navigation by neglecting the Earth rotation and assuming that the GPS signals are available all the time. Consequently, the *North-East-Down* reference frame is assumed to be the inertial reference frame even though the Earth is moving relatively to a star fixed reference frame. This is, indeed, a good approximation for ship navigation. The solutions presented here are not intended for INS stand-alone applications or cases with GPS failure. For strapdown inertial navigation applied to marine systems, the interested reader is recommended to consult Vik and Fossen (2002).

IMU Measurements

Let the IMU measurements be denoted by ω_{imu} and $\mathbf{f}_{\text{imu}}$ representing *angular velocity* and *specific force* (force divided by mass). The linear acceleration $\mathbf{a}_{\text{imu}}$ is related to specific force as:

$$\mathbf{f}_{\text{imu}} = \mathbf{a}_{\text{imu}} - \mathbf{g}^b \tag{6.153}$$

where $\mathbf{g}^b$ is the acceleration of gravity decomposed in the b-frame. Let $\mathbf{a}^b$ and $\boldsymbol{\omega}_{ib}^b$ denote the b-frame linear acceleration and angular velocity of the moving body. These quantities relate to the IMU measurements as (Vik and Fossen 2001):

$$\mathbf{a}^b = (\mathbf{I} + \boldsymbol{\Delta}_{\text{acc}})\mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}} + \mathbf{g}^b + \mathbf{w}_1 \tag{6.154}$$

$$\boldsymbol{\omega}_{ib}^b = (\mathbf{I} + \boldsymbol{\Delta}_{\text{gyro}})\boldsymbol{\omega}_{\text{imu}} + \mathbf{b}_{\text{gyro}} + \mathbf{w}_2 \tag{6.155}$$

where $\boldsymbol{\Delta}_{\text{acc}}$ and $\boldsymbol{\Delta}_{\text{gyro}}$ are matrices due to scale factor and misalignments angles (the IMU is not perfectly mounted), $\mathbf{b}_{\text{acc}}$ and $\mathbf{b}_{\text{gyro}}$ are the accelerometer and gyro biases, and $\mathbf{w}_1$ and $\mathbf{w}_2$ are zero mean white noise. Since we are not considering stand-alone INS applications (no GPS updates) and only low-speed applications, it can be assumed that $\boldsymbol{\Delta}_{\text{gyro}} = \boldsymbol{\Delta}_{\text{acc}} = 0$. This is due to the fact that a continuous GPS update at 1–10 Hz will remove these errors terms. In fact this is a feedback system where disturbances are suppressed. For local navigation $\boldsymbol{\omega}_{en}^b$ will be small–i.e., the NED reference frame is not moving with respect to ECEF. For marine vessels it is a good assumption to assume that effects due to the Earth rotation $\boldsymbol{\omega}_{ie}^b$ can be neglected. Hence:

$$\boldsymbol{\omega}_{ib}^b = \boldsymbol{\omega}_{ie}^b + \boldsymbol{\omega}_{en}^b + \boldsymbol{\omega}_{nb}^b \approx \boldsymbol{\omega}_{nb}^b \tag{6.156}$$

and:

$$\mathbf{a}^b \approx \mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}} + \mathbf{g}^b + \mathbf{w}_1 \tag{6.157}$$

$$\boldsymbol{\omega}_{nb}^b \approx \boldsymbol{\omega}_{\text{imu}} + \mathbf{b}_{\text{gyro}} + \mathbf{w}_2 \tag{6.158}$$

In the case of GPS failure $\boldsymbol{\Delta}_{\text{gyro}}$ and $\boldsymbol{\Delta}_{\text{acc}}$ should be estimated on-line to avoid drift. The stand-alone INS solution for this case is found in Vik and Fossen (2001) .

6.3.1 Integration Filter for Position and Linear Velocity

Assuming that the NED reference frame is the inertial system implies that:

$$\dot{\mathbf{v}}^n = \mathbf{a}^n = \mathbf{R}_n^b(\boldsymbol{\Theta})\mathbf{a}^b \tag{6.159}$$

where $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$ is a vector of Euler angles and $\mathbf{R}_n^b(\boldsymbol{\Theta})$ is the rotation matrix between the NED and body-fixed reference frames, see Section 2.2. Substitution of (6.157) into (6.159), yields:

$$\dot{\mathbf{v}}^n \approx \mathbf{R}_b^n(\boldsymbol{\Theta})\left(\mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}} + \mathbf{w}_1\right) + \mathbf{g}^n \tag{6.160}$$

where $\mathbf{g}^n = [0, 0, g]^\top$ is the acceleration of gravity in the NED reference frame and $\mathbf{g}^n = \mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{g}^b$. The general strapdown equations are found in Vik and Fossen (2001).

Integration of IMU and GPS Position Measurements

Consider the following kinematic model:

$$\begin{aligned}
\dot{\mathbf{p}}^n &= \mathbf{v}^n && (6.161)\\
\dot{\mathbf{v}}^n &= \mathbf{R}_b^n(\boldsymbol{\Theta})\left(\mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}} + \mathbf{w}_1\right) + \mathbf{g}^n && (6.162)\\
\dot{\mathbf{b}}_{\text{acc}} &= \mathbf{w}_3 && (6.163)\\
\mathbf{y}_1 &= \mathbf{p}^n && (6.164)
\end{aligned}$$

where $\mathbf{w}_5$ is Gaussian white noise and $\mathbf{y}_1 = [n_{\text{gps}}, e_{\text{gps}}, d_{\text{gps}}]^\top$ is the GPS position measurement vector. Since $E(\mathbf{w}_1) = E(\mathbf{w}_3) = \mathbf{0}$, the following observer structure is proposed:

$$\begin{aligned}
\dot{\hat{\mathbf{p}}}^n &= \hat{\mathbf{v}}^n + \mathbf{K}_1\tilde{\mathbf{y}}_1 && (6.165)\\
\dot{\hat{\mathbf{v}}}^n &= \mathbf{R}_b^n(\boldsymbol{\Theta})\left(\mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}}\right) + \mathbf{g}^n + \mathbf{K}_2\tilde{\mathbf{y}}_1 && (6.166)\\
\dot{\hat{\mathbf{b}}}_{\text{acc}} &= \mathbf{K}_3\mathbf{R}_b^n(\boldsymbol{\Theta})^\top\tilde{\mathbf{y}}_1 && (6.167)\\
\hat{\mathbf{y}}_1 &= \hat{\mathbf{p}}^n && (6.168)
\end{aligned}$$

where $\tilde{\mathbf{y}}_1 = \mathbf{y}_1 - \hat{\mathbf{y}}_1 = \mathbf{p}^n - \hat{\mathbf{p}}^n$. An algorithm for the computation of $\boldsymbol{\Theta}$ from the IMU accelerometers and a compass is presented in Section 6.3.2. The b-frame velocity estimate is therefore computed as:

$$\hat{\mathbf{v}}^b = \mathbf{R}_n^b(\boldsymbol{\Theta})\hat{\mathbf{v}}^n \tag{6.169}$$

The observer error dynamics becomes:

$$\begin{bmatrix} \dot{\tilde{\mathbf{p}}}^n \\ \dot{\tilde{\mathbf{v}}}^n \\ \dot{\tilde{\mathbf{b}}}_{\text{acc}} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_1 & \mathbf{I} & \mathbf{0} \\ -\mathbf{K}_2 & \mathbf{0} & \mathbf{R}_b^n(\boldsymbol{\Theta}) \\ -\mathbf{K}_3\mathbf{R}_b^n(\boldsymbol{\Theta})^\top & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}^n \\ \tilde{\mathbf{v}}^n \\ \tilde{\mathbf{b}}_{\text{acc}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{R}_b^n(\boldsymbol{\Theta}) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_3 \end{bmatrix} \tag{6.170}$$

$$\Updownarrow$$

$$\dot{\mathbf{x}} = \mathbf{A}(\boldsymbol{\Theta})\mathbf{x} + \mathbf{E}\mathbf{w} \tag{6.171}$$

The gains $\mathbf{K}_1, \mathbf{K}_2$, and $\mathbf{K}_3$ must be chosen such that $\mathbf{x}$ converge exponential to zero for $\mathbf{w} = \mathbf{0}$. The main problem in doing this is that the matrix $\mathbf{A}(\boldsymbol{\Theta})$ depends on the attitude $\boldsymbol{\Theta}$. It turns out that this problem can be solved by applying the result of Lindegaard and Fossen (2001*a*).

Property 6.1
A matrix $\mathbf{K} \in \mathbb{R}^{3\times 3}$ *is said to* commute *with the rotation* $\mathbf{R}(\boldsymbol{\Theta})$ *if:*

$$\mathbf{K}\mathbf{R}(\boldsymbol{\Theta}) = \mathbf{R}(\boldsymbol{\Theta})\mathbf{K} \tag{6.172}$$

Examples of $\mathbf{K}$-matrices satisfying Property 6.1 are linear combinations:

$$\mathbf{K} = a_1\mathbf{R}(\boldsymbol{\Theta}) + a_2\mathbf{I} + a_3\mathbf{k}^\top\mathbf{k} \tag{6.173}$$

where

$$\mathbf{k} = [0, 0, 1]^\top \tag{6.174}$$

is the axis of rotation and a_i $(i = 1, \ldots, 3)$ are scalars.

Defining a transformation matrix:

$$\mathbf{T}(\boldsymbol{\Theta}) = \text{diag}\{\mathbf{R}_b^n(\boldsymbol{\Theta}), \mathbf{R}_b^n(\boldsymbol{\Theta}), \mathbf{I}\} \tag{6.175}$$

and a constant system matrix corresponding to (6.170), i.e.:

$$\mathbf{A} = \begin{bmatrix} -\mathbf{K}_1 & \mathbf{I} & \mathbf{0} \\ -\mathbf{K}_2 & \mathbf{0} & \mathbf{I} \\ -\mathbf{K}_3 & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{6.176}$$

implies that:

$$\mathbf{A}(\boldsymbol{\Theta}) = \mathbf{T}(\boldsymbol{\Theta})\mathbf{A}\mathbf{T}^\top(\boldsymbol{\Theta}) \tag{6.177}$$

if the observer gain matrices $\mathbf{K}_i$ $(i = 1, \ldots, 3)$ and $\mathbf{T}_b$ commute with the rotation matrix $\mathbf{R}_b^n(\boldsymbol{\Theta})$. Note that since $\mathbf{R}_b^n(\boldsymbol{\Theta})$ is orthogonal, that is $\mathbf{R}_b^n(\boldsymbol{\Theta})^\top = \mathbf{R}_b^n(\boldsymbol{\Theta})^{-1}$, Property 6.1 implies that:

$$\mathbf{K}_i = \mathbf{R}_b^n(\boldsymbol{\Theta})\mathbf{K}\mathbf{R}_b^n(\boldsymbol{\Theta})^\top \tag{6.178}$$

Consequently, if the matrices $\mathbf{K}_i$ are chosen diagonal, they satisfy (6.173) and stability can be checked by computing the eigenvalues of $\mathbf{A}$. A necessary condition for global exponential stability is that the eigenvalues of $\mathbf{A}$ lie in the left half-plane–i.e., $\mathbf{A}$ must be *Hurwitz*. Notice that the eigenvalues of $\mathbf{A}$ and $\mathbf{A}(\boldsymbol{\Theta})$ are equal.

Matlab:
The following example shows how the observer gains can be computed in Matlab^{TR}.

Example 6.7 (Observer Pole Placement)
The observer gains can be found by pole placement using the following commands:

```
Ao = [0, 1, 0
      0, 0, 1
      0, 0, 0];
C = [1, 0, 0];
K = place(Ao',C',[-1,-2,-3])'
```

resulting in three poles at $-1, -2$ *and* -3 *and* $K = [6, 11, 6]^\top$. *Notice that this system represents two integrators and a constant bias.*

Integration of IMU and GPS Position and Velocity Measurements

It is straightforward to modify the observer (6.165)–(6.168) to included GPS velocity measurements, $\mathbf{y}_2 = \mathbf{v}^n$. Moreover:

$$\dot{\hat{\mathbf{p}}}^n = \hat{\mathbf{v}}^n + \mathbf{K}_{11}\tilde{\mathbf{y}}_1 + \mathbf{K}_{21}\tilde{\mathbf{y}}_2 \tag{6.179}$$

$$\dot{\hat{\mathbf{v}}}^n = \mathbf{R}_b^n(\Theta)\left(\mathbf{f}_{\text{imu}} + \mathbf{b}_{\text{acc}}\right) + \mathbf{g}^n + \mathbf{K}_{12}\tilde{\mathbf{y}}_1 + \mathbf{K}_{22}\tilde{\mathbf{y}}_2 \tag{6.180}$$

$$\dot{\hat{\mathbf{b}}}_{\text{acc}} = \mathbf{K}_{13}\mathbf{R}_b^n(\Theta)^\top \tilde{\mathbf{y}}_1 + \mathbf{K}_{23}\mathbf{R}_b^n(\Theta)^\top \tilde{\mathbf{y}}_2 \tag{6.181}$$

$$\hat{\mathbf{y}}_1 = \hat{\mathbf{p}}^n \tag{6.182}$$

$$\hat{\mathbf{y}}_2 = \hat{\mathbf{v}}^n \tag{6.183}$$

where $\tilde{\mathbf{y}}_i = \mathbf{y}_i - \hat{\mathbf{y}}_i$ $(i = 1, 2)$ results in the error dynamics:

$$\begin{bmatrix} \dot{\tilde{\mathbf{p}}}^n \\ \dot{\tilde{\mathbf{v}}}^n \\ \dot{\tilde{\mathbf{b}}}_{\text{acc}} \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_{11} & \mathbf{I} - \mathbf{K}_{21} & \mathbf{0} \\ -\mathbf{K}_{12} & -\mathbf{K}_{22} & \mathbf{R}_b^n(\Theta) \\ -\mathbf{K}_{13}\mathbf{R}_b^n(\Theta)^\top & -\mathbf{K}_{23}\mathbf{R}_b^n(\Theta)^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}^n \\ \tilde{\mathbf{v}}^n \\ \tilde{\mathbf{b}}_{\text{acc}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{R}_b^n(\Theta) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_3 \end{bmatrix} \tag{6.184}$$

$$\Updownarrow$$

$$\dot{\mathbf{x}} = \mathbf{A}(\Theta)\mathbf{x} + \mathbf{E}\mathbf{w} \tag{6.185}$$

Choosing $\mathbf{K}_{ij}$ $(i = 1, 2, 3, j = 1, 2)$ diagonal so they commute with $\mathbf{R}_b^n(\Theta)$, yields:

$$\dot{\mathbf{x}} = \mathbf{T}(\Theta)\mathbf{A}\mathbf{T}^\top(\Theta)\mathbf{x} + \mathbf{E}\mathbf{w} \tag{6.186}$$

where all $\mathbf{K}_{ij}$ must be chosen such that:

$$\mathbf{A} = \begin{bmatrix} -\mathbf{K}_1 & \mathbf{I} - \mathbf{K}_{21} & \mathbf{0} \\ -\mathbf{K}_2 & -\mathbf{K}_{22} & \mathbf{I} \\ -\mathbf{K}_3 & -\mathbf{K}_{23} & \mathbf{0} \end{bmatrix} \tag{6.187}$$

is *Hurwitz*.

6.3.2 Attitude Observer

Before designing the attitude observer, it will be shown how the three-axes linear IMU accelerations can be transformed to roll and pitch angles. This mapping, together with a compass, is used to construct an Euler angle measurement vector Θ which again is used to remove the drift when the gyro measurements (angular velocities) are integrated; see Figure 6.21.

The attitude observer in this section can be viewed as a special case of Vik and Fossen (2001).

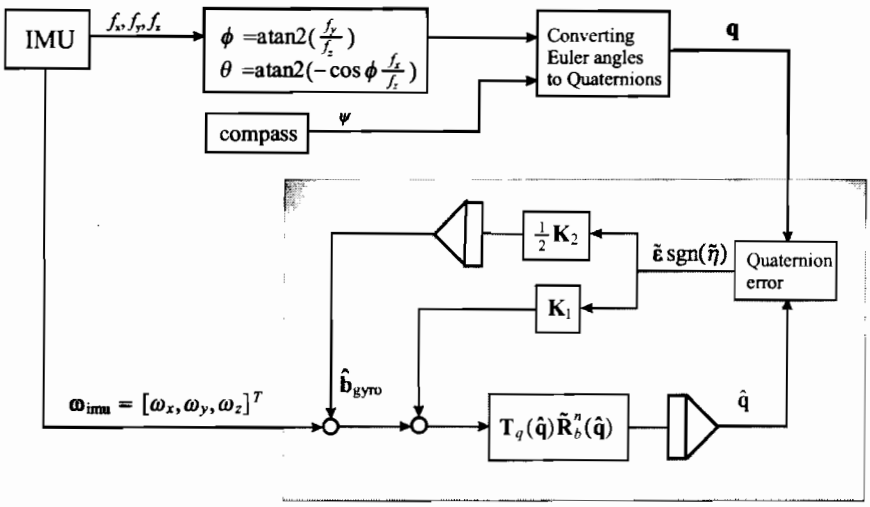


Figure 6.21: Block diagram showing the nonlinear attitude observer with the IMU acceleration mapping.

Mapping from Linear Accelerations to Roll and Pitch Angles

The IMU specific force measurements can be transformed to roll and pitch angles by noticing that the three orthogonal accelerometers mounted in an IMU onboard a vessel at rest produce:

$$\mathbf{f}_{\text{imu}} = \mathbf{R}_n^b(\boldsymbol{\Theta})\mathbf{g}^n \tag{6.188}$$

$$\Updownarrow$$

$$\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \mathbf{R}_n^b(\boldsymbol{\Theta}) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -g\sin\theta \\ g\cos\theta\sin\phi \\ g\cos\theta\cos\phi \end{bmatrix} \tag{6.189}$$

Taking the ratios:

$$\frac{f_y}{f_z} = \tan\phi, \quad \frac{f_x}{f_z} = -\frac{\tan\theta}{\cos\phi}, \quad \cos\phi \neq 0 \tag{6.190}$$

implies that:

$$\phi = \text{atan2}\left(\frac{f_y}{f_z}\right) \tag{6.191}$$

$$\theta = \text{atan2}\left(-\cos\phi\frac{f_x}{f_z}\right) \tag{6.192}$$

Notice that the solution for θ is singular for $\phi = \pm 90$ deg. When combined with a compass measuring the heading ψ the attitude vector $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$ is completely determined. The

Euler angles Θ can easily be transformed to unit quaternion measurements $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top$ by using Algorithm 2.2 from Section 2.2.3. This representation is advantageous when designing the attitude observer.

Quaternion-Based Attitude Observer

In Section 2.2.2 the unit quaternion differential equation was written:

$$\dot{\mathbf{q}} = \mathbf{T}_q(\mathbf{q})\boldsymbol{\omega}_{nb}^b \tag{6.193}$$

with

$$\mathbf{T}_q(\mathbf{q}) = \frac{1}{2}\begin{bmatrix} -\boldsymbol{\varepsilon}^\top \\ \eta\mathbf{I} + \mathbf{S}(\boldsymbol{\varepsilon}) \end{bmatrix} \tag{6.194}$$

Substituting the gyro measurement equation:

$$\boldsymbol{\omega}_{nb}^b \approx \boldsymbol{\omega}_{\text{imu}} + \mathbf{b}_{\text{gyro}} + \mathbf{w}_2 \tag{6.195}$$

into this expression, yields:

$$\dot{\mathbf{q}} = \mathbf{T}_q(\mathbf{q})\left[\boldsymbol{\omega}_{\text{imu}} + \mathbf{b}_{\text{gyro}} + \mathbf{w}_2\right] \tag{6.196}$$

$$\dot{\mathbf{b}}_{\text{gyro}} = \mathbf{w}_4 \tag{6.197}$$

where $\mathbf{b}_{\text{gyro}}$ is the gyro bias. A nonlinear attitude observer for this system is, see Vik and Fossen (2001) and Vik (2000):

$$\dot{\hat{\mathbf{q}}} = \mathbf{T}_q(\hat{\mathbf{q}})\tilde{\mathbf{R}}_b^n(\hat{\mathbf{q}})\left[\boldsymbol{\omega}_{\text{imu}} + \hat{\mathbf{b}}_{\text{gyro}} + \mathbf{K}_1\tilde{\boldsymbol{\varepsilon}}\,\text{sgn}(\tilde{\eta})\right] \tag{6.198}$$

$$\dot{\hat{\mathbf{b}}}_{\text{gyro}} = \frac{1}{2}\mathbf{K}_2\tilde{\boldsymbol{\varepsilon}}\,\text{sgn}(\tilde{\eta}) \tag{6.199}$$

The observer structure is shown in Figure 6.21. The quaternion estimation error is defined as:

$$\tilde{\mathbf{q}} = \hat{\mathbf{q}}^* \otimes \mathbf{q} \tag{6.200}$$

where $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top$ and $\hat{\mathbf{q}}^* = [\hat{\eta}, -\hat{\varepsilon}_1, -\hat{\varepsilon}_2, -\hat{\varepsilon}_3]^\top$ is the conjugate of $\hat{\mathbf{q}}$ – i.e. the vector $\hat{\boldsymbol{\varepsilon}} = [\hat{\varepsilon}_1, \hat{\varepsilon}_2, \hat{\varepsilon}_3]^\top$ is multiplied with -1. The symbol $\otimes$ denotes the *quaternion product* which is defined as (Chou 1992):

$$\begin{aligned} \mathbf{q}_1 \otimes \mathbf{q}_2 &= \begin{bmatrix} \eta_1\eta_2 - \boldsymbol{\varepsilon}_1^\top\boldsymbol{\varepsilon}_2 \\ \eta_2\boldsymbol{\varepsilon}_1 + \eta_1\boldsymbol{\varepsilon}_2 + \boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2 \end{bmatrix} \\ &= \begin{bmatrix} \eta_1 & -\boldsymbol{\varepsilon}_1^\top \\ \boldsymbol{\varepsilon}_1 & \eta_1\mathbf{I} + \mathbf{S}(\boldsymbol{\varepsilon}_1) \end{bmatrix}\mathbf{q}_2 \end{aligned} \tag{6.201}$$

This yields:

$$\tilde{\mathbf{q}} = \begin{bmatrix} \hat{\eta}\eta + \hat{\boldsymbol{\varepsilon}}^\top\boldsymbol{\varepsilon} \\ \eta\hat{\boldsymbol{\varepsilon}} - \hat{\eta}\boldsymbol{\varepsilon} - \hat{\boldsymbol{\varepsilon}}^\top\boldsymbol{\varepsilon} \end{bmatrix} \tag{6.202}$$

Table 6.6: Alternative choices of attitude update laws. The first alternative is GAS while the other two are (local) asymptotically stable due to unstable equilibria.

$H(\tilde{\eta})$	Update law	Stable eq.	Unstable eq.
$1-\lvert\tilde{\eta}\rvert$	$-\mathbf{K}_1\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta})$	$\tilde{\eta}=\pm 1$	
$1-\tilde{\eta}$	$-\mathbf{K}_1\tilde{\varepsilon}$	$\tilde{\eta}=1$	$\tilde{\eta}=-1$
$1+\tilde{\eta}$	$\mathbf{K}_1\tilde{\varepsilon}$	$\tilde{\eta}=-1$	$\tilde{\eta}=1$

Notice that $\tilde{\mathbf{q}} \neq \mathbf{q}-\hat{\mathbf{q}}$. After some tedious calculations, it can be shown that the observer error dynamics becomes (Vik 2000):

$$\dot{\tilde{\mathbf{q}}} = \mathbf{T}(\tilde{\mathbf{q}})\left[\tilde{\mathbf{b}}_{\text{gyro}} - \mathbf{K}_1\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta})\right] \tag{6.203}$$

$$\dot{\tilde{\mathbf{b}}}_{\text{gyro}} = -\frac{1}{2}\mathbf{K}_2\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta}) \tag{6.204}$$

where $\mathbf{K}_i = \mathbf{K}_i^\top > 0\,(i=1,2)$. The expression for $\dot{\tilde{\mathbf{q}}}$ can also be written as:

$$\dot{\tilde{\mathbf{q}}} = \frac{\mathbf{1}}{2}\begin{bmatrix} -\tilde{\varepsilon}^\top \\ \tilde{\eta}\mathbf{I}+\mathbf{S}(\tilde{\varepsilon}) \end{bmatrix}\left[\tilde{\mathbf{b}}_{\text{gyro}} - \mathbf{K}_1\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta})\right] \tag{6.205}$$

Stability of the error dynamics follows from (assuming that $\mathbf{w}_2 = \mathbf{w}_4 = \mathbf{0}$ since these signals are zero mean):

$$V = \frac{1}{2}\tilde{\mathbf{b}}_{\text{gyro}}^\top\mathbf{K}_2^{-1}\tilde{\mathbf{b}}_{\text{gyro}} + H(\tilde{\eta}) \tag{6.206}$$

where different candidates for $H(\tilde{\eta})$ are found in found in Table 6.6 (Fjellstad and Fossen 1994*b*).

Time differentiation of V along the trajectories of $\tilde{\mathbf{b}}_{\text{gyro}}$ and $\tilde{\eta}$, yields:

$$\dot{V} = \tilde{\mathbf{b}}_{\text{gyro}}^\top\mathbf{K}_2^{-1}\dot{\tilde{\mathbf{b}}}_{\text{gyro}} + \frac{\partial H(\tilde{\eta})}{\partial\tilde{\eta}}\dot{\tilde{\eta}} \tag{6.207}$$

Hence, choosing $H(\tilde{\eta}) = 1-\lvert\tilde{\eta}\rvert$, see the first row in Table 6.6, yields:

$$\begin{aligned}\dot{V} &= -\tilde{\mathbf{b}}_{\text{gyro}}^\top\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta}) + \mathrm{sgn}(\tilde{\eta})\tilde{\varepsilon}^\top\left[\tilde{\mathbf{b}}_{\text{gyro}} - \mathbf{K}_1\tilde{\varepsilon}\,\mathrm{sgn}(\tilde{\eta})\right] \\ &= -\tilde{\varepsilon}^\top\mathbf{K}_1\tilde{\varepsilon} \leq 0\end{aligned} \tag{6.208}$$

It can then be concluded that the equilibrium points $\tilde{\eta} = \pm 1$ of the attitude observer error dynamics is asymptotically stable using Krasovskii-LaSalle's theorem; see Appendix A.2.

Vertical Reference Unit (VRU)

The special solution of the observer when only ϕ and θ are estimated (no compass measurement) is referred to as a *vertical reference unit* (VRU). The performance of state-of-the-art VRUs has been evaluated by Ingram *et al.* (1996) .

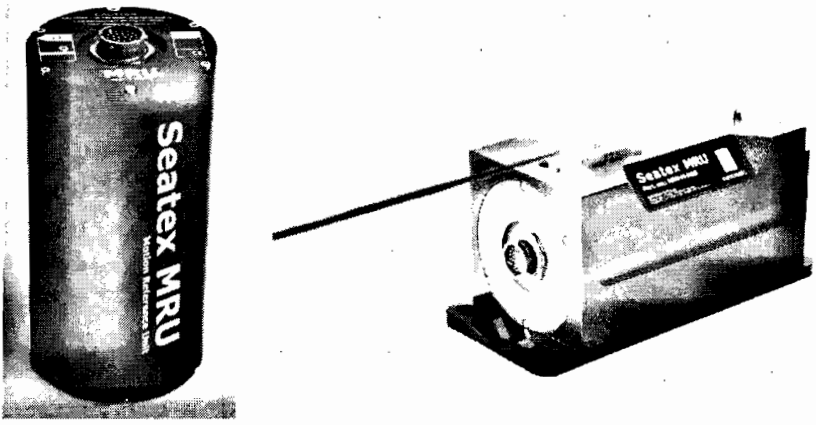


Figure 6.22: The Seatex Motion Reference Unit (MRU). In its simplest form the MRU is a VRU, while the most sophisticated unit is an IMU. Courtesy to Kongsberg Seatex, http://www.seatex.no/.

A VRU is particularly useful if you want to transform the GPS position and velocity measurements $\mathbf{p}^n_{\text{gps}} = [n_{\text{gps}}, d_{\text{gps}}, e_{\text{gps}}]^\top$ (North-East-Down) and $\mathbf{v}^n_{\text{gps}} = [\dot{n}_{\text{gps}}, \dot{d}_{\text{gps}}, \dot{e}_{\text{gps}}]^\top$ for a GPS receiver located at the position $\mathbf{p}^b_{\text{v2g}} = [x_{\text{v2g}}, y_{\text{v2g}}, z_{\text{v2g}}]^\top$ (vector from the vessel origin to the GPS receiver) to the vessel coordinate origin. Moreover, the NED position and linear velocity of the vessel are:

$$\mathbf{p}^n_{\text{vessel}} = \mathbf{p}^n_{\text{gps}} - \mathbf{R}^n_b(\boldsymbol{\Theta})\mathbf{p}^b_{\text{v2g}} \tag{6.209}$$

$$\mathbf{v}^n_{\text{vessel}} = \mathbf{v}^n_{\text{gps}} - \mathbf{R}^n_b(\boldsymbol{\Theta})\mathbf{S}(\boldsymbol{\omega}^b_{nb})\mathbf{p}^b_{\text{v2g}} \tag{6.210}$$

where $\boldsymbol{\Theta} = [\phi, \theta, \psi]^\top$ and $\boldsymbol{\omega}^b_{nb} = [p, q, r]^\top$. Here ϕ and θ are the VRU measurements and ψ is the compass measurement. The NED velocity equation makes use of $\dot{\mathbf{p}}^b_{\text{v2g}} = \mathbf{0}$, that is the position of the GPS receiver onboard the vessel is constant (rigid body). Consequently,

$$\mathbf{v}^n_{\text{vessel}} = \dot{\mathbf{p}}^n_{\text{vessel}} = \dot{\mathbf{p}}^n_{\text{gps}} - \dot{\mathbf{R}}^n_b(\boldsymbol{\Theta})\mathbf{p}^b_{\text{v2g}} \tag{6.211}$$

where $\dot{\mathbf{p}}^n_{\text{gps}} = \mathbf{v}^n_{\text{gps}}$ and $\dot{\mathbf{R}}^n_b(\boldsymbol{\Theta}) = \mathbf{R}^n_b(\boldsymbol{\Theta})\mathbf{S}(\boldsymbol{\omega}^b_{nb})$; see Theorem 2.2 in Section 2.2.1.

6.4 Exercises

Exercise 6.1 *Simulate the ship and wave models in Example 6.3 for different values of K_2 and K_3 with $\sigma = 6.25$. Comment on the results. What happens if there is an uncertainty in ω_0? Moreover, is K_2 and K_3 sensitive for variations in ω_0?*

Exercise 6.2 *Consider the ship model in Example 6.3 and compute the continuous-time steady-state Kalman filter for $\sigma = 6.25$. Simulate the system and compare the results with those obtained in Example 6.3.*

Exercise 6.3 *Show how a continuous-time Kalman filter can be designed for the autopilot model in Example 6.3 when both heading ψ and yaw rate r are measured.*
(Hint: Kalman filter with two measurement equations).

Exercise 6.4 *Explain briefly how wave filtering could be implemented for a small ship and a large tanker in terms of wave frequency, control bandwidth and filter frequency.*

Chapter 7

Control Methods for Marine Vessels

Control design for marine vessels have been an active field of research since the first autopilot was constructed by *Elmer Sperry* in 1911. Modern control systems are based on a variety of design techniques like PID-control, linear quadratic optimal and stochastic control, $\mathcal{H}_\infty$-control methods, fuzzy systems, neural networks and nonlinear control theory to mention some. The presentation in this chapter is, however, limited to methods which have been successfully implemented on-board ships, underwater vehicles, and floating vessels by the author. This includes PID control systems design, linear quadratic optimal control, state feedback linearization, and integrator backstepping.

The system or control models considered are based on (2.1)–(2.2); see Chapters 2 and 3. Only full state feedback is discussed in this chapter. For nonlinear output feedback and state estimation see Nijmeijer and Fossen (1999) and references therein. Kalman filtering as a tool for state estimation is described more closely in Gelb *et al.* (1988), and Brown and Hwang (1998), for instance. Chapter 6 also discusses state estimation for marine systems.

The methods described in this chapter are the foundation for the industrial ROV and ship control systems to be presented in Parts III and IV. These methods have mainly been chosen since the performance have been documented through implementations and full scale experiments.

Linear Versus Nonlinear Control Design

In Section 7.2 conventional linear quadratic optimal control theory with focus on tracking control and disturbance feedforward is reviewed. For this purpose the linearized vessel dy-

namics presented in Section 3.3.2 is applied–,i.e.:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{7.1}$$

This equation is based on several assumptions like zero or constant cruise speed u, together with the assumptions that the velocities v, w, p, q, and r are small. In addition, the kinematic equation $\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}$ must be linearized under a set of assumptions on the Euler angles ϕ, θ, and ψ.

When linearizing the equations of motion, several model properties like symmetry of the inertia matrix, skew-symmetry of the Coriolis and centripetal matrix, and positiveness of the damping matrix are destroyed and this often complicates the control design. Also physical properties which are important tools for good engineering judgement are lost. This is illustrated by comparing the LQ design procedure with the nonlinear techniques in Sections 7.3 and 7.4.

The nonlinear methods are attractive due to their simplicity and design flexibility. The assumptions on u, v, w, p, q, r, and ϕ, θ, ψ which are needed when linearizing the models are also avoided.

Consequently, nonlinear vessel models in the standard form (Fossen 1991):

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{7.2}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{w} \tag{7.3}$$

are used extensively.

It is important to understand the physical properties of the model in order to know which terms in the model that can be omitted when deriving a model-based nonlinear controller. This is an important question since model inaccuracies can destabilize a feedback control system. Often better results are obtained when uncertain terms are chosen to be zero in the controller. Both feedback linearization and backstepping designs will be used to derive nonlinear control systems. In addition, parallels to linear quadratic optimal control are drawn.

7.1 PID-Control and Acceleration Feedback

In this section it is shown how PID-controllers can be designed to exploit acceleration feedback in marine systems. This topic is also covered by Lindegaard (2003) where experimental results with a model ship is used to document performance improvements due to acceleration feedback. Acceleration feedback can be implemented in conjuncture with PID-control without increasing the demand for control energy. A mass-damper-spring system is used to demonstrate the main concept.

7.1.1 Linear Mass-Damper-Spring Systems

Consider the two equivalent systems:

$$m\ddot{x} + d\dot{x} + kx = 0 \tag{7.4}$$

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = 0 \tag{7.5}$$

implying that:

$$2\zeta\omega_n = \frac{d}{m}, \qquad \omega_n^2 = \frac{k}{m} \tag{7.6}$$

For 2nd-order systems it is convenient to define:

$\omega_n = \sqrt{\frac{k}{m}}$ natural frequency (undamped oscillator when $d = 0$)
$\zeta = \frac{d}{2m\omega_n}$ relative damping ratio

Matlab:
The step responses in Figure 7.1 is computed using, see `ExMDS.m`:

```
wn = 1;  % natural frequency

subplot(211)
t = 0:0.01:20;
z = 0.5; sys = tf([wn*wn],[1 2*z*wn wn*wn]); step(sys,t)
hold on
z = 1.0; sys = tf([wn*wn],[1 2*z*wn wn*wn]); step(sys,t)
z = 2.0; sys = tf([wn*wn],[1 2*z*wn wn*wn]); step(sys,t)
hold off

subplot(212)
t = 0:0.01:50;
z = 0.1; sys = tf([wn*wn],[1 2*z*wn wn*wn]); step(sys,t)
hold on
sys = tf([wn*wn],[1 0 wn*wn]); step(sys,t)
hold off
```

Damped Oscillator

For the damped system $d > 0$, the frequency of the oscillation will be smaller than the undamped system. This can be explained by considering the eigenvalues of the mass-damper-spring system (7.5):

$$\lambda_{1,2} = -\underbrace{\zeta\omega_n}_{a} \pm j\omega \tag{7.7}$$

From Figure 7.2 it is seen that:

$$a^2 + \omega^2 = \omega_n^2, \qquad \zeta = \frac{a}{\omega_n} = \cos\phi \tag{7.8}$$

and

a = absolute damping factor
ω = frequency of oscillation (damped system)

The undamped oscillator is obtained by choosing $a = 0$. It is convenient to set:

$$\omega = r\,\omega_n \tag{7.9}$$

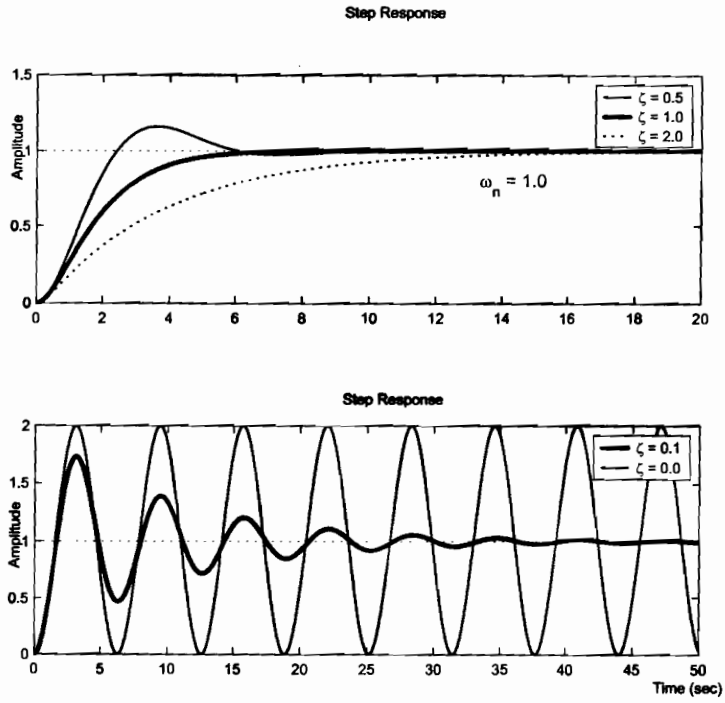


Figure 7.1: The upper plot shows a mass-damper-spring system for different relative damping ratios. The lower plot shows the undamped oscillator together with a damped oscillator. The plots are generated by ExMDS.m.

where r is a reduction factor denoting the ratio between the natural frequency ω_n and the frequency ω of the linearly damped system. For marine vessels a reduction of 0.5% percent in the natural frequency is common (Faltinsen 1990). Hence:

$$r = 1 - \frac{0.5}{100} = 0.995 \tag{7.10}$$

From (7.8) and (7.9) it is seen that:

$$a^2 + (r\omega_n)^2 = \omega_n^2 \tag{7.11}$$

$$\Updownarrow$$

$$a = \underbrace{\sqrt{1-r^2}}_{\zeta}\,\omega_n \tag{7.12}$$

and further:

$$\begin{aligned} \frac{d}{m} &= 2\zeta\omega_n \\ &= 2\sqrt{1-r^2}\omega_n \\ &= 2\sqrt{1-r^2}\sqrt{\frac{k}{m}} \end{aligned} \tag{7.13}$$

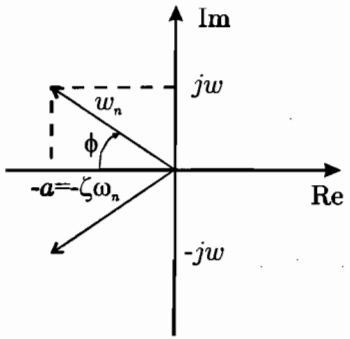


Figure 7.2: Grapical illustration of natural frequency ω_n, frequency of the damped system ω, and absolute damping factor a.

which yields the following formula for linear damping:

$$d = 2\sqrt{1-r^2}\sqrt{km} \tag{7.14}$$

This formula is quite useful to determine the linear damping in *heave, roll,* and *pitch* of an uncontrolled marine vessel (open loop) since the mass m and spring (metacentric) coefficient k are easily obtained by other methods; see Sections 3.4.2 and 3.4.5. The frequency of oscillation is:

$$\omega = \sqrt{\frac{k}{m} - \left(\frac{d}{2m}\right)^2} \tag{7.15}$$

which for $d = 0$ reduces to the natural frequency of the undamped oscillator:

$$\omega \overset{d=0}{=} \sqrt{\frac{k}{m}} = \omega_n \tag{7.16}$$

Damping in *surge, sway,* and *yaw* can, however, not be determined by Formula (7.14) since $k = 0$ in a pure *mass-damper* system. Linear damping for such a system:

$$m\ddot{x} + d\dot{x} = \tau \tag{7.17}$$

can be found by specifying the time constant $T > 0$. Let $T = m/d$ such that (7.17) becomes:

$$T\ddot{x} + \dot{x} = \frac{1}{d}\tau \tag{7.18}$$

which yields the following design formula for the *mass-damper*:

$$d = \frac{m}{T} \tag{7.19}$$

Equations (7.14) and (7.19) will be referred to as the *linear damping formulas* for a *mass-damper-spring* and *mass-damper* system, respectively.

Example 7.1 (Linear Damping in Roll and Pitch for Submarines)
Consider the linear pitch equation (3.274):

$$(I_y - M_{\dot{q}})\ddot{\theta} - M_q\dot{\theta} + BG_z\, W\, \theta = \tau_5$$

Hence, linear damping can be computed by using (7.14):

$$-M_q = 2\sqrt{1-r^2}\,\sqrt{BG_z\, W(I_y - M_{\dot{q}})} > 0$$

where $M_{\dot{q}}, W$ and BG_z are assumed to be known, and $r > 0$ is a design parameter. For roll a similar expression is obtained, see (3.281):

$$-K_p = 2\sqrt{1-r^2}\,\sqrt{BG_z\, W(I_x - K_{\dot{p}})} > 0$$

7.1.2 Acceleration Feedback

Consider a mass damper spring system:

$$m\ddot{x} + d\dot{x} + kx = \tau + w \tag{7.20}$$

Let the control law be:

$$\tau = \tau_{PID} - K_m\ddot{x} \tag{7.21}$$

where $K_m > 0$ is the acceleration feedback gain and τ_{PID} represents a conventional PID-controller. This yields:

$$(m + K_m)\ddot{x} + d\dot{x} + kx = \tau_{PID} + w \tag{7.22}$$

or equivalently:

$$\ddot{x} + \frac{d}{m+K_m}\dot{x} + \frac{k}{m+K_m}x = \frac{1}{m+K_m}\tau_{PID} + \frac{1}{m+K_m}w \tag{7.23}$$

From this expression it is noticed that besides increasing the mass from m to $m + K_m$, acceleration feedback also reduces the gain in front of the disturbance w from $1/m$ to $1/(m+K_m)$. Hence, the system is expected to be less sensitive to an external disturbances w if acceleration feedback is applied.

This design can be further improved by introducing a frequency dependent *virtual mass* (Sagatun *et al.* 2001), that is:

$$\tau = \tau_{PID} - h_m(s)\ddot{x} \tag{7.24}$$

If $h_m(s)$ is chosen as a low-pass filter:

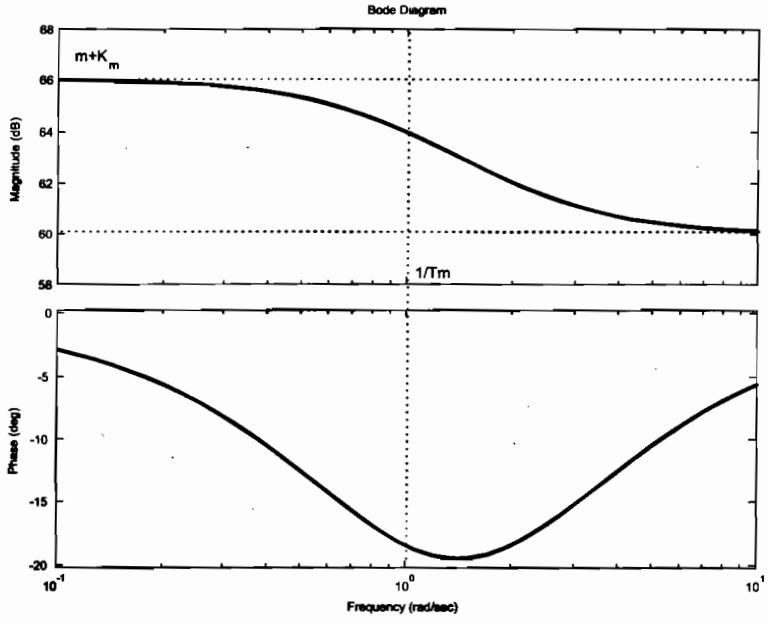


Figure 7.3: Frequency dependent mass $m_{\text{total}}(s)$ for $m = K_m = 1000 = 60$ (dB) and $T_m = 1$ (s).

$$h_m(s) = \frac{K_m}{1 + T_m s} \tag{7.25}$$

with gain $K_m > 0$ and time constant $T_m > 0$, it is seen that:

$$\underbrace{\left(m + \frac{K_m}{1 + T_m s}\right)}_{m_{\text{total}}(s)} \ddot{x} + d\dot{x} + kx = \tau_{PID} + w \tag{7.26}$$

where the total mass of the system in closed loop is:

$$m_{\text{total}}(s) = m + \frac{K_m}{1 + T_m s} = \frac{mT_m s + (m + K_m)}{T_m s + 1} \tag{7.27}$$

Hence, it can be concluded that the total mass is $m + K_m$ at low frequencies ($s \to 0$) while at high frequencies ($s \to \infty$) the total mass $m + K_m$ reduces to m. This is shown in Figure 7.3.

The filter $h_m(s)$ can be chosen rather arbitrarily depending on the application. For instance, a low-pass filter will remove high frequency acceleration feedback components while a notch structure can be used to remove 1st-order wave-induced disturbances. This is seen by letting:

$$g(s) = \frac{1}{1 + h_m(s)} \tag{7.28}$$

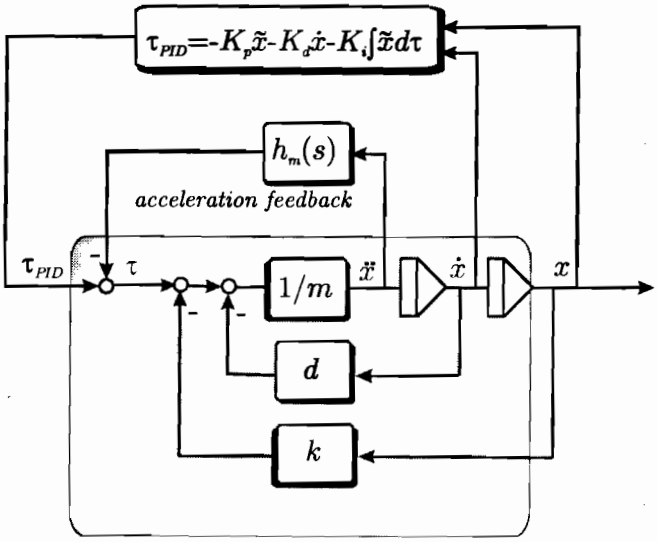


Figure 7.4: Acceleration feedback (inner loop) and PID feedback (outer loop).

such that (7.26) takes the form:

$$\ddot{x} + g(s)d\dot{x} + g(s)kx = g(s)\tau_{PID} + g(s)w \tag{7.29}$$

where $g(s)$ is chosen such that the disturbance w is suppressed in a limited frequency band (low-pass, high-pass, notch etc.).

It will next be shown how a PID-controller can be designed independently of the acceleration feedback loop.

7.1.3 Acceleration Feedback + PID Control

Consider a PID-controller:

$$\tau_{PID} = -\left(K_p\tilde{x} + K_d\dot{x} + K_i\int_o^t \tilde{x}(\tau)d\tau\right) \tag{7.30}$$

with gains $K_p > 0, K_d > 0$, and $K_i > 0$.

For simplicity, assume that $h_m(s) = K_m$ and $K_i = 0$ in (7.24). This gives a conventional PD-controller with fixed gain acceleration feedback:

$$\tau = -\underbrace{(K_p\tilde{x} + K_d\dot{x})}_{\text{PD-controller}} \underbrace{-K_m\ddot{x}}_{\substack{\text{acceleration} \\ \text{feedback}}} \tag{7.31}$$

The closed-loop system becomes:

$$(m + K_m)\ddot{x} + (d + K_d)\dot{x} + (k + K_p)x = w \tag{7.32}$$

such that:

$$\omega_n = \sqrt{\frac{k + K_p}{m + K_m}} \tag{7.33}$$

$$\zeta = \frac{d + K_d}{2(m + K_m)\omega_n} \tag{7.34}$$

Pole placement of the mass-damper-spring system suggests that K_p and K_d can be computed by specifying ω_n and ζ in (7.33) and (7.34). Solving for K_p and K_d, yields:

$$K_p = (m + K_m)\omega_n^2 - k \tag{7.35}$$

$$K_d = 2\zeta\omega_n(m + K_m) - d \tag{7.36}$$

such that (7.32) becomes:

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = \omega_n^2 x_d + \frac{1}{m + K_m} w \tag{7.37}$$

$$\Downarrow \quad \{m + K_m \gg 1\}$$

$$\frac{x}{x_d}(s) \approx \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{7.38}$$

This is a good approximation for $m + K_m \gg 1$. An even better approach is to add integral action $K_i > 0$ to compensate for a large constant disturbance w. Let the PID-controller be written as:

$$\tau = \underbrace{-K_p\left(1 + T_d s + \frac{1}{T_i s}\right)\tilde{x}}_{\text{PID}} \quad \underbrace{-K_m\ddot{x}}_{\substack{\text{acceleration} \\ \text{feedback}}} \tag{7.39}$$

where $T_d = K_d/K_p$ and $T_i = K_p/K_i$ are the derivative and integral time constants, respectively. A *rule-of-thumb* is to choose:

$$\frac{1}{T_i} \approx \frac{\omega_n}{10} \tag{7.40}$$

which states that the integrator is 10 times slower than the natural frequency ω_n. This yields:

$$K_i = \frac{\omega_n}{10} K_p = \frac{\omega_n}{10}\left[(m + K_m)\omega_n^2 - k\right] \tag{7.41}$$

The natural frequency ω_n can be related to the system bandwidth ω_b by using the following definition:

Definition 7.1 (Control Bandwidth)
The control bandwidth of a system $y = h(s)u$ with negative unity feedback is defined as the frequency ω_b (rad/s) at which the loop transfer function $l(s) = h(s) \cdot 1$ is:

$$|l(j\omega)|_{\omega=\omega_b} = \frac{\sqrt{2}}{2}$$

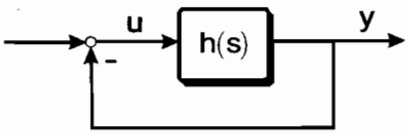


Figure 7.5: Closed-loop feedback system.

or equivalently,

$$20 \log |l(j\omega)|_{\omega=\omega_b} = -3 \; dB$$

From this definition it can be shown that the control bandwidth of a 2nd-order system with natural frequency ω_n and relative damping ratio ζ:

$$h(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{7.42}$$

with negative unity feedback is (see Figure 7.5):

$$\omega_b = \omega_n \sqrt{1 - 2\zeta^2 + \sqrt{4\zeta^4 - 4\zeta^2 + 2}} \tag{7.43}$$

For a critically damped system, $\zeta = 1.0$, this expression reduces to:

$$\omega_b = \omega_n \sqrt{\sqrt{2} - 1}\, \omega_n \approx 0.64\, \omega_n \tag{7.44}$$

Pole Placement Algorithm (PID and Acceleration Feedback)

1. Specify the bandwidth $\omega_b > 0$ and the relative damping ratio $\zeta > 0$.
2. Compute the natural frequency: $\omega_n = \frac{1}{\sqrt{1-2\zeta^2+\sqrt{4\zeta^4-4\zeta^2+2}}}\omega_b$
3. Specify the gain: $K_m \geq 0$ (optionally acceleration feedback)
4. Compute the P-gain: $K_p = (m + K_m)\omega_n^2 - k$
5. Compute the D-gain: $K_d = 2\zeta\omega_n(m + K_m) - d$
6. Compute the I-gain: $K_i = \frac{\omega_n}{10} K_p$

Example 7.2 (Ship Autopilot Design)
Consider the Nomoto model (Nomoto et al. *1957):*

$$T\ddot{\psi} + \dot{\psi} = K\delta \tag{7.45}$$

where ψ is the yaw angle and δ is the rudder angle (control input). From (7.20) it is seen that:

$$m = \frac{T}{K}, \quad d = \frac{1}{K}, \quad k = 0 \tag{7.46}$$

The PID and acceleration feedback controller gains are found by using pole placement in

terms of the design parameters K_m, ω_n, and ζ, resulting in:

$$\begin{aligned} K_m &\geq 0 \\ K_p &= \frac{T + KK_m}{K}\omega_n^2 > 0 \\ K_d &= \frac{T + KK_m}{K}2\zeta\omega_n - \frac{1}{K} > 0 \\ K_i &= \frac{T + KK_m}{10K}\omega_n^3 > 0 \end{aligned}$$

For $K_m = 0$ (no yaw rate feedback) this reduces to a conventional autopilot of PID-type with gains:

$$\begin{aligned} K_p &= \frac{\omega_n^2 T}{K} > 0 \\ K_d &= \frac{2\zeta\omega_n T - 1}{K} > 0 \\ K_i &= \frac{\omega_n^3 T}{10K} > 0 \end{aligned}$$

7.1.4 MIMO Acceleration Feedback and Nonlinear PID Control

The PID control concept can be generalized to nonlinear mechanical system by exploiting the kinematic equations of motion in the design. Consider the nonlinear model:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{7.47}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \mathbf{w} \tag{7.48}$$

where $\boldsymbol{\eta}$ and $\boldsymbol{\nu}$ are assumed to be measured. Consider the control law:

$$\boldsymbol{\tau} = \mathbf{g}(\boldsymbol{\eta}) - \mathbf{H}_m(\mathrm{s})\dot{\boldsymbol{\nu}} - \mathbf{J}^\top(\boldsymbol{\eta})\boldsymbol{\tau}_{\mathrm{PID}} \tag{7.49}$$

with acceleration feedback $\mathbf{H}_m(\mathrm{s})\dot{\boldsymbol{\nu}}$, gravity compensation $\mathbf{g}(\boldsymbol{\eta})$, and PID-controller:

$$\boldsymbol{\tau}_{\mathrm{PID}} = \mathbf{K}_p\tilde{\boldsymbol{\eta}} + \mathbf{K}_d\dot{\boldsymbol{\eta}} + \mathbf{K}_i\int_o^t \tilde{\boldsymbol{\eta}}(\tau)d\tau \tag{7.50}$$

For simplicity, assume that $\mathbf{K}_i = \mathbf{0}$ and $\mathbf{H}_m(s) = \mathbf{K}_m$ (PD-control with fixed gain acceleration feedback). This yields the closed-loop system:

$$\mathbf{H}\dot{\boldsymbol{\nu}} + [\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})]\boldsymbol{\nu} + \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}} = \mathbf{w} \tag{7.51}$$

where $\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \boldsymbol{\eta}_d$,

$$\mathbf{K}_d^*(\boldsymbol{\eta}) = \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_d\mathbf{J}(\boldsymbol{\eta}) \tag{7.52}$$

and:

$$\mathbf{H} = \mathbf{M} + \mathbf{K}_m$$

In the analysis it is assumed that $\dot{\boldsymbol{\eta}}_d = \mathbf{0}$, that is, regulation of $\boldsymbol{\eta}$ to $\boldsymbol{\eta}_d$ = constant. A Lyapunov function candidate for this system is:

$$V = \underbrace{\frac{1}{2}\boldsymbol{\nu}^\top \mathbf{H}\boldsymbol{\nu}}_{\substack{\text{kinetic} \\ \text{energy}}} + \underbrace{\frac{1}{2}\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}}}_{\substack{\text{potential} \\ \text{energy}}} \tag{7.53}$$

For marine vessels at *low speed* it can be assumed that:

$$\mathbf{M} = \mathbf{M}^\top > 0$$

However, this is not true at higher speeds e.g. in a maneuvering situation; see Property 3.3 in Section 3.2.1. Lyapunov control design for systems with a *nonsymmetric inertia matrix* is discussed in Section 7.1.5.

Time differentiation of (7.53) along the trajectories of $\boldsymbol{\nu}$ and $\tilde{\boldsymbol{\eta}}$ under the assumptions that $\mathbf{M} = \mathbf{M}^\top$, $\mathbf{K}_p = \mathbf{K}_p^\top$ and $\mathbf{K}_m = \mathbf{K}_m^\top$, yields:

$$\begin{aligned} \dot{V} &= \boldsymbol{\nu}^\top \mathbf{H}\dot{\boldsymbol{\nu}} + \dot{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}} \\ &= \boldsymbol{\nu}^\top (\mathbf{H}\dot{\boldsymbol{\nu}} + \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p \tilde{\boldsymbol{\eta}}) \end{aligned} \tag{7.54}$$

since $\dot{\tilde{\boldsymbol{\eta}}} = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d = \dot{\boldsymbol{\eta}}$ and $\dot{\boldsymbol{\eta}}^\top = \boldsymbol{\nu}^\top \mathbf{J}^\top(\boldsymbol{\eta})$. Substituting (7.51) into (7.54) yields:

$$\begin{aligned} \dot{V} &= \boldsymbol{\nu}^\top (\mathbf{w} - [\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})]\boldsymbol{\nu}) \\ &= \boldsymbol{\nu}^\top \mathbf{w} - \boldsymbol{\nu}^\top [\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})]\, \boldsymbol{\nu} \end{aligned} \tag{7.55}$$

since $\boldsymbol{\nu}^\top \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} = 0$ for all $\boldsymbol{\nu}$; see Property 3.7 in Section 3.3.

If $\mathbf{w} = \mathbf{0}$, *Krasovskii–LaSalle's* Theorem A.2 in Appendix A.1 can be used to prove that the system (7.47)–(7.48) with nonlinear PD-control ($\mathbf{K}_i = \mathbf{0}$) is *globally asymptotically stable* (GAS). Moreover, the trajectories will converge to the set Ω found from:

$$\dot{V}(\mathbf{x}) = -\boldsymbol{\nu}^\top \left[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\eta})\right] \boldsymbol{\nu} \equiv 0 \tag{7.56}$$

which is true for $\boldsymbol{\nu} = \mathbf{0}$. Therefore:

$$\Omega = \{(\tilde{\boldsymbol{\eta}}, \boldsymbol{\nu}) : \boldsymbol{\nu} = \mathbf{0})\} \tag{7.57}$$

Now, $\boldsymbol{\nu} \equiv \mathbf{0}$ implies that $\mathbf{H}\dot{\boldsymbol{\nu}} = -\mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p \tilde{\boldsymbol{\eta}}$ which is non-zero as long as $\tilde{\boldsymbol{\eta}} \neq \mathbf{0}$. Hence, the system cannot get "stuck" at an equilibrium point value other than $\tilde{\boldsymbol{\eta}} = \mathbf{0}$. Since the equilibrium point $(\tilde{\boldsymbol{\eta}}, \boldsymbol{\nu}) = (\mathbf{0}, \mathbf{0})$ is the largest invariant set M in Ω, the equilibrium point is GAS according to Theorem A.2.

In the case $\mathbf{w} \neq \mathbf{0}$ but $\dot{\mathbf{w}} = \mathbf{0}$, the system trajectories will converge to a ball about the origin $(\tilde{\boldsymbol{\eta}}, \boldsymbol{\nu}) = (\mathbf{0}, \mathbf{0})$. The radius of the ball depends on the magnitude of the disturbance $\mathbf{w}$. This is referred to as uniform ultimate boundedness (UUB).

If integral action is included with $\mathbf{K}_i > 0$ (PID-control), it is possible to prove local asymptotic stability (LAS) also for the case $\mathbf{w} \neq \mathbf{0}$. This result is well known from robotics (Arimoto and Miyazaki 1984). The bias term $\mathbf{w}$ can also be removed by using parameter adaptation (Fossen *et al.* 2001).

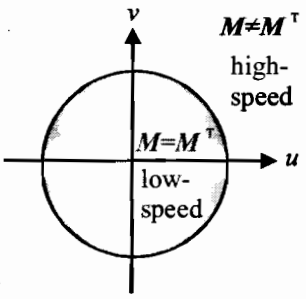


Figure 7.6: Low and high speed regimes for a ship. The total speed is $U = \sqrt{u^2 + v^2}$ where u and v are the velocity in surge and sway.

7.1.5 Inertia Shaping Techniques using Acceleration Feedback

The assumptions that $\mathbf{M} = \mathbf{M}^\top > 0$ can be relaxed in order to describe both low-speed applications and maneuvering situations (Fossen *et al.* 2002). This is a non-trivial problem since the system inertia matrix will be nonsymmetrical for marine vessels moving at high speed while it is symmetric at zero speed (station-keeping); see Figure 7.6.

The main idea is to modify the system inertia matrix of a marine vessel through acceleration feedback. The problem of applying the kinetic energy of a system with nonsymmetric inertia matrix as a Lyapunov function candidate is easiest explained by considering the following case study:

Case Study: Nonsymmetric Inertia Matrix Consider the problem of energy-based control when the system inertia $\mathbf{M}$ is nonsymmetrical due to hydrodynamic added mass. Moreover, for marine vessels in transit (non-zero speed) it can be shown that:

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A \tag{7.58}$$

where the rigid-body system inertia matrix $\mathbf{M}_{RB} = \mathbf{M}_{RB}^\top > 0$ and the hydrodynamic added inertia matrix $\mathbf{M}_A$ is nonsymmetrical (Property 3.3 in Section 3.2.1), that is:

$$\mathbf{M}_A \neq \mathbf{M}_A^\top > 0 \tag{7.59}$$

Notice that $\mathbf{M}_A$ is nonsymmetrical due to forward speed effects and wave-induced disturbances (Salvesen *et al.* 1970). This implies that the kinetic energy can be written:

$$\begin{aligned} V &= \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu} \\ &= \frac{1}{2}\boldsymbol{\nu}^\top \left(\frac{1}{2}(\mathbf{M}+\mathbf{M}^\top)+\frac{1}{2}(\mathbf{M}-\mathbf{M}^\top)\right)\boldsymbol{\nu} \\ &= \frac{1}{4}\boldsymbol{\nu}^\top (\mathbf{M}+\mathbf{M}^\top)\boldsymbol{\nu} \end{aligned} \tag{7.60}$$

where $\frac{1}{2}(\mathbf{M}+\mathbf{M}^\top)$ is the *symmetrical* and $\frac{1}{2}(\mathbf{M}-\mathbf{M}^\top)$ is the *skew-symmetrical* parts of $\mathbf{M}$, respectively. Consequently, $\frac{1}{2}\boldsymbol{\nu}^\top(\mathbf{M}-\mathbf{M}^\top)\boldsymbol{\nu} = 0$ for all $\boldsymbol{\nu}$.

Next, time differentiation along the trajectories of $\boldsymbol{\nu}$ yields:

$$\dot{V} = \frac{1}{2}\boldsymbol{\nu}^\top(\mathbf{M} + \mathbf{M}^\top)\dot{\boldsymbol{\nu}} \tag{7.61}$$

This approach fails for vessel models in the form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}(\boldsymbol{\nu}) = \boldsymbol{\tau} \tag{7.62}$$

where $\mathbf{n}(\boldsymbol{\nu})$ is a vector of nonlinear Coriolis, damping, and restoring terms and $\boldsymbol{\tau}$ is the control input. The main reason is that only $\mathbf{M}\dot{\boldsymbol{\nu}}$ in the expression for $\dot{V}$ can be be substituted from the system model (7.62) while the expression $\mathbf{M}^\top\dot{\boldsymbol{\nu}}$ is not available from (7.62).

This problem can be solved by using acceleration feedback to shape the system inertia matrix. The most important shaping technique will be symmetrization of

$$\mathbf{H} = \underbrace{\mathbf{M}_{RB} + \mathbf{M}_A}_{\mathbf{M}} + \mathbf{K}_m \tag{7.63}$$

through $\mathbf{K}_m$. Two design techniques for inertia symmetrization will be discussed:

Positive acceleration feedback (decreasing the system inertia)
A symmetric system inertia matrix is obtained by positive acceleration feedback:

$$\mathbf{K}_m = -\mathbf{M}_A < 0 \tag{7.64}$$

which yields

$$\mathbf{H} = \mathbf{H}^\top = \mathbf{M}_{RB} > 0 \tag{7.65}$$

Positive feedback in the inner acceleration loop will not destabilize the system since $\mathbf{M}_{RB} > 0$. However, if $\mathbf{M}_A$ is uncertain, positive feedback $\mathbf{K}_m = -\mathbf{M}_A$ might destabilize the system if the uncertainty is in the same magnitude as the norm of $\mathbf{M}_{RB}$ since this may lead to $\mathbf{H} < 0$. Therefore, negative acceleration feedback should rather be applied to avoid robustness problems.

Negative acceleration feedback (increasing the system inertia)
The system inertia can be increased by applying negative acceleration feedback:

$$\mathbf{K}_m = \frac{1}{2}\left(\mathbf{M}_m - \mathbf{M}_m^\top\right) + \Delta\mathbf{K} > 0 \quad \text{where } \Delta\mathbf{K} = \Delta\mathbf{K}^\top \geq 0$$

resulting in:

$$\mathbf{H} = \mathbf{H}^\top = \mathbf{M}_{RB} + \frac{1}{2}\left(\mathbf{M}_m + \mathbf{M}_m^\top\right) + \Delta\mathbf{K} > 0$$

The gain matrix $\Delta\mathbf{K}$ can be used to increase the system inertia further since the feedback term $\frac{1}{2}(\mathbf{M}_m - \mathbf{M}_m^\top)\dot{\boldsymbol{\nu}}$ ensures symmetrization. Clearly, if the inertia is increased by acceleration feedback, the closed loop system will be less sensitive to external disturbances, see Lindegaard (2003), for instance.

The symmetrization technique of this section suggest that the control law:

$$\tau = \mathbf{g}(\eta) - \mathbf{K}_m \dot{\nu} - \mathbf{J}^\top(\eta)\tau_{\text{PID}} \tag{7.66}$$

$$\tau_{\text{PID}} = -\mathbf{K}_p\tilde{\eta} - \mathbf{K}_d\dot{\eta} - \mathbf{K}_i \int_o^t \tilde{\eta}(\tau)d\tau \tag{7.67}$$

can be analyzed by using standard Lyapunov analysis based using (7.53) since $\mathbf{H} = \mathbf{H}^\top > 0$.

A special solution exists for the horizontal motion of a vessel (surge, sway, and yaw) since only two linear accelerometers (surge and sway) are required to symmetrize the inertia matrix. This solution is attractive both in dynamic positioning and in particular in maneuvering situations where $\mathbf{M}_A \neq \mathbf{M}_A{}^\top$. The design philosophy is demonstrated in Section 7.4.8 where a ship maneuvering system is designed using backstepping and acceleration feedback in surge and sway.

7.2 Linear Quadratic Optimal Control

In this chapter, we will briefly review some results from Athans and Falb (1966) on linear quadratic (LQ) optimal control theory.

Linearized Vessel Model

As shown in Section 3.3.2 the 6 DOF nonlinear dynamics (2.2) can be approximated by a linear system:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{7.68}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{7.69}$$

where $\mathbf{x} = [\eta_p^\top, \nu^\top]^\top$ and $\mathbf{u} = \tau$ under the assumptions of small roll and pitch angles, and small speed (alternatively constant cruise speed).

The system performance output $\mathbf{y}$ is defined by specifying the matrix $\mathbf{C}$. For instance, position control is obtained by choosing:

$$\mathbf{C} = \begin{bmatrix} \mathbf{I}_{6\times 6} & \mathbf{0}_{6\times 6} \end{bmatrix} \tag{7.70}$$

The *NED* positions are computed from the vessel parallel coordinates (see Section 3.3.2):

$$\eta = \mathbf{P}(\psi)\eta_p = \begin{bmatrix} \mathbf{R}(\psi) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{I}_{3\times 3} \end{bmatrix} \eta_p \tag{7.71}$$

In order to design a linear control law the system $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ must be controllable while observability (see Definition 6.2 in Section 6.1.3) is necessary if some of the states must be estimated. Controllability for linear time-invariant systems is given by the following definition:

Definition 7.2 (Controllability)
The state and input matrix $(\mathbf{A}, \mathbf{B})$ *must satisfy the* controllability condition *to ensure that there exists a control* $\mathbf{u}(t)$ *which can drive any arbitrary state* $\mathbf{x}(t_0)$ *to another arbitrary state* $\mathbf{x}(t_1)$ *for* $t_1 > t_0$. *The* controllability condition *requires that the matrix (Gelb* et al. *1988):*

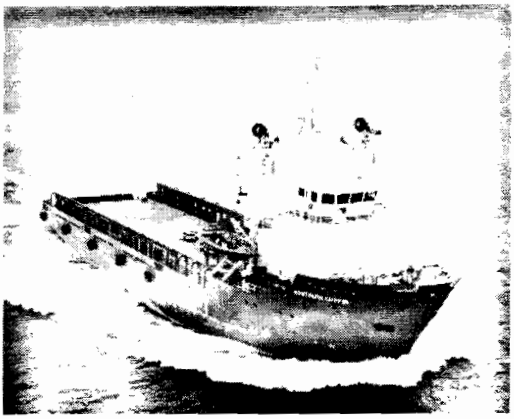


Figure 7.7: Supply vessel of length $L = 76.2$ (m).

$$\mathcal{C} = [\mathbf{B} \mid \mathbf{AB} \mid ... \mid (\mathbf{A})^{n-1}\mathbf{B}] \tag{7.72}$$

must be of full row rank such that a least a right inverse exists.

Matlab:
The following example demonstrates how observability and controllability can be checked for a ship in surge, sway and yaw.

Example 7.3 (Observability and Controllability of Ships)
Consider the supply vessel in Figure 7.7. The non-dimensional system matrices are (Fossen et al. *1996):*

$$\mathbf{M}'' = \begin{bmatrix} 1.1274 & 0 & 0 \\ 0 & 1.8902 & -0.0744 \\ 0 & -0.0744 & 0.1278 \end{bmatrix}, \mathbf{D}'' = \begin{bmatrix} 0.0358 & 0 & 0 \\ 0 & 0.1183 & -0.0124 \\ 0 & -0.0041 & 0.0308 \end{bmatrix} \tag{7.73}$$

These values are defined in accordance to the Bis-system such that:

$$\mathbf{M} = \mathbf{T}\mathbf{M}''\mathbf{T}^{-1}, \qquad \mathbf{D} = \sqrt{g/L}\,\mathbf{T}\mathbf{D}''\mathbf{T}^{-1} \tag{7.74}$$

where $\mathbf{T} = \text{diag}\{1, 1, L\}$*; see Section 8.1.3. The linear state-space model in surge, sway and yaw is obtained as (see Section 3.3.2):*

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{M}^{-1} \end{bmatrix}, \quad \mathbf{C} = [\mathbf{I}_{3\times3}, \mathbf{0}_{3\times3}] \tag{7.75}$$

where $\mathbf{x} = [\boldsymbol{\eta}_p^\top, \boldsymbol{\nu}^\top]^\top$ *and* $\mathbf{u} = \boldsymbol{\tau}$*. The number of states are* $n = 6$*. Notice that only the positions* (n, e) *and yaw angle* ψ *are defined as the outputs. Observability and*

controllability can be checked in Matlab™ using the commands; see `ExObsCtr.m`:

```
n_obs = rank(obsv(A,C))
n_ctr = rank(ctrb(A,B))
```

Since $n\_obs = n\_ctr = n = 6$ *the supply vessel is both observable and controllable.*

7.2.1 Linear Quadratic Regulator

A fundamental design problem is the regulator problem, where it is necessary to regulate the outputs $\mathbf{y} \in \mathbb{R}^m$ of the system to zero or a constant value while ensuring that they satisfy time-response specifications. A linear quadratic regulator (LQR) can be designed for this purpose by considering the state-space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{7.76}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{7.77}$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^r$, and $\mathbf{y} \in \mathbb{R}^m$. The feedback control law is found by minimizing the performance index:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T \left(\mathbf{y}^\top \mathbf{Q}\mathbf{y} + \mathbf{u}^\top \mathbf{R}\mathbf{u} \right) dt \right.$$
$$\left. = \frac{1}{2} \int_0^T \left(\mathbf{x}^\top \mathbf{C}^\top \mathbf{Q}\mathbf{C}\mathbf{x} + \mathbf{u}^\top \mathbf{R}\mathbf{u} \right) \, dt \right\} \tag{7.78}$$

where $\mathbf{R} = \mathbf{R}^\top > 0$ and $\mathbf{Q} = \mathbf{Q}^\top \geq 0$ are the weighting matrices. The steady-state solution to this problem is (Athans and Falb 1966):

$$\mathbf{u} = \underbrace{-\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty}_{\mathbf{G}} \mathbf{x} \tag{7.79}$$

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty + \mathbf{C}^\top \mathbf{Q}\mathbf{C} = \mathbf{0} \tag{7.80}$$

where $\mathbf{P}_\infty = \lim_{t \to \infty} \mathbf{P}(t)$.

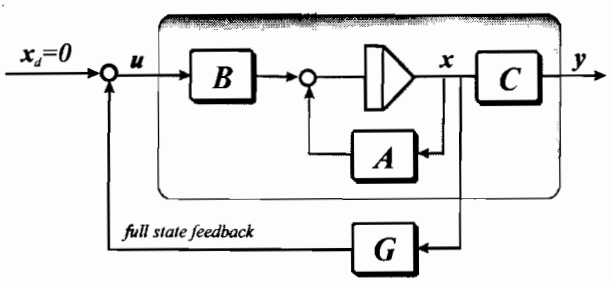


Figure 7.8: Block diagram showing the linear quadratic regulator (LQR).

Matlab:
The steady-state LQR feedback control law is computed as; see the script `ExLQR.m`:

```
Q = diag([1]);  % user editable tracking error weights (dim m x m)
R = diag([1]);  % user editable input weights (dim r x r)

% System matrices
A = [0 1; -1 -2];    % user editable state matrix (dim n x n)
B = [0; 1];          % user editable input matrix (dim n x r)
C = [1 0];           % user editable output matrix (dim m x n)

% Compute the optimal feedback gain matrix G
[K,P,E] = lqr(A,B,C'*Q*C,R);
G = -K
```

The Matlab™ function `lqr.m` also returns the eigenvalues of the closed-loop system:

$$\dot{\mathbf{x}} = (\mathbf{A} + \mathbf{BG})\mathbf{x} \tag{7.81}$$

denoted by the symbol `E`.

7.2.2 Extensions to Trajectory Tracking and Integral Action

The LQR can be redesigned to track a time-varying reference trajectory $\mathbf{x}_d \in \mathbb{R}^n$ for a large class of mechanical systems possessing certain structural properties. This section presents a simple solution to this problem while a more general solution is presented in Section 7.2.3.

Transformation of the LQ Tracker to a Set-Point Regulation Problem

In order to transform a trajectory tracking problem, *reference feedforward* can be used. Unmeasured slowly-varying or constant disturbances are compensated by including integral action. This is usually done by augmenting an integral state $\dot{\mathbf{z}} = \mathbf{e}$ to the system model. A mass-damper-spring system will be used to demonstrate the design methodology.

Example 7.4 (Mass-Damper-Spring Trajectory Tracking Problem)
Consider the mass-damper-spring system:

$$\begin{aligned} \dot{x} &= v \\ m\dot{v} + dv + kx &= \tau \end{aligned}$$

Define:

$$\tau = \tau_{\mathrm{FF}} + \tau_{\mathrm{LQ}} \tag{7.82}$$

where the feedforward terms is chosen as:

$$\tau_{\mathrm{FF}} = m\dot{v}_d + dv_d + kx_d \tag{7.83}$$

such that:

$$m\ddot{e} + d\dot{e} + ke = \tau_{\mathrm{LQ}} \tag{7.84}$$

where $e = x - x_d$ and $\dot{e} = v - v_d$. The desired states are computed using a reference model:

$$\dot{x}_d = v_d \tag{7.85}$$

$$\dot{v}_d = \phi(v_d, r) \tag{7.86}$$

where r is the input. The trajectory tracking control problem has now been transformed to a LQ set-point regulation problem given by (7.83) which can be written in state-space form as:

$$\begin{aligned} \dot{\mathbf{x}} &= \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{d}{m} & -\frac{k}{m} \end{bmatrix}}_{\mathbf{A}} \mathbf{x} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}}_{\mathbf{B}} u \\ e &= \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{\mathbf{C}} \mathbf{x} \end{aligned}$$

with $\mathbf{x} = [e, \dot{e}]^\top$ and $u = \tau_{\mathrm{LQ}}$.

Integral Action

In example 7.4 it was shown that a feedforward term τ_{FF} could transfer the LQ trajectory tracking problem to a LQR problem. For the system model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{7.87}$$

integral action is obtained by augmenting the integral state $\mathbf{z} \in \mathbb{R}^m$ to the state vector. Let:

$$\dot{\mathbf{z}} = \mathbf{y} = \mathbf{C}\mathbf{x} \tag{7.88}$$

where the C-matrix is used to extract potential integral states from the x-vector; see Example 7.4. This system is a standard LQR problem:

$$\dot{\mathbf{x}}_a = \mathbf{A}_a\mathbf{x}_a + \mathbf{B}_a\mathbf{u} \tag{7.89}$$

where $\mathbf{x}_a = [\mathbf{z}^\top, \mathbf{x}^\top]^\top$ and:

$$\mathbf{A}_a = \begin{bmatrix} \mathbf{0} & \mathbf{C} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}, \quad \mathbf{B}_a = \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix} \tag{7.90}$$

The control objective is regulation of $\mathbf{x}_a$ to zero using $\mathbf{u}$. This is obtained by defining the performance index as:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^t \left(\mathbf{x}_a^\top \mathbf{Q}_a \mathbf{x}_a + \mathbf{u}^\top \mathbf{R} \mathbf{u} \right) d\tau \right\} \tag{7.91}$$

where $\mathbf{R} = \mathbf{R}^\top > 0$ and $\mathbf{Q}_a = \mathbf{Q}_a^\top \geq 0$ are the weighting matrices. Hence, the solution of the LQR set-point regulation problem is (see Section 7.2.1):

$$\begin{aligned} \mathbf{u} &= -\mathbf{R}^{-1}\mathbf{B}_a^\top \mathbf{P}_\infty \mathbf{x}_a \\ &= -\mathbf{R}^{-1}[\mathbf{0} \;\; \mathbf{B}^\top] \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{x} \end{bmatrix} \\ &= -\underbrace{\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_{12}}_{\mathbf{K}_i} \mathbf{z} - \underbrace{\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_{22}}_{\mathbf{K}_p} \mathbf{x} \end{aligned} \tag{7.92}$$

where $\mathbf{P}_{12}$ and $\mathbf{P}_{22}$ are found by solving the ARE (type `help lqr` in Matlab™):

$$\mathbf{P}_\infty \mathbf{A}_a + \mathbf{A}_a^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B}_a \mathbf{R}^{-1} \mathbf{B}_a^\top \mathbf{P}_\infty + \mathbf{Q}_a = \mathbf{0} \tag{7.93}$$

Notice that the feedback term, $\mathbf{u}$, includes feedback from the tracking errors $\mathbf{e}$ and $\dot{\mathbf{e}}$, and the integral state:

$$\mathbf{z} = \int_0^t \mathbf{e}(\tau)\, d\tau \tag{7.94}$$

7.2.3 General Solution of the LQ Trajectory Tracking Problem

Consider the state-space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{7.95}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{7.96}$$

The LQ trajectory tracking control problem is addressed under the assumption that both the state vector $\mathbf{x}$ and disturbance vector $\mathbf{w}$ are measured or at least obtained by state estimation. If the estimated values are used for $\mathbf{x}$ and $\mathbf{w}$, stability can be proven by applying a *separation principle*. This is known as LQG control in the literature and it involves the design of a *Kalman filter* for reconstruction of the unmeasured states which again requires that the system is *observable*. For simplicity, full state feedback is assumed in this chapter. The interested reader is recommended to consult the extensive literature on LQG control for output feedback control; see Athans and Falb (1966), Brian *et al.* (1989), for instance.

Reference Feedforward Assumptions

Consider a time-varying reference system:

$$\dot{\mathbf{x}}_d = \phi(\mathbf{x}_d, \mathbf{r}) \tag{7.97}$$

$$\mathbf{y}_d = \mathbf{C}\mathbf{x}_d \tag{7.98}$$

where $\mathbf{x}_d \in \mathbb{R}^n$ is the desired state, $\mathbf{y}_d \in \mathbb{R}^p$ $(p \leq n)$ is the desired output, $\mathbf{r} \in \mathbb{R}^r$ $(r \leq n)$ is a bounded commanded input, and $\phi : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^p$.

If linear theory is assumed the dynamics of the desired state can be conveniently represented by:

$$\phi(\mathbf{x}_d, \mathbf{r}) = \mathbf{A}_d\mathbf{x}_d + \mathbf{B}_d\mathbf{r} \tag{7.99}$$

This is a linear reference model for feedforward (tracking) control; see Section 5.1 for how to choose $\mathbf{A}_d$ and $\mathbf{B}_d$. A special case is regulation:

$$\mathbf{y}_d = \mathbf{C}\mathbf{x}_d = \text{constant} \tag{7.100}$$

Disturbance Feedforward Assumptions

Two cases of disturbance feedforward are considered:

1. The disturbance vector $\mathbf{w}$ = constant for all $t > T_p$ where T_p is the present time. An example of this is a ship exposed to constant (or at least slowly-varying) wind forces. This is a resonable assumption since the *average* wind speed and direction are not likely to change in minutes.

2. The disturbance $\mathbf{w} = \mathbf{w}(t)$ varies as a function of time t for future time $t > T_p$. This is the case for most physical disturbances. However, a feedforward solution requires that $\mathbf{w}$ is known (or at least estimated) for $t \geq 0$. In many cases this is unrealistic so the best we can do is to assume that $\mathbf{w}(t) = \mathbf{w}(T_p)$ = constant e.g. in a finite future time horizon so that it conforms to Case 1 above.

Control Objective

The control objective is to design a linear quadratic optimal tracking controller using a time-varying smooth reference trajectory $\mathbf{y}_d$ given by the system (7.97)–(7.98). Assume that the desired output $\mathbf{y}_d = \mathbf{C}\mathbf{x}_d$ is known for all time $t \in [0, T]$ where T is the final time. Define the error signal:

$$\begin{aligned} \mathbf{e} &= \mathbf{y} - \mathbf{y}_d \\ &= \mathbf{C}[\mathbf{x} - \mathbf{x}_d] \end{aligned} \tag{7.101}$$

The goal is to design an optimal tracking controller that tracks the desired output, i.e. regulates the error e to zero while minimizing:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2}\mathbf{e}^\top(T)\mathbf{Q}_f\mathbf{e}(T) + \frac{1}{2}\int_{t_0}^{T} (\mathbf{e}^\top\mathbf{Q}\mathbf{e} + \mathbf{u}^\top\mathbf{R}\mathbf{u})dt \right\}$$
$$\text{subject to} \quad \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w}, \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{7.102}$$

where $\mathbf{R} = \mathbf{R}^\top > 0$ and $\mathbf{Q} = \mathbf{Q}^\top \geq 0$ are the tracking error and control weighting matrices, respectively. The weight matrix $\mathbf{Q}_f = \mathbf{Q}_f^\top \geq 0$ can be included to add penalty on the final state. Notice that this is a *finite time-horizon* optimal control problem and it has to be solved by using the *Differential Riccati Equation* (DRE); see Athans and Falb (1966) pp. 793-801.

It is assumed that the desired output signal comes from a linear reference generator given by:

$$\dot{\mathbf{x}}_d = \mathbf{A}_d\mathbf{x}_d + \mathbf{B}_d\mathbf{r} \tag{7.103}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x}_d \tag{7.104}$$

where $\mathbf{r}$ is a given reference input which is filtered through the generator. $\mathbf{C}$ is the same output matrix as in the plant. A special case of (7.102) is the one with no weight on the final state, that is $\mathbf{Q}_f = 0$, resulting in the quadratic performance index:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T (\mathbf{e}^\top \mathbf{Q}\mathbf{e} + \mathbf{u}^\top \mathbf{R}\mathbf{u})\, dt \right\} \tag{7.105}$$

Substituting (7.69) into (7.105) yields the equivalent formulation:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T (\tilde{\mathbf{x}}^\top \tilde{\mathbf{Q}}\tilde{\mathbf{x}} + \mathbf{u}^\top \mathbf{R}\mathbf{u})\, dt \right\} \tag{7.106}$$

where $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d$ and:

$$\tilde{\mathbf{Q}} = \mathbf{C}^\top \mathbf{Q}\mathbf{C} \geq 0 \tag{7.107}$$

Linear Time-Varying Systems

It can be shown that the optimal control law is (Brian *et al.* 1989):

$$\mathbf{u} = -\mathbf{R}^{-1}\mathbf{B}^\top [\mathbf{P}\mathbf{x} + \mathbf{h}_1 + \mathbf{h}_2] \tag{7.108}$$

where $\mathbf{P}$, $\mathbf{h}_1$ and $\mathbf{h}_2$ originates from the system *Hamiltonian*. $\mathbf{P}$ accounts for the feedback part, $\mathbf{h}_1$ accounts for the feedforward part due to the time-varying nature of the reference signal $\mathbf{y}_d$, and $\mathbf{h}_2$ accounts for the feedforward part due to the measurable time-varying disturbance $\mathbf{w}$. The equations that need to be solved are:

$$\dot{\mathbf{P}} = -\mathbf{P}\mathbf{A} - \mathbf{A}^\top \mathbf{P} + \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P} - \tilde{\mathbf{Q}} \tag{7.109}$$

$$\dot{\mathbf{h}}_1 = -[\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}]^\top \mathbf{h}_1 + \tilde{\mathbf{Q}}\mathbf{x}_d \tag{7.110}$$

$$\dot{\mathbf{h}}_2 = -[\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}]^\top \mathbf{h}_2 - \mathbf{P}\mathbf{E}\mathbf{w} \tag{7.111}$$

with:

$$\mathbf{P}(T) = \tilde{\mathbf{Q}}_f \tag{7.112}$$

$$\mathbf{h}_1(T) = -\tilde{\mathbf{Q}}_f \mathbf{x}_d(T) \tag{7.113}$$

$$\mathbf{h}_2(T) = \mathbf{0} \tag{7.114}$$

where $\tilde{\mathbf{Q}}_f = \mathbf{C}^\top \mathbf{Q}_f \mathbf{C}$. Equations (7.109)–(7.111) represent three differential equations; a matrix DRE and two vector differential equations (adjoint operators), respectively. Notice that the initial conditions for these equations are not known, but rather the final conditions are known. Consequently, they have to be integrated *backward* in time a priori to find the initial conditions, and then be executed forward in time again with the closed-loop plant from $[0, T]$.

There are different ways of doing this. One attractive method is to discretize the system and run the resulting difference equation backward. A simple Euler integration routine for (7.109) is given below where δ is set as a small **negative** sampling time. Moreover, using a 1st-order Taylor expansion, yields:

$$\mathbf{P}(t+\delta) \approx \mathbf{P}(t) + \delta\{-\mathbf{PA} - \mathbf{A}^\top\mathbf{P} + \mathbf{PBR}^{-1}\mathbf{B}^\top\mathbf{P} - \tilde{\mathbf{Q}}\} \tag{7.115}$$

with $\mathbf{P}(T) = \tilde{\mathbf{Q}}_f$ produces $\mathbf{P}(0)$. Another procedure, is to simulate backwards in time. The system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) + \mathbf{G}(\mathbf{x}, t)\mathbf{u} \qquad t \in [T, 0] \tag{7.116}$$

can be simulated backwards in time by the following change of integration variable $t = T - \tau$ with $dt = -d\tau$, and:

$$-\frac{\mathbf{dx}(T-\tau)}{d\tau} = \mathbf{f}(\mathbf{x}(T-\tau), T-\tau) + \mathbf{G}(\mathbf{x}(T-\tau), T-\tau)\mathbf{u}(T-\tau) \tag{7.117}$$

Let $\mathbf{z}(\tau) = \mathbf{x}(T-\tau)$, then:

$$\frac{\mathbf{dz}(\tau)}{d\tau} = -\mathbf{f}(\mathbf{z}(\tau), T-\tau) - \mathbf{G}(\mathbf{z}(\tau), T-\tau)\mathbf{u}(T-\tau) \tag{7.118}$$

This system can now be simulated forward in time with initial condition $\mathbf{z}(0) = \mathbf{x}(T)$.

The method is demonstrated in Example 7.5 where it is assumed that both $\mathbf{x}_d$ and $\mathbf{w}$ are time-varying but known for all future t. Later a special case dealing with constant values for $\mathbf{x}_d$ and $\mathbf{w}$ will be studied.

Example 7.5 (Optimal Time-Varying LQ Trajectory Tracking Problem)
Consider a mass-damper-spring system:

$$m\ddot{x} + d\dot{x} + kx = u + w \tag{7.119}$$

where m is the mass, d is the damping coefficient, k is the spring stiffness coefficient, u is the input, and w is the disturbance. Choosing the states as $x_1 = x$ and $x_2 = \dot{x}$, the following state-space realization is obtained:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{d}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} w \tag{7.120}$$

For simplicity, assume that $m = k = 1$ and $d = 2$ such that:

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w \tag{7.121}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x} \tag{7.122}$$

where $\mathbf{x} = [x_1, x_2]^\top$. *The disturbance signal is assumed to be known for all future time and it is simply chosen as:*

$$w = \cos(t) \tag{7.123}$$

Similarly the reference signal is assumed to be known for all future time and it is given by the generator:

$$\dot{\mathbf{x}}_d = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix} \mathbf{x}_d + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \tag{7.124}$$

$$y_d = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_d \tag{7.125}$$

where

$$r = \sin(t) \tag{7.126}$$

The Matlab$^{\text{TR}}$ *GNC toolbox script* `ExLQFinHor.m` *demonstrates how forward and backward integration can be implemented for the mass-damper-spring system. The simulation results are shown in Figures 7.9–7.10.*

Approximate Solution for Linear Time-Invariant Systems

Unfortunately, the theory dealing with the limiting case:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \lim_{T \to \infty} \int_0^T \left(\mathbf{e}^\top \mathbf{Q} \mathbf{e} + \mathbf{u}^\top \mathbf{R} \mathbf{u} \right) dt \right\} \tag{7.127}$$

is not available. This solution is attractive since it represents a steady-state solution of the LQ trajectory tracking problem. Fortunately, this problem can be circumvented by assuming that T is large but still limited, that is:

$$0 \ll T_1 \leq T < \infty \tag{7.128}$$

where T_1 is a large constant. For $T \to \infty$ the solution of (7.109) will tend to the constant matrix $\mathbf{P}_\infty$ satisfying the *Algebraic Riccati Equation* (ARE):

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^\top \mathbf{P}_\infty + \tilde{\mathbf{Q}} = \mathbf{0} \tag{7.129}$$

This solution is interpreted as the steady-state solution of (7.109) where $\mathbf{P}(t) \approx \mathbf{P}_\infty$ for all $t \in [0, T_1]$. This is verified in the upper plot of Figure 7.10. Furthermore, it is assumed that:

$$\mathbf{x}_d = \text{constant}, \quad \mathbf{w} = \text{constant}, \quad \forall\, t \in [0, T_1]. \tag{7.130}$$

In practise the assumption that $\mathbf{x}_d$ is constant can be relaxed to $\mathbf{x}_d$ being slowly-varying compared to the state dynamics. A similar argument can be used for $\mathbf{w}$. It is also common to drop the disturbance feedforward term since integral action in the controller can compensate for non-zero slowly-varying disturbances.

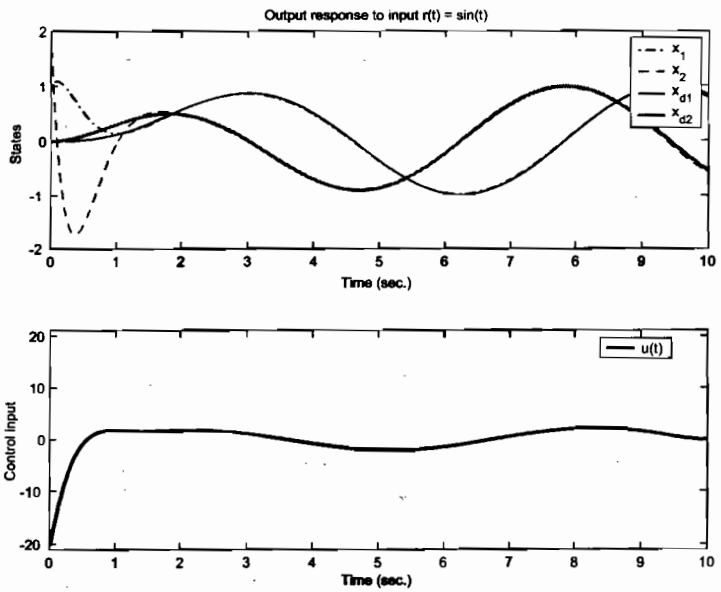


Figure 7.9: Upper plot: states x_1 and x_2 and the reference trajectories x_{d1} and x_{d2} as function of time. Lower plot: optimal control u as function of time.

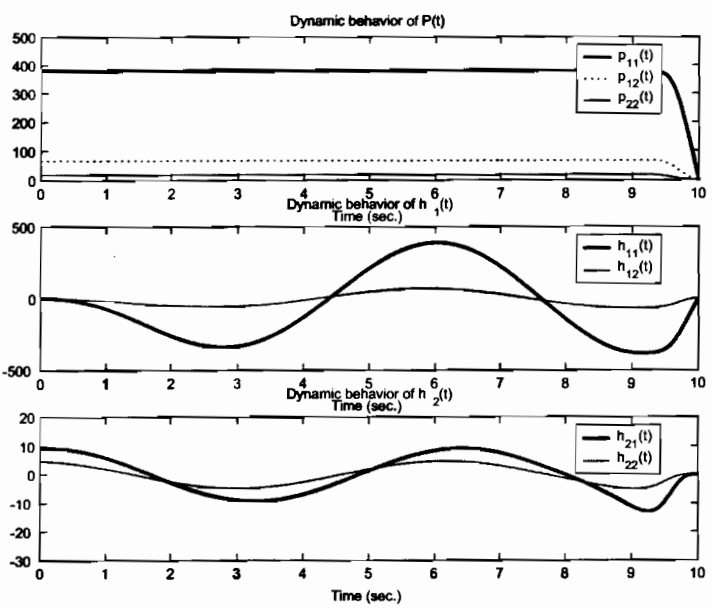


Figure 7.10: Optimal solutions of the elements in $\mathbf{P}$, $\mathbf{h}_1$ and $\mathbf{h}_2$ as function of time.

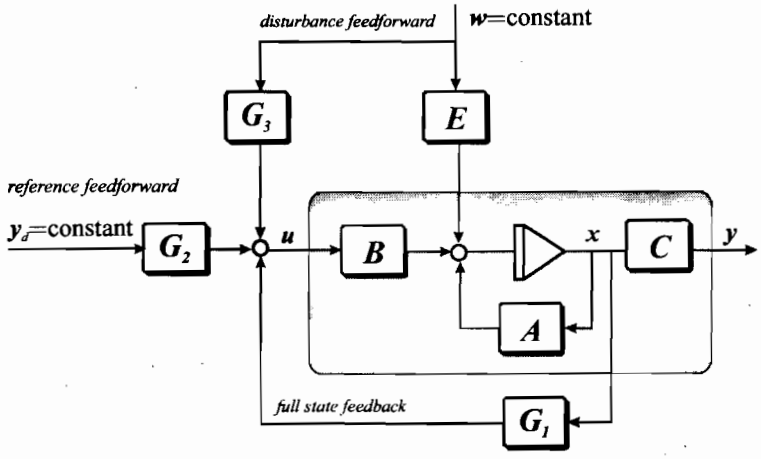


Figure 7.11: Block diagram showing the full state feedback LQ tracker solution with disturbance feedforward.

Next, if the eigenvalues of the matrix:

$$\mathbf{A}_c = \mathbf{A} + \mathbf{B}\mathbf{G}_1 \quad \text{where} \quad \mathbf{G}_1 = -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty \tag{7.131}$$

have negative real parts, that is:

$$\lambda_i(\mathbf{A}_c) < 0 \quad (i = 1 \ldots, n) \tag{7.132}$$

the steady-state solution for $\mathbf{h}_1$ and $\mathbf{h}_2$ in (7.110) and (7.111) on $[0, T_1]$ can be approximated as:

$$\mathbf{h}_{1\infty} = (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \tilde{\mathbf{Q}} \mathbf{x}_d \tag{7.133}$$

$$\mathbf{h}_{2\infty} = -(\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \mathbf{P}_\infty \mathbf{E}\mathbf{w} \tag{7.134}$$

Substitution of (7.129) into (7.108) yields the steady-state optimal control law:

$$\mathbf{u} = \mathbf{G}_1\, \mathbf{x} + \mathbf{G}_2\, \mathbf{y}_d + \mathbf{G}_3 \mathbf{w} \tag{7.135}$$

where $\mathbf{y}_d$=constant and $\mathbf{w}$ =constant, and:

$$\mathbf{G}_1 = -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty \tag{7.136}$$

$$\mathbf{G}_2 = -\mathbf{R}^{-1}\mathbf{B}^\top (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \mathbf{C}^\top \mathbf{Q} \tag{7.137}$$

$$\mathbf{G}_3 = \mathbf{R}^{-1}\mathbf{B}^\top (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \mathbf{P}_\infty \mathbf{E} \tag{7.138}$$

This solution is shown in Figure 7.11.

Matlab:
The function `lqtracker.m` is implemented in the GNC toolbox for computation of the matrices $\mathbf{G}_1, \mathbf{G}_2$ and $\mathbf{G}_3$:

```
function [G1,G2,G3] = lqtracker(A,B,C,Q,R)
[K,P,E] = lqr(A,B,C'*Q*C,R);
G1 = -inv(R)*B'*P;
Temp = inv((A+B*G1)');
G2 = -inv(R)*B'*Temp*C'*Q;
G3 = inv(R)*B'*Temp*P*E;
```

For a mass-damper-spring system the optimal trajectory tracking controller is found using `ExLQtrack.m`:

```
%Design matrices
Q = diag([1]);    % tracking error weights
R = diag([1]);    % input weights

% System matrices
A = [0 1; -1 -2];     % state matrix
B = [0; 1];           % input matrix
C = [1 0];            % output matrix

% Optimal gain matrices
[G1,G2,G3] = lqtracker(A,B,C,Q,R)
```

SISO Systems

Consider the SISO state-space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{E}\mathbf{w} \tag{7.139}$$

$$y = \mathbf{c}^{\mathsf{T}}\mathbf{x} \tag{7.140}$$

where $\mathbf{x} \in \mathbb{R}^n$, $u \in \mathbb{R}$ and $y \in \mathbb{R}$. For SISO systems, the performance index (7.105) simplifies to:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \lim_{T\to\infty} \int_0^T \left(q\, e^2 + r\, u^2\right)\, d\tau \right.$$

$$\left. = \frac{q}{2} \lim_{T\to\infty} \int_0^T \left(e^2 + \frac{r}{q}\, u^2\right)\, dt \right\} \tag{7.141}$$

where $q \geq 0$ and $r > 0$ are two scalars. By choosing $q = 1$ (without loss of generality) and defining $\lambda = r/q > 0$, the performance index (7.141) reduces to:

$$J^* = \min_{\mathbf{u}} \left\{ \frac{1}{2} \lim_{T\to\infty} \int_{t_o}^T \left(e^2 + \lambda u^2\right)\, dt \right\} \tag{7.142}$$

Consequently, the steady-state optimal solution can be approximated as:

$$u = \mathbf{g}_1^\top \mathbf{x} + g_2\, y_d + \mathbf{g}_3^\top \mathbf{w} \tag{7.143}$$

where

$$\mathbf{g}_1^\top = -\frac{1}{\lambda}\mathbf{b}^\top \mathbf{P}_\infty \tag{7.144}$$

$$g_2 = -\frac{1}{\lambda}\mathbf{b}^\top (\mathbf{A} + \mathbf{b}\mathbf{g}_1^\top)^{-\top}\mathbf{c} \tag{7.145}$$

$$\mathbf{g}_3^\top = \frac{1}{\lambda}\mathbf{b}^\top (\mathbf{A} + \mathbf{b}\mathbf{g}_1^\top)^{-\top}\mathbf{P}_\infty \mathbf{E} \tag{7.146}$$

Here $\mathbf{P}_\infty = \mathbf{P}_\infty^\top > 0$ is the solution of the *algebraic Riccati equation* (ARE):

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \frac{1}{\lambda}\mathbf{P}_\infty \mathbf{b}\mathbf{b}^\top \mathbf{P}_\infty + \mathbf{c}\mathbf{c}^\top = \mathbf{0} \tag{7.147}$$

For a mass-damper-spring system the term $\mathbf{g}_1^\top \mathbf{x}$ can be viewed as a PD-controller while $g_2 y_d$ and $\mathbf{g}_3^\top \mathbf{w}$ represent reference and disturbance feedforward, respectively.

7.3 State Feedback Linearization

The basic idea with feedback linearization is to transform the nonlinear systems dynamics into a linear system (Freund 1973). Conventional control techniques like pole placement and linear quadratic optimal control theory can then be applied to the linear system. In robotics, this technique is commonly referred to as *computed torque* control (Sciavicco and Siciliano 1996). Feedback linearization are discussed in more detail by Isidori (1989) and Slotine and Li (1991).

Feedback linearization is easily applicable to ships and underwater vehicles since these models basically are nonlinear *mass-damper-spring* systems which can be transformed into a linear system by using a nonlinear mapping (Fossen 1994). Transformations that can be used both for *b*- and *n*-frame applications will be presented. With *b*-frame applications we mean tracking control of the body-fixed velocities while *n*-frame applications will be position and attitude control. Combined position and velocity schemes are also discussed.

7.3.1 Decoupling in the b-Frame (Velocity)

The control objective is to transform the vessel dynamics into a linear system

$$\dot{\boldsymbol{\nu}} = \mathbf{a}^b \tag{7.148}$$

where $\mathbf{a}^b$ can be interpreted as a body-fixed *commanded acceleration* vector. The body-fixed vector representation should be used to control the vessels' linear and angular velocities.

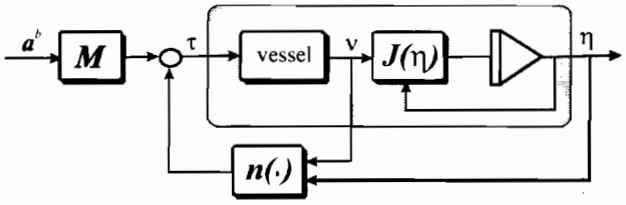


Figure 7.12: Nonlinear decoupling in the b-frame.

Consider the nonlinear vessel dynamics in the form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) = \boldsymbol{\tau} \tag{7.149}$$

where $\boldsymbol{\eta}$ and $\boldsymbol{\nu}$ are assumed to be measured and $\mathbf{n}$ is the nonlinear vector:

$$\mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) = \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) \tag{7.150}$$

The nonlinearities can be canceled out by simply selecting the control law as (see Figure 7.12):

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{a}^b + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) \tag{7.151}$$

where the commanded acceleration vector $\mathbf{a}^b$ can be chosen by e.g. pole placement or linear quadratic optimal control theory. However, note that to investigate optimality of the original system, the optimal control and cost function must be transformed back through the nonlinear mapping.

Pole Placement

Let $\boldsymbol{\Lambda} > 0$ be a diagonal design matrix

$$\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$$

used to specify the desired control bandwidth, $\boldsymbol{\nu}_d$ the desired linear and angular velocity vector and $\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu} - \boldsymbol{\nu}_d$ the velocity tracking error. Then the commanded acceleration vector can be chosen as a PI-controller with acceleration feedforward:

$$\mathbf{a}^b = \dot{\boldsymbol{\nu}}_d - \mathbf{K}_p\tilde{\boldsymbol{\nu}} - \mathbf{K}_i \int_0^t \tilde{\boldsymbol{\nu}}(\tau)d\tau \tag{7.152}$$

Choosing the gains as:

$$\mathbf{K}_p = 2\boldsymbol{\Lambda}. \qquad \mathbf{K}_i = \boldsymbol{\Lambda}^2$$

yields the 2nd-order error dynamics:

$$\mathbf{M}(\dot{\boldsymbol{\nu}} - \mathbf{a}^b) = \mathbf{M}(\dot{\tilde{\boldsymbol{\nu}}} + 2\boldsymbol{\Lambda}\tilde{\boldsymbol{\nu}} + \boldsymbol{\Lambda}^2 \int_0^t \tilde{\boldsymbol{\nu}}(\tau)d\tau) = \mathbf{0} \tag{7.153}$$

This implies that for each DOF both poles are in $s = -\lambda_i$ $(i = 1, \ldots, n)$. Consequently:

$$(s + \lambda_i)^2 \int_0^t \tilde{\boldsymbol{\nu}}(\tau)d\tau = 0, \quad (i = 1, \ldots, n) \tag{7.154}$$

The reference model of Section 5.1 can be used to generate a smooth velocity trajectory $\boldsymbol{\nu}_d$ for tracking control.

Example 7.6 (Speed Control)
Consider a simplified model of a ship in surge, that is:

$$m\dot{u} + d_1 u + d_2|u|u = \tau \tag{7.155}$$

The commanded acceleration is calculated as:

$$a^b = \dot{u}_d - K_p(u - u_d) - K_i \int_0^t (v - v_d)d\tau \tag{7.156}$$

This suggests that the speed controller should be computed as:

$$\tau = m[\dot{u}_d - K_p(u - u_d) - K_i \int_0^t (v - v_d)d\tau] + d_1 u + d_2|u|u \tag{7.157}$$

with reference model (see Section 5.1)

$$\ddot{u}_d + 2\zeta\omega\dot{u}_d + \omega^2 u_d = \omega^2 r^b \tag{7.158}$$

where $\zeta > 0$ and $\omega > 0$ are the reference model damping ratio and natural frequency while r^b is the commanded input (desired surge speed).

7.3.2 Decoupling in the n-Frame (Position and Attitude)

For position and attitude control the vessel's dynamics and kinematics are decoupled in the NED reference frame. Consider:

$$\ddot{\boldsymbol{\eta}} = \mathbf{a}^n \tag{7.159}$$

where $\mathbf{a}^n$ can be interpreted as the n-frame commanded acceleration. Consider the dynamic and kinematic equations in the form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) = \boldsymbol{\tau} \tag{7.160}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\,\boldsymbol{\nu} \tag{7.161}$$

where both $\boldsymbol{\eta}$ and $\boldsymbol{\nu}$ are assumed measured. Differentiation of the kinematic equation with respect to time yields:

$$\dot{\boldsymbol{\nu}} = \mathbf{J}^{-1}(\boldsymbol{\eta})[\ddot{\boldsymbol{\eta}} - \dot{\mathbf{J}}(\boldsymbol{\eta})\boldsymbol{\nu}] \tag{7.162}$$

The nonlinear control law:

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{a}^b + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) \tag{7.163}$$

applied to the vessel dynamics, yields:

$$\mathbf{M}(\dot{\boldsymbol{\nu}} - \mathbf{a}^b) = \mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})[\ddot{\boldsymbol{\eta}} - \dot{\mathbf{J}}(\boldsymbol{\eta})\boldsymbol{\nu} - \mathbf{J}(\boldsymbol{\eta})\mathbf{a}^b] = \mathbf{0} \tag{7.164}$$

Choosing:

$$\mathbf{a}^n = \dot{\mathbf{J}}(\boldsymbol{\eta})\boldsymbol{\nu} + \mathbf{J}(\boldsymbol{\eta})\mathbf{a}^b \tag{7.165}$$

yields the linear decoupled system:

$$\mathbf{M}^*(\ddot{\boldsymbol{\eta}} - \mathbf{a}^n) = \mathbf{0} \tag{7.166}$$

where $\mathbf{M}^* = \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta}) > 0$. From (7.165) it is seen that:

$$\mathbf{a}^b = \mathbf{J}^{-1}(\boldsymbol{\eta})[\mathbf{a}^n - \dot{\mathbf{J}}(\boldsymbol{\eta})\boldsymbol{\nu}] \tag{7.167}$$

where the commanded acceleration $\mathbf{a}^n$ should be chosen as a PID-control law with acceleration feedforward:

$$\mathbf{a}^n = \ddot{\boldsymbol{\eta}}_d - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}} - \mathbf{K}_p\tilde{\boldsymbol{\eta}} - \mathbf{K}_i\int_0^t \tilde{\boldsymbol{\eta}}(\tau)d\tau \tag{7.168}$$

where $\mathbf{K}_p$, $\mathbf{K}_d$, and $\mathbf{K}_i$ are positive definite matrices chosen such that the error dynamics:

$$\ddot{\tilde{\boldsymbol{\eta}}} + \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}} + \mathbf{K}_p\tilde{\boldsymbol{\eta}} + \mathbf{K}_i\int_0^t \tilde{\boldsymbol{\eta}}(\tau)d\tau = \mathbf{0} \tag{7.169}$$

is stable. One simple pole placement algorithm for PID-control is:

$$(s + \lambda_i)^3\int_0^t \tilde{\boldsymbol{\eta}}(\tau)d\tau = \mathbf{0}, \quad (i = 1, \ldots, n) \tag{7.170}$$

which yields:

$$\begin{aligned} \mathbf{K}_p &= 3\boldsymbol{\Lambda} = 3\mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_n\} \\ \mathbf{K}_d &= 3\boldsymbol{\Lambda}^2 = 3\mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_n\} \\ \mathbf{K}_i &= \boldsymbol{\Lambda}^3 = \mathrm{diag}\{\lambda_1^3, \lambda_2^3, \ldots, \lambda_n^3\} \end{aligned}$$

This is shown in Figure 7.13. When implementing the tracking control a 3rd-order reference model can be used to compute smooth position and attitude trajectories $\boldsymbol{\eta}_d$; see Section 5.1.

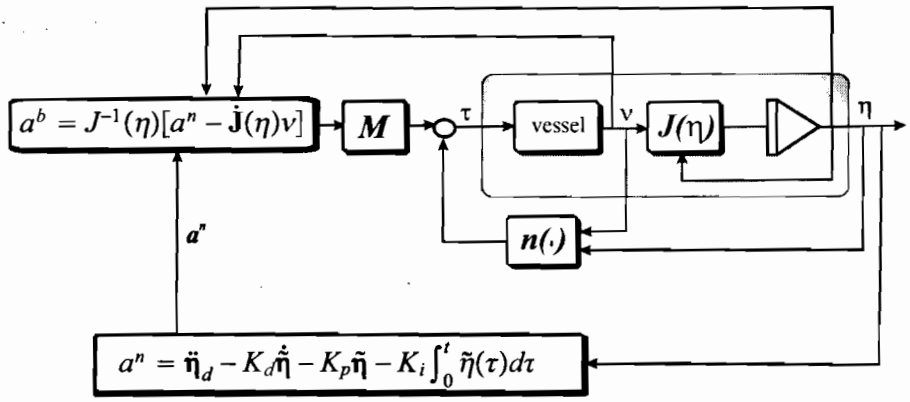


Figure 7.13: Nonlinear decoupling in the n-frame with transformation to the vessel b-frame.

Example 7.7 (Heading Control System)
Consider the ship model (Norrbin 1963):

$$\dot{\psi} = r \tag{7.171}$$

$$m\dot{r} + d_1 r + d_2|r|r = \tau \tag{7.172}$$

where ψ is the yaw angle. Hence, the commanded acceleration can be calculated as (Fossen and Paulsen 1992):

$$a^n = \dot{r}_d - K_d(r - r_d) - K_p(\psi - \psi_d) - K_i\int_0^t (\psi - \psi_d)d\tau \tag{7.173}$$

where r_d is the desired angular velocity and ψ_d is the desired heading angle. For this particular example $a^n = a^b$, which yields the decoupling control law:

$$\tau = m\left[\dot{r}_d - K_d(r - r_d) - K_p(\psi - \psi_d) - K_i\int_0^t (\psi - \psi_d)d\tau\right] + d_1 r + d_2|r|r \tag{7.174}$$

with reference model (see Section 5.1):

$$\psi_d^{(3)} + (2\zeta + 1)\omega\ddot{\psi}_d + (2\zeta + 1)\omega^2\dot{\eta}_d + \omega^3\eta_d = \omega^3 r^n \tag{7.175}$$

7.3.3 Adaptive Feedback Linearization

So far only feedback linearization under the assumption that all model parameters are *known* has been discussed. In this section, a *parameter adaptation law* to be used together with the previous control laws is derived. Taking the control law to be:

$$\tau = \hat{\mathbf{M}}\mathbf{a}^b + \hat{\mathbf{n}}(\boldsymbol{\nu}, \boldsymbol{\eta}) \tag{7.176}$$

where the *hat* denotes the adaptive parameter estimates, yields the error dynamics:

$$\mathbf{M}[\dot{\boldsymbol{\nu}} - \mathbf{a}^b] = [\hat{\mathbf{M}} - \mathbf{M}]\mathbf{a}^b + [\hat{\mathbf{n}}(\boldsymbol{\nu}, \boldsymbol{\eta}) - \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta})] \tag{7.177}$$

If the equations of motion are linear in a parameter vector $\boldsymbol{\theta}$, the following parameterization can be applied:

$$[\hat{\mathbf{M}} - \mathbf{M}]\mathbf{a}^b + [\hat{\mathbf{n}}(\boldsymbol{\nu}, \boldsymbol{\eta}) - \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta})] = \boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\tilde{\boldsymbol{\theta}} \tag{7.178}$$

Here $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$ is the *unknown* parameter error vector and $\boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})$ is a *known* matrix function of measured signals usually referred to as the *regressor matrix*. Using the result $\mathbf{a}^n = \dot{\mathbf{J}}(\boldsymbol{\eta})\boldsymbol{\nu} + \mathbf{J}(\boldsymbol{\eta})\mathbf{a}^b$ from (7.165), gives:

$$\mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})[\ddot{\boldsymbol{\eta}} - \mathbf{a}^n] = \boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\tilde{\boldsymbol{\theta}} \tag{7.179}$$

Premultiplying this expression with $\mathbf{J}^{-\top}(\boldsymbol{\eta})$ and letting $\mathbf{M}^*(\boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})$ yields the *n*-frame error dynamics:

$$\mathbf{M}^*(\boldsymbol{\eta})[\ddot{\boldsymbol{\eta}} - \mathbf{a}^n] = \mathbf{J}^{-\top}(\boldsymbol{\eta})\boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\tilde{\boldsymbol{\theta}} \tag{7.180}$$

Furthermore, let the commanded acceleration be chosen as the PD-type control:

$$\mathbf{a}^n = \ddot{\boldsymbol{\eta}}_d - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}} - \mathbf{K}_p\tilde{\boldsymbol{\eta}} \tag{7.181}$$

where $\mathbf{K}_p > 0$ and $\mathbf{K}_d > 0$. Hence, the error dynamics can be expressed according to:

$$\mathbf{M}^*(\boldsymbol{\eta})[\ddot{\tilde{\boldsymbol{\eta}}} + \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}} + \mathbf{K}_p\tilde{\boldsymbol{\eta}}] = \mathbf{J}^{-\top}(\boldsymbol{\eta})\boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\tilde{\boldsymbol{\theta}} \tag{7.182}$$

Writing this expression in state-space form, yields:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{J}^{-\top}(\boldsymbol{\eta})\boldsymbol{\Phi}(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\tilde{\boldsymbol{\theta}} \tag{7.183}$$

where $\mathbf{x} = [\tilde{\boldsymbol{\eta}}^\top, \dot{\tilde{\boldsymbol{\eta}}}^\top]^\top$ and:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ (\mathbf{M}^*(\boldsymbol{\eta}))^{-1} \end{bmatrix} \tag{7.184}$$

Convergence of $\tilde{\boldsymbol{\eta}}$ to zero can be proven by defining:

$$V(\mathbf{x}, \tilde{\boldsymbol{\theta}}, t) = \mathbf{x}^\top\mathbf{P}(t)\mathbf{x} + \tilde{\boldsymbol{\theta}}^\top\boldsymbol{\Gamma}^{-1}\tilde{\boldsymbol{\theta}} \tag{7.185}$$

where $\mathbf{P} = \mathbf{P}^\top > 0$ and $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^\top > 0$. Differentiating V with respect to time and substituting the error dynamics into the expression for $\dot{V}$, yields:

$$\dot{V} = \mathbf{x}^\top(\dot{\mathbf{P}} + \mathbf{P}\mathbf{A} + \mathbf{A}^\top\mathbf{P})\mathbf{x} + 2(\mathbf{x}^\top\mathbf{P}\mathbf{B}\mathbf{J}^{-\top}\boldsymbol{\Phi} + \dot{\tilde{\boldsymbol{\theta}}}^\top\boldsymbol{\Gamma}^{-1})\tilde{\boldsymbol{\theta}} \tag{7.186}$$

where $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^\top > 0$ is a positive definite weighting matrix of appropriate dimension. This suggests the parameter update law (assuming constant parameters–i.e., $\dot{\boldsymbol{\theta}} = \mathbf{0}$):

$$\dot{\hat{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma}\boldsymbol{\Phi}^\top(\mathbf{a}^b, \boldsymbol{\nu}, \boldsymbol{\eta})\mathbf{J}^{-1}(\boldsymbol{\eta})\mathbf{y} \tag{7.187}$$

where a new signal vector $\mathbf{y}$ defined as:

$$\mathbf{y} = \mathbf{C}\mathbf{x}, \qquad \mathbf{C} = \mathbf{B}^\top \mathbf{P} \tag{7.188}$$

has been introduced. In order to prove that $\dot{V} \leq 0$, let:

$$\mathbf{C} = [c_0\mathbf{I} \;\; c_1\mathbf{I}] \tag{7.189}$$

where $c_0 > 0$ and $c_1 > 0$ are two scalars to be interpreted later. Furthermore, let:

$$\mathbf{P}\mathbf{A} + \mathbf{A}^\top\mathbf{P} = -\mathbf{Q}; \qquad \mathbf{Q} = \mathbf{Q}^\top > 0 \tag{7.190}$$

where $\mathbf{P}$ and $\mathbf{Q}$ are defined according to (Asare and Wilson 1986):

$$\mathbf{P} = \begin{bmatrix} c_0\mathbf{M}^*\mathbf{K}_d + c_1\mathbf{M}^*\mathbf{K}_p & c_0\mathbf{M}^* \\ c_0\mathbf{M}^* & c_1\mathbf{M}^* \end{bmatrix} \tag{7.191}$$

$$\mathbf{Q} = \begin{bmatrix} 2c_0\mathbf{M}^*\mathbf{K}_p & \mathbf{0} \\ \mathbf{0} & 2(c_1\mathbf{M}^*\mathbf{K}_d - c_0\mathbf{M}^*) \end{bmatrix} \tag{7.192}$$

Assume that there exists a constant $\beta > 0$ such that:

$$\mathbf{x}^\top\dot{\mathbf{P}}\mathbf{x} = \mathbf{x}^\top \begin{bmatrix} c_0\dot{\mathbf{M}}^*\mathbf{K}_d + c_1\dot{\mathbf{M}}^*\mathbf{K}_p & c_0\dot{\mathbf{M}}^* \\ c_0\dot{\mathbf{M}}^* & c_1\dot{\mathbf{M}}^* \end{bmatrix} \mathbf{x} \leq \beta\mathbf{x}^\top \begin{bmatrix} \mathbf{M}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^* \end{bmatrix} \mathbf{x} \tag{7.193}$$

Hence, $\mathbf{P} = \mathbf{P}^\top > 0$, $c_0 > 0$, $c_1 > 0$ and $\mathbf{x}^\top\mathbf{Q}\mathbf{x} > \mathbf{x}^\top\dot{\mathbf{P}}\mathbf{x}$ implies that:

$$\dot{V} = \mathbf{x}^\top(\dot{\mathbf{P}} - \mathbf{Q})\mathbf{x} \leq 0 \tag{7.194}$$

if the following requirements are satisfied:

$$\begin{array}{ll} \text{(i)} & (c_0\mathbf{K}_d + c_1\mathbf{K}_p)c_1 > c_0^2\mathbf{I} \\ \text{(ii)} & 2c_0\mathbf{K}_p > \beta\mathbf{I} \\ \text{(iii)} & 2(c_1\mathbf{K}_d - c_0\mathbf{I}) > \beta\mathbf{I} \end{array}$$

Here β usually is taken to be a small positive constant while $\mathbf{K}_p > 0$ and $\mathbf{K}_d > 0$ can be chosen as diagonal matrices. Consequently, convergence of $\tilde{\eta}$ and $\dot{\tilde{\eta}}$ to zero is guaranteed by applying *Barbălat's lemma* (Barbălat 1959); see Appendix A.2. It is also seen that the parameter vector $\tilde{\theta}$ will be bounded but not necessarily convergent.

Adaptive feedback linearization has been applied to the ship autopilot control problem of Example 7.7 by Fossen and Paulsen (1992).

7.4 Integrator Backstepping

Backstepping is a design methodology for construction of a feedback control law through a *recursive* construction of a control Lyapunov function (CLF). Nonlinear backstepping designs are strongly related to feedback linearization. While feedback linearization methods, however, cancel all nonlinearities in the system it will be shown that when applying the backstepping design methodology more design flexibility is obtained. In particular the designer is given the possibility to exploit "good" nonlinearities while "bad" nonlinearities can be dominated e.g. by adding nonlinear damping. Hence, additional robustness is obtained, which is important in industrial control systems since cancellation of all nonlinearities require precise models that are difficult to obtain in practise.

7.4.1 A Brief History of Backstepping

The idea of integrator backstepping seems to have appeared simultaneously, often implicit, in the works of Koditschek (1987), Sonntag and Sussmann. (1988), Tsinias (1989), and Byrnes and Isidori (1989). Stabilization through an integrator (Kokotovic and Sussmann 1989) can be viewed as a special case of stabilization through an SPR transfer function which is a frequently used technique in the early adaptive designs (see Parks 1966, Landau 1979, Narendra and Annaswamy 1989). Extensions to nonlinear cascades by using passivity arguments have been done by Ortega (1991) and Byrnes *et al.* (1991). Integrator backstepping appeared as a recursive design technique in Saberi *et al.* (1990) and it was further developed by Kanellakopoulos *et al.* (1992). The relationship between backstepping and passivity has been established by Lozano *et al.* (1992). For the interested reader, a tutorial overview of backstepping is given by Kokotovic (1991).

Adaptive and nonlinear backstepping designs are described in detail by Krstic *et al.* (1995). This includes methods for parameter adaptation, tuning functions, and modular designs for both full state feedback and output feedback (observer backstepping).

In Sepulchre *et al.* (1997) extensions to forwarding, passivity, and cascaded designs are made. Also discussions on stability margins and optimality are included.

In Krstic and Deng (1998) stochastic systems with focus on stochastic stability and regulation, stochastic adaptive backstepping designs and disturbance attenuation are presented.

The focus of this section is practical designs with implementation considerations for mechanical systems. This is done by exploiting the nonlinear system properties of mechanical systems like dissipativness (good damping), symmetry of the inertia matrix, and the skew-symmetric property of the Coriolis and centripetal matrix. In addition, emphasis is placed on control design with integral action. Two techniques for integral action in nonlinear systems using backstepping designs are discussed (see Loria *et al.* 1999, Fossen *et al.* 2001).

Finally, this section is written in a vectorial setting in order to exploit the structural properties of nonlinear MIMO systems. This technique is referred to as *vectorial backstepping* (see Fossen and Berge 1997, Fossen and Grøvlen 1998).

7.4.2 The Main Idea of Integrator Backstepping

Integrator backstepping is a recursive design technique using *control Lyapunov functions (CLF)*. The CLF concept is a generalization of Lyapunov design results by e.g. Jacobson (1977) and Jurdjevic and Quinn (1978).

Definition 7.3 (Control Lyapunov Function)
A smooth positive definite and radially unbounded function $V : \mathbb{R}^n \to \mathbb{R}_+$ is called a control Lyapunov function for (see Arstein 1983, Sontag 1983):

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) \tag{7.195}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^r$ if:

$$\inf_{\mathbf{u} \in \mathbb{R}^r} \left\{ \frac{\partial V}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{f}(\mathbf{x}, \mathbf{u}) \right\} < 0, \quad \forall \mathbf{x} \neq 0 \tag{7.196}$$

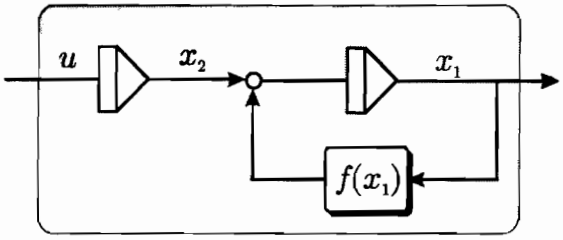


Figure 7.14: 2nd-order nonlinear system with one single nonlinearity $f(x_1)$ and a pure integraor at the input.

The main idea of integrator backstepping can be demonstrated by considering a simple nonlinear scalar system:

$$\dot{x}_1 = f(x_1) + x_2 \tag{7.197}$$
$$\dot{x}_2 = u \tag{7.198}$$
$$y = x_1 \tag{7.199}$$

where $x_1 \in \mathbb{R}, x_2 \in \mathbb{R}, y \in \mathbb{R}$ and $u \in \mathbb{R}$. The second equation represent a pure integrator; see Figure 7.14.

Let the design objective be regulation of $y(t) \to 0$ as $t \to \infty$. The only equilibrium point with $y = 0$ is $(x_1, x_2) = (0, -f(0))$ corresponding to $\dot{x}_1 = f(0) + x_2 = 0$. The design objective is to render the equilibrium point GAS or GES. Since the nonlinear system (7.197)–(7.198) consists of two states x_1 and x_2, this will be a recursive design in 2 steps. Equations (7.197)–(7.198) are therefore treated as two cascaded systems, each with a single input and output. The recursive design starts with the system x_1 and continues with x_2. A change of coordinates:

$$\mathbf{z} = \phi(\mathbf{x}) \tag{7.200}$$

is introduced during the recursive design process where $\mathbf{x} = [x_1, x_2]^\top, \mathbf{z} = [z_1, z_2]^\top$ is a new state vector, and $\phi(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^n$ is a transformation to be interpreted later. The backstepping transformation is a *global diffeomorphism*–i.e., a mapping with smooth functions $\phi(\mathbf{x})$ and $\phi^{-1}(\mathbf{x})$. Hence, the existence of an inverse transformation:

$$\mathbf{x} = \phi^{-1}(\mathbf{z}) \tag{7.201}$$

is guaranteed.

Step 1

For the first system (7.197) the state x_2 is chosen as a *virtual control* input while it is recalled that our design objective is to regulate the output $y = x_1$ to zero. Hence, the first backstepping variable is chosen as:

$$z_1 = x_1 \tag{7.202}$$

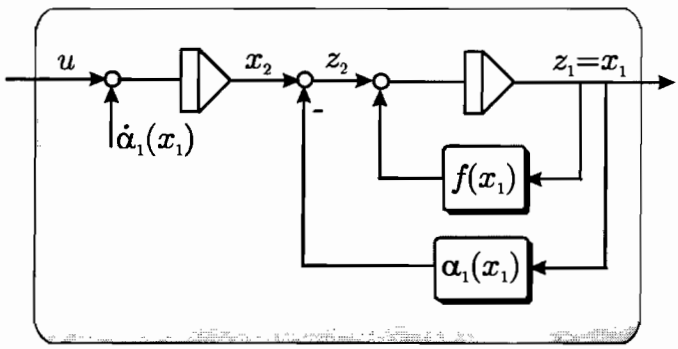


Figure 7.15: Stabilization of the x_1-system by means of the stabilizing function $\alpha_1 = \alpha_1(x_1)$. Note that $\dot{\alpha}_1(x_1)$ when integrated cancels out the feedback term $-\alpha_1(x_1)$.

The virtual control is defined as:

$$x_2 := \alpha_1 + z_2 \tag{7.203}$$

where

$$\begin{array}{lcl} \alpha_1 & = & \text{stabilizing function} \\ z_2 & = & \text{new state variable} \end{array}$$

Hence, the z_1-system can be written:

$$\dot{z}_1 = f(z_1) + \alpha_1 + z_2 \tag{7.204}$$

The new state variable z_2 will not be used in the first step, but its presence is important since z_2 is needed to couple the z_1-system to the next system, that is the z_2-system to be considered in the next step. Moreover, integrator backstepping implies that the coordinates during the recursive design is changed from (x_1, x_2) to (z_1, z_2).

A CLF for the z_1-system is:

$$V_1 = \frac{1}{2} z_1^2 \tag{7.205}$$

$$\begin{aligned} \dot{V}_1 &= z_1 \dot{z}_1 \\ &= z_1(f(z_1) + \alpha_1) + z_1 z_2 \end{aligned} \tag{7.206}$$

We now turn our attention to the design of the stabilizing function α_1 which will provide the necessary feedback for the z_1-system. For instance, choosing the stabilizing function as (feedback linearizing control):

$$\alpha_1 = -f(z_1) - k_1 z_1 \tag{7.207}$$

where $k_1 > 0$ is the feedback gain, yields:

$$\dot{V}_1 = -k_1 z_1^2 + z_1 z_2 \tag{7.208}$$

and:

$$\dot{z}_1 = -k_1 z_1 + z_2 \tag{7.209}$$

A block diagram showing the stabilizing function and the new state variable is shown in Figure 7.15. Hence, if $z_2 = 0$ then the z_1-system is stabilized. We now turn our attention to the z_2-system.

Step 2

The z_2-dynamics is computed by time differentiation of (7.203):

$$\begin{aligned} \dot{z}_2 &= \dot{x}_2 - \dot{\alpha}_1 \\ &= u - \dot{\alpha}_1 \end{aligned} \tag{7.210}$$

A CLF for the z_2-system is:

$$V_2 = V_1 + \frac{1}{2} z_2^2 \tag{7.211}$$

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \dot{z}_2 z_2 \\ &= (-k_1 z_1^2 + z_1 z_2) + \dot{z}_2 z_2 \\ &= -k_1 z_1^2 + z_2(z_1 + \dot{z}_2) \\ &= -k_1 z_1^2 + z_2(u - \dot{\alpha}_1 + z_1) \end{aligned} \tag{7.212}$$

Since our system has relative degree 2, the control input u appears in the second step. Hence, choosing the control law as:

$$u = \dot{\alpha}_1 - z_1 - k_2 z_2 \tag{7.213}$$

with $k_2 > 0$, yields:

$$\dot{V}_2 = -k_1 z_1^2 - k_2 z_2^2 < 0, \ \forall z_1 \neq 0, z_2 \neq 0 \tag{7.214}$$

Implementation Aspects

When implementing the control law (7.213) it is important to avoid expressions involving the time derivatives of the states. For this simple system only $\dot{\alpha}_1$ must be evaluated. This can be done by time differentiation of $\alpha_1(x_1)$ along the trajectory of x_1. Hence, $\dot{\alpha}_1$ can be computed without using the state derivatives:

$$\begin{aligned} \dot{\alpha}_1 &= -\frac{\partial f(x_1)}{\partial x_1}\dot{x}_1 - k_1 \dot{x}_1 \\ &= -\left(\frac{\partial f(x_1)}{\partial x_1} + k_1\right)(f(x_1) + x_2) \end{aligned} \tag{7.215}$$

The final expression for the control law is then:

$$u = -\left(\frac{\partial f(x_1)}{\partial x_1} + k_1\right)(f(x_1) + x_2) - x_1 - k_2(x_2 + f(x_1) + k_1 x_1) \tag{7.216}$$

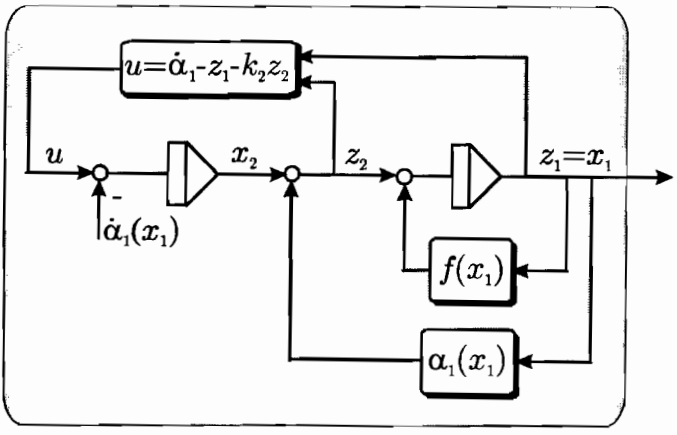


Figure 7.16: Stabilization of the x_2-system by means of the control input $u = u(\dot{\alpha}_1, z_1, z_2)$.

If $f(x_1) = -x_1$ (linear theory), it is seen that:

$$\begin{aligned} u &= -(-1+k_1)(-x_1+x_2) - x_1 - k_2(x_2 - x_1 + k_1 x_1) \\ &= -\underbrace{(2+k_1k_2-k_1-k_2)}_{K_p} x_1 - \underbrace{(k_1+k_2-1)}_{K_d} x_2 \end{aligned} \tag{7.217}$$

which is a standard PD-control law. In general, the expression for u is, however, a nonlinear feedback control law depending on the nonlinear function $f(x_1)$.

Backstepping Coordinate Transformation

The backstepping coordinate transformation $\mathbf{z} = \boldsymbol{\phi}(\mathbf{x})$ takes the form:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 + f(x_1) + k_1 x_1 \end{bmatrix} \tag{7.218}$$

while the inverse transformation $\mathbf{x} = \boldsymbol{\phi}^{-1}(\mathbf{z})$ is:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 - f(z_1) - k_1 z_1 \end{bmatrix} \tag{7.219}$$

The Final Check

If you have performed the backstepping design procedure correctly the dynamics of the closed-loop system in (z_1, z_2) coordinates can always be written as the sum of a diagonal and skew-symmetric matrix times the state vector. This can be seen by writing the resulting dynamics in the form:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \underbrace{-\begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}}_{\text{diagonal matrix}} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}}_{\text{skew-symmetrical matrix}} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{7.220}$$

or equivalently:

$$\dot{\mathbf{z}} = -\mathbf{K}\mathbf{z} + \mathbf{S}\mathbf{z} \tag{7.221}$$

where $\mathbf{z} = [z_1, z_2]^\top$, $\mathbf{K} = \text{diag}\{k_1, k_2\} > 0$ and:

$$\mathbf{S} = -\mathbf{S}^\top = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{7.222}$$

where $\mathbf{S}$ satisfies $\mathbf{z}^\top \mathbf{S}\mathbf{z} = 0, \forall \mathbf{z}$. In some cases the diagonal matrix will be a function of the state, that is $\mathbf{K}(\mathbf{z}) > 0$. This is the case when nonlinear damping is added or when some of the nonlinearities not are cancelled by the controller.

Investigation of Stability

It is also seen that:

$$V_2 = \frac{1}{2}\mathbf{z}^\top \mathbf{z} \tag{7.223}$$

$$\dot{V}_2 = \mathbf{z}^\top(-\mathbf{K}\mathbf{z} + \mathbf{S}\mathbf{z}) = -\mathbf{z}^\top \mathbf{K}\mathbf{z} \tag{7.224}$$

Hence, Lyapunov's direct method for autonomous systems ensures that the equilibrium point $(x_1, x_2) = (0, -f(0))$ is GAS. In fact, this system will also be GES since it can be shown that the state vector $\mathbf{x}$ decays exponentially to zero by using Theorem A.3, that is:

$$\|\mathbf{z}(t)\|_2 \leq e^{-\beta(t-t_0)} \|\mathbf{z}(t_0)\|_2 \tag{7.225}$$

where $\beta = \lambda_{\min}(\mathbf{K}) > 0$ is the convergence rate.

A generalization to single-input single-output (SISO) mass-damper-spring systems is done in Section 7.4.3 while extensions to MIMO control is made in Section 7.4.6.

Backstepping versus Feedback Linearization

The backstepping control law of the previous section is in fact equal to a feedback linearizing controller since the nonlinear function $f(x_1)$ is perfectly compensated for by choosing the stabilizing function as:

$$\alpha_1 = -f(x_1) - k_1 z_1 \tag{7.226}$$

The disadvantage with this approach is that a perfect model is required. This is impossible in practise. Consequently, an approach of cancelling all the nonlinearities may be sensitive for modeling errors.

One of the nice features of backstepping is that the stabilizing functions can be modified to exploit so-called "good" nonlinearities. For instance, assume that:

$$f(x_1) = -a_0 x_1 - a_1 x_1^2 - a_2 |x_1| x_1 \tag{7.227}$$

where a_0, a_1 and a_2 are assumed to be *unknown* positive constants. Since both $a_0 x_1$ and $a_2 |x_1| x_1$ tend to damp out the motion these two expressions should be exploited in the control design and therefore not cancelled out. On the contrary, the destabilizing term $a_1 x_1^2$ must

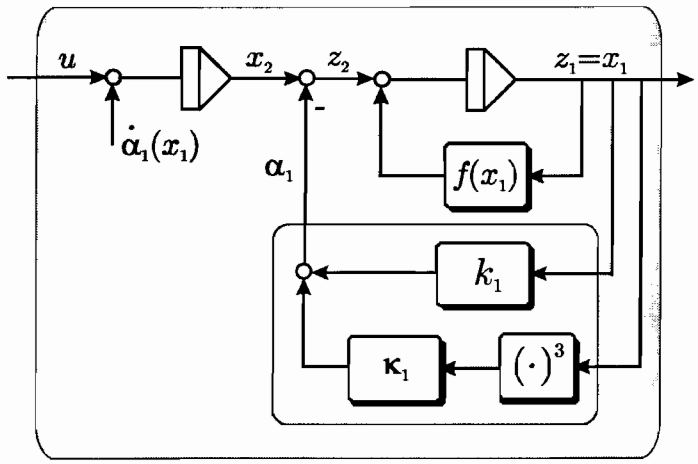


Figure 7.17: Domination of destabilizing terms by adding nonlinear damping.

be perfectly compensated for or dominated by adding a nonlinear damping term proportional to x_1^3 (remember that $z_1 = x_1$).

Nonlinear damping suggests the following candidate for the stabilizing function:

$$\alpha_1 = \underbrace{-k_1 z_1}_{\substack{\text{linear} \\ \text{damping}}} \quad \underbrace{-\kappa_1 z_1^3}_{\substack{\text{nonlinear} \\ \text{damping}}} \tag{7.228}$$

where $k_1 > 0$ and $\kappa_1 > 0$; see Figure 7.17. Hence:

$$\begin{aligned} \dot{z}_1 &= f(z_1) + (\alpha_1 + z_2) \\ &= -a_0 z_1 - a_1 z_1^2 - a_2 |z_1| z_1 - (k_1 + \kappa_1 z_1^2) z_1 + z_2 \\ &= -(\underbrace{a_0 + a_2 |z_1|}_{\text{good damping}} + k_1) z_1 - \underbrace{a_1 z_1^2}_{\text{bad damping}} - \kappa_1 z_1^3 + z_2 \end{aligned} \tag{7.229}$$

Consider the CLF:

$$V_1 = \frac{1}{2} z_1^2 \tag{7.230}$$

$$\dot{V}_1 = -(a_0 + a_2 |z_1| + k_1) z_1^2 - a_1 z_1^3 - \kappa_1 z_1^4 + z_1 z_2 \tag{7.231}$$

In the next step it is seen that:

$$\begin{aligned} V_2 &= V_1 + \frac{1}{2} z_2^2 \\ \dot{V}_2 &= -(\underbrace{a_0 + a_2 |z_1|}_{\text{energy dissipation}} + k_1) z_1^2 - \underbrace{a_1 z_1^3}_{\substack{\text{energy dissipation/} \\ \text{generation}}} - \kappa_1 z_1^4 + z_2 (z_1 + u - \dot{\alpha}_1) \end{aligned}$$

From this expression it can be concluded that the good damping terms contribute to the energy dissipation. The bad damping term, however, must be dominated by the nonlinear damping term. Choosing:

$$u = \dot{\alpha}_1 - k_2 z_2 - z_1 \tag{7.232}$$

finally yields:

$$\dot{V}_2 = -(a_0 + a_2 |z_1| + k_1) z_1^2 - a_1 z_1^3 - \kappa_1 z_1^4 - k_2 z_2^2 \tag{7.233}$$

This expression can be rewritten by *completing the squares*. Consider the expression:

$$\left(\frac{1}{2\sqrt{\kappa_1}} x + \sqrt{\kappa_1} y\right)^2 = \frac{1}{4\kappa_1} x^2 + xy + \kappa_1 y^2 \geq 0 \tag{7.234}$$

$$\Updownarrow$$

$$-xy - \kappa_1 y^2 = -\left(\frac{1}{2\sqrt{\kappa_1}} x + \sqrt{\kappa_1} y\right)^2 + \frac{1}{4\kappa_1} x^2 \tag{7.235}$$

Equation (7.233) with $x = a_1 z_1$ and $y = z_1^2$, yields:

$$\dot{V}_2 = -\left(\frac{a_1}{2\sqrt{\kappa_1}} z_1 + \sqrt{\kappa_1} z_1^2\right)^2 + \frac{a_1^2}{4\kappa_1} z_1^2 - (a_0 + a_2 |z_1| + k_1) z_1^2 - k_2 z_2^2 \tag{7.236}$$

Since:

$$\begin{aligned} -\left(\frac{a_1}{2\sqrt{\kappa_1}} z_1 + \sqrt{\kappa_1} z_1^2\right)^2 &\leq 0 \\ -a_2 |z_1| &\leq 0 \end{aligned} \tag{7.237}$$

it then follows that:

$$\dot{V}_2 \leq -\left(a_0 + k_1 - \frac{a_1^2}{4\kappa_1}\right) z_1^2 - k_2 z_2^2 \tag{7.238}$$

Hence, by choosing the controller gains according to:

$$\kappa_1 > 0 \tag{7.239}$$

$$k_1 > \frac{a_1^2}{4\kappa_1} - a_0 \tag{7.240}$$

$$k_2 > 0 \tag{7.241}$$

our design goal to render $\dot{V}_2 < 0$ is satisfied. Notice that the controller (7.232) with (7.228) is implemented without using the unknown parameters a_0, a_1, and a_2. Hence, a robust nonlinear controller is derived by using backstepping. This result is more attractive than the one obtained from feedback linearization which needs perfect model cancellation.

7.4.3 Backstepping of SISO Mass-Damper-Spring Systems

The results of Section 7.4.2 can be generalized to the following class of single-input single-output (SISO) mechanical systems:

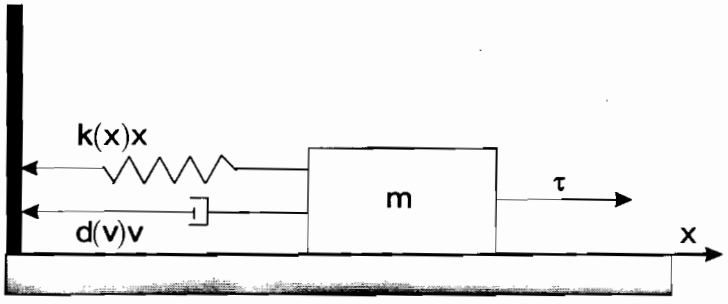


Figure 7.18: Nonlinear mass-damper-spring system.

$$\dot{x} = v \tag{7.242}$$
$$m\dot{v} + d(v)v + k(x)x = \tau \tag{7.243}$$
$$y = x \tag{7.244}$$

where x is the position, v is the velocity and:

m	=	mass (positive)
$d(v)v$	=	nonlinear damping force (non-negative)
$k(x)x$	=	nonlinear spring force (non-negative)

The nonlinear mass-damper-spring system is shown in Figure 7.18.

Nonlinear Tracking Control

Backstepping of the mass-damper-spring can be performed by choosing the output:

$$e = y - y_d \tag{7.245}$$

where e is the tracking error and $y_d(t) \in C^r$ is an r times differentiable (smooth) and bounded reference trajectory; see Section 5.1. Regulation of $y = x$ to zero is obtained by choosing $\dot{y}_d = y_d = 0$. Time differentiation of e yields the following model:

$$\dot{e} = v - \dot{y}_d \tag{7.246}$$
$$m\dot{v} = \tau - d(v)v - k(x)x \tag{7.247}$$

The backstepping control law solving this problem is derived in two recursive steps similar to the integrator backstepping example in Section 7.4.2.

Step 1:

Let $z_1 = e = y - y_d$, such that:

$$\dot{z}_1 = v - \dot{y}_d \tag{7.248}$$

Taking v as *virtual control*:

$$v = \alpha_1 + z_2 \tag{7.249}$$

where z_2 is a new state variable to be interpreted later, yields:

$$\dot{z}_1 = \alpha_1 + z_2 - \dot{y}_d \tag{7.250}$$

Next, the stabilizing function α_1 is chosen as:

$$\alpha_1 = \dot{y}_d - [k_1 + n_1(z_1)]z_1 \tag{7.251}$$

where $k_1 > 0$ is a feedback gain and $n_1(z_1) \geq 0$ is a nonlinear damping term e.g. a nonlinear nondecreasing function $n_1(z_1) = \kappa_1 |z_1|^{n_1}$ with $n_1 > 0$ and $\kappa_1 \geq 0$. This yields:

$$\dot{z}_1 = -[k_1 + n_1(z_1)]z_1 + z_2 \tag{7.252}$$

A CLF for z_1 is:

$$V_1 = \frac{1}{2}z_1^2 \tag{7.253}$$

$$\begin{aligned} \dot{V}_1 &= z_1\dot{z}_1 \\ &= -[k_1 + n_1(z_1)]z_1^2 + z_1 z_2 \end{aligned} \tag{7.254}$$

Step 2:

The second step stabilizes the z_2-dynamics. Moreover, from (7.249) it is seen that:

$$\begin{aligned} m\dot{z}_2 &= m\dot{v} - m\dot{\alpha}_1 \\ &= \tau - d(v)v - k(x)x - m\dot{\alpha}_1 \end{aligned} \tag{7.255}$$

Let V_2 be the second CLF which is chosen to reflect the kinetic energy $\frac{1}{2}mv^2$ of the system. However, it makes sense to replace the velocity v with z_2 in order to solve the trajectory tracking control problem. This is usually referred to as "pseudo kinetic energy":

$$V_2 = V_1 + \frac{1}{2}mz_2^2, \quad m > 0 \tag{7.256}$$

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + mz_2\dot{z}_2 \\ &= -[k_1 + n_1(z_1)]z_1^2 + z_1 z_2 + z_2[\tau - d(v)v - k(x)x - m\dot{\alpha}_1] \end{aligned} \tag{7.257}$$

Since the input τ appears in $\dot{V}_2$, a value for τ can be prescribed such that $\dot{V}_2$ becomes negative definite. For instance:

$$\tau = m\dot{\alpha}_1 + d(v)v + k(x)x - z_1 - k_2 z_2 - n_2(z_2)z_2 \tag{7.258}$$

where $k_2 > 0$ and $n_2(z_2) = \kappa_2 |z_2|^{n_2} \geq 0$ can be specified by the designer. This yields:

$$\dot{V}_2 = -[k_1 + n_1(z_1)]z_1^2 - [k_2 + n_2(z_2)]z_2^2 \tag{7.259}$$

When implementing the control law, $\dot{\alpha}_1$ is computed by taking the time derivative of α_1 along the trajectories of y_d and z_1, see (7.251). Hence:

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial \dot{y}_d}\ddot{y}_d - \frac{\partial \alpha_1}{\partial z_1}\dot{z}_1 = \ddot{y}_d - \frac{\partial \alpha_1}{\partial z_1}(v - \dot{y}_d) \tag{7.260}$$

in order to avoid the state derivatives in the control law. Notice that the desired state $y_d$is assumed to be smooth such that $\dot{y}_d$ and $\ddot{y}_d$ exist.

Resulting Error Dynamics

The resulting error dynamics is written:

$$\begin{bmatrix} 1 & 0 \\ 0 & m \end{bmatrix}\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = -\begin{bmatrix} k_1 + n_1(z_1) & 0 \\ 0 & k_2 + n_2(z_2) \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

$$\Updownarrow$$

$$\mathbf{M}\dot{\mathbf{z}} = -\mathbf{K}(\mathbf{z})\mathbf{z} + \mathbf{S}\mathbf{z} \tag{7.261}$$

where $\mathbf{z} = [z_1, z_2]^\top$ and

$$\begin{aligned} \mathbf{M} &= \text{diag}\{1, m\} \\ \mathbf{K}(\mathbf{z}) &= -\text{diag}\{k_1 + n_1(z_1), k_2 + n_2(z_2)\} \\ \mathbf{S} &= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \end{aligned}$$

Hence, the equilibrium point $(z_1, z_2) = (0, 0)$ is GES. This can be seen from $V_2(\mathbf{z}) = \frac{1}{2}\mathbf{z}^\top\mathbf{M}\mathbf{z}$ which after time differentiation yields $\dot{V}_2(\mathbf{z}) = -\mathbf{z}^\top\mathbf{K}\mathbf{z}$ since $\mathbf{z}^\top\mathbf{S}\mathbf{z} = 0, \forall \mathbf{z}$. Notice that kinetic energy has been applied in the Lyapunov analysis to achieve this.

Set-Point Regulation

Set-point regulation is obtained by choosing $\dot{y}_d = y_d = 0$. For simplicity let $n_1(z_1) = n_2(z_2) = 0$ such that:

$$\begin{aligned} z_1 &= x \\ \alpha_1 &= -k_1 z_1 \end{aligned}$$

and

$$\tau = m\dot{\alpha}_1 + d(v)v + k(x)x - z_1 - k_2 z_2 \tag{7.262}$$

Nonlinear PD-Control

The backstepping control law (7.262) can also be viewed as a nonlinear PD-control law:

$$u = -K_p(x)x - K_d(v)v \tag{7.263}$$

by writing (7.262) as:

$$\begin{aligned} u &= [d(v) - mk_1]v + [k(x) - 1]x - k_2(v + k_1 x) \\ &= [d(v) - mk_1 - k_2]v + [k(x) - 1 - k_1 k_2]x \end{aligned} \tag{7.264}$$

Hence:

$$K_p(x) = k_1 k_2 + 1 - k(x) \tag{7.265}$$

$$K_d(v) = mk_1 + k_2 - d(v) \tag{7.266}$$

Nonlinear PID-Control

The nonlinear PD-controller (7.263) can be extended to include integral action by using *constant parameter adaptation* or by *augmenting an additional integrator* to the plant. The following two methods for "backstepping with integral action" will be presented:

1. *Constant parameter adaptation:* An unknown constant (or slowly-varying) disturbance is added to the dynamic model. This constant or bias is estimated on-line by using adaptive control. The resulting system with parameter estimator can be shown to be UGAS for the case of regulation and trajectory tracking (Fossen *et al.* 2001).
2. *Integrator augmentation:* An additional integrator is augmented on the right-side of the integrator chain in order to obtain zero steady-state errors. The resulting system is proven to be GES.

7.4.4 Integral Action by Constant Parameter Adaptation

The constant parameter adaptation technique is based on Fossen *et al.* (2001). For simplicity a mass-damper-spring system is considered. Hence, adaptive backstepping results in a control law of PID-type since a 2nd-order mechanical system is used. Two representations of the integral controller are discussed depending on which state equation the disturbance w enters. These are:

1. *Integral action based on "matching" between the disturbance and the control input:*

$$\dot{x} = v \tag{7.267}$$
$$m\dot{v} + d(v)v + k(x)x = \tau + w \tag{7.268}$$
$$\dot{w} = 0 \tag{7.269}$$

   In this case the control input τ can compensate for the *unknown* parameter w directly (both terms are in the same state equation).

2. *Integral action based on "extended matching" between the disturbance and the control input:*

$$\dot{x} = v + w \tag{7.270}$$
$$m\dot{v} + d(v)v + k(x)x = \tau \tag{7.271}$$
$$\dot{w} = 0 \tag{7.272}$$

In the extended matching case there is a structural obstacle since the control law cannot be used to compensate for the unknown term w directly. This is due the fact that w and τ does not enter the same state equation. However, this problem can be solved by *adaptive backstepping*.

Case 1: Direct "matching" of the disturbance and the control input

The tracking control law for the "matching" case can be designed can by considering the tracking error:

$$z_1 = x - x_d \tag{7.273}$$

with:

$$\begin{aligned} \dot{z}_1 &= \dot{x} - \dot{x}_d \\ &= v - \dot{x}_d \\ &= (\alpha_1 + z_2) - v_d \end{aligned} \tag{7.274}$$

where z_2 is a new state variable and $v = \alpha_1 + z_2$ is viewed as the virtual control for z_1. Choosing the stabilizing control:

$$\alpha_1 = \dot{x}_d - k_1 z_1 \tag{7.275}$$

yields:

$$\dot{z}_1 = -k_1 z_1 + z_2 \tag{7.276}$$

The definition $z_2 := v - \alpha_1$ implies that:

$$\dot{z}_2 = \dot{v} - \ddot{x}_d + k_1(v - \dot{x}_d) \tag{7.277}$$

$$m\dot{z}_2 = \tau - d(v)v - k(x)x + w - m\ddot{x}_d + mk_1(v - \dot{x}_d) \tag{7.278}$$

Consider the CLF:

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2p}\tilde{w}^2, \;\; p > 0 \tag{7.279}$$

$$\begin{aligned} \dot{V}_1 &= z_1\dot{z}_1 + \frac{1}{p}\tilde{w}\dot{\tilde{w}} \\ &= z_1 z_2 - k_1 z_1^2 + \frac{1}{p}\tilde{w}\dot{\tilde{w}} \end{aligned} \tag{7.280}$$

where $\tilde{w} = \hat{w} - w$ is the parameter estimation error. Next, consider the CLF:

$$V_2 = V_1 + \frac{1}{2}mz_2^2 \tag{7.281}$$

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z_2(m\dot{z}_2) \\ &= z_1 z_2 - k_1 z_1^2 + \frac{1}{p}\tilde{w}\dot{\tilde{w}} \\ &\quad + z_2[\tau - d(v)v - k(x)x + w - m\ddot{x}_d + mk_1(v - \dot{x}_d)] \end{aligned} \tag{7.282}$$

where it is noticed that $\dot{\tilde{w}} = \dot{\hat{w}}$. Choosing the control law as:

$$\tau = d(v)\alpha_1 + k(x)x - \hat{w} + m\ddot{x}_d - mk_1(v - \dot{x}_d) - z_1 - k_2 z_2 \tag{7.283}$$

where $\alpha_1 = v - z_2$, yields:

$$\dot{V}_2 = -k_1 z_1^2 - [k_2 + d(v)]z_2^2 + \tilde{w}\left(\frac{1}{p}\dot{\hat{w}} - z_2\right) \leq 0 \tag{7.284}$$

Choosing the update law as:

$$\dot{\hat{w}} = pz_2 \tag{7.285}$$

finally yields:

$$\dot{V}_2 = -k_1 z_1^2 - [k_2 + d(v)]z_2^2 \tag{7.286}$$

The error dynamics takes the form:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 \\ -1 & -k_2 - d(v) \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tilde{w} \tag{7.287}$$

$$\dot{\tilde{w}} = -p \begin{bmatrix} 0 & -1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{7.288}$$

$$\Updownarrow$$

$$\dot{\mathbf{z}} = \mathbf{h}(\mathbf{z}, t) + \mathbf{b}\tilde{w} \tag{7.289}$$

$$\dot{\tilde{w}} = -p\mathbf{b}^\top \left(\frac{\partial W(\mathbf{z}, t)}{\partial \mathbf{z}} \right)^\top \tag{7.290}$$

Notice that the dissipative term $d(v) = d(z_2 + \alpha_1) = d(z_2 - k_1 z_1 + \dot{x}_d(t)) > 0, \forall v$ has not been "cancelled out" in order to exploit this as good damping in the error dynamics. The price for exploiting the so-called *"good nonlinearities"* in the design is that the error dynamics becomes *nonautonomous*. Since the feedback gains are assumed to be positive, that is $k_1 > 0$ and $k_2 > 0$, $p > 0$, $\mathbf{b} = [0, -1]^\top$, and $\mathbf{b}^\top \mathbf{b} = 1 > 0$, Theorem A.5 with $W(\mathbf{z}) = \frac{1}{2}\mathbf{z}^\top \mathbf{z}$ guarantees that the non-autonomous systems (7.287)–(7.288) is UGAS.

Notice that if a feedback linearizing controller is applied instead of (7.283) (replacing the damping term $d(v)\alpha_1$ with $d(v)v$):

$$\tau = d(v)v + k(x)x - \hat{w} + m\ddot{x}_d - mk_1(v - \dot{x}_d) - z_1 - k_2 z_2 \tag{7.291}$$

the error dynamic becomes ***autonomous***, that is:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 \\ -1 & -k_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tilde{w} \tag{7.292}$$

In this case, *Krasovskii–LaSalle's invariant set theorem* (Theorem A.2) can be used to prove GAS.

Case 2: "Extended matching" of the disturbance and the control input

In the extended matching case, backstepping is applied to overcome the structural obstacle (see Krstic *et al.* 1995). Let $z_1 = x - x_d$ denote the tracking error. Consider:

$$\begin{aligned} \dot{z}_1 &= \dot{x} - \dot{x}_d \\ &= v + w - \dot{x}_d \\ &:= (\alpha_1 + z_2) + w - \dot{x}_d \end{aligned} \tag{7.293}$$

where z_2 is a **new state** variable and $v = \alpha_1 + z_2$ is viewed as the virtual control for z_1. Choosing the stabilizing control:

$$\alpha_1 = \dot{x}_d - k_1 z_1 - \hat{w} \tag{7.294}$$

yields:

$$\dot{z}_1 = -k_1 z_1 - \hat{w} + z_2 \tag{7.295}$$

The definition $z_2 = v - \alpha_1$ implies that:

$$\dot{z}_2 = \dot{v} - \dot{v}_d + k_1(v - v_d) + k_1(\hat{w} - \tilde{w}) + \dot{\hat{w}} \tag{7.296}$$

Hence:

$$V_1 = \frac{1}{2} z_1^2 + \frac{1}{2p}\tilde{w}^2, \quad p > 0 \tag{7.297}$$

$$\begin{aligned} \dot{V}_1 &= z_1 \dot{z}_1 + \frac{1}{p}\tilde{w}\dot{\tilde{w}} \\ &= z_1 z_2 - k_1 z_1^2 + \tilde{w}\left(\frac{1}{p}\dot{\tilde{w}} - z_1\right) \end{aligned} \tag{7.298}$$

The choice of update law for $\hat{w}$ is postponed to the next step. Moreover:

$$V_2 = V_1 + \frac{1}{2} m z_2^2 \tag{7.299}$$

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z_2(m\dot{z}_2) \\ &= z_1 z_2 - k_1 z_1^2 + \tilde{w}\left(\frac{1}{p}\dot{\tilde{w}} - z_1 - mk_1 z_2\right) \\ &+ z_2[\tau - d(v)v - k(x)x - m\ddot{x}_d + mk_1(v - \dot{x}_d) + mk_1\hat{w} + m\dot{\hat{w}}] \end{aligned} \tag{7.300}$$

Choosing the parameter update law and control law as:

$$\dot{\hat{w}} = p(z_1 + mk_1 z_2) \tag{7.301}$$

and

$$\begin{aligned} \tau &= m\ddot{x}_d + d(v)\alpha_1 + k(x)x - mk_1(v - \dot{x}_d) - mk_1\hat{w} - m\dot{\hat{w}} - z_1 - k_2 z_2 \\ &= m\ddot{x}_d + d(v)\alpha_1 + k(x)x \\ &\quad -mk_1(v - \dot{x}_d) - mk_1\hat{w} - (mp + 1)z_1 - (mpk_1 + k_2)z_2 \end{aligned} \tag{7.302}$$

where $\alpha_1 = v - z_2$, yields:

$$\dot{V}_2 = -k_1 z_1^2 - [k_2 + d(v)]z_2^2 \tag{7.303}$$

The error dynamics takes the form:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 \\ -1 & -k_2 - d(v) \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} -1 \\ -mk_1 \end{bmatrix} \tilde{w} \tag{7.304}$$

$$\dot{\tilde{w}} = -p \begin{bmatrix} -1 & -mk_1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{7.305}$$

$$\Updownarrow$$

$$\dot{\mathbf{z}} = \mathbf{h}(\mathbf{z}, t) + \mathbf{b}\tilde{w} \tag{7.306}$$

$$\dot{\tilde{w}} = -p\mathbf{b}^\top \left(\frac{\partial W(\mathbf{z}, t)}{\partial \mathbf{z}} \right)^\top \tag{7.307}$$

Notice that the dissipative damping term $d(v) > 0, \forall v$ has not been "cancelled out" in the error dynamics. This implies that the resulting error dynamics is *non-autonomous*. The feedback gains are assumed to satisfy $k_1 > 0$ and $k_2 > 0$. Since $p > 0, \mathbf{b} = [-1, -mk_1]^\top$ and $\mathbf{b}^\top \mathbf{b} = 1 + m^2 k_1^2 > 0$. Theorem A.5 with $W(\mathbf{z}) = \frac{1}{2}\mathbf{z}^\top \mathbf{z}$ guarantees that the non-autonomous systems (7.304)–(7.305) is UGAS.

The difference between these two cases are that the extended matching case use both z_1 and z_2 to compute $\hat{w}$ while the direct matching case only uses z_2, that is the last z_i-variable $(i = 1, \ldots, n)$. Hence, it seems attractive to formulate a SISO backstepping control law with integral action using direct matching.

7.4.5 Integrator Augmentation Technique

Consider the 2nd-order mass-damper-spring system:

$$\dot{x} = v \tag{7.308}$$

$$m\dot{v} + d(v)v + k(x)x = \tau + w \tag{7.309}$$

$$y = x \tag{7.310}$$

where w is a constant *unknown* disturbance. Let e denote the tracking error:

$$e = y - y_d \tag{7.311}$$

where y_d is the desired output. Hence:

$$\dot{e} = v - \dot{y}_d \tag{7.312}$$

$$m\dot{v} + d(v)v + k(x)x = \tau + w \tag{7.313}$$

Nonlinear PD

If $w = 0$, backstepping results in a nonlinear control law of PD-type similar to the result in Section 7.4.3. However, by augmenting the plant with an additional integrator at the right end of the integrator chain, nonlinear PID-control can be obtained.

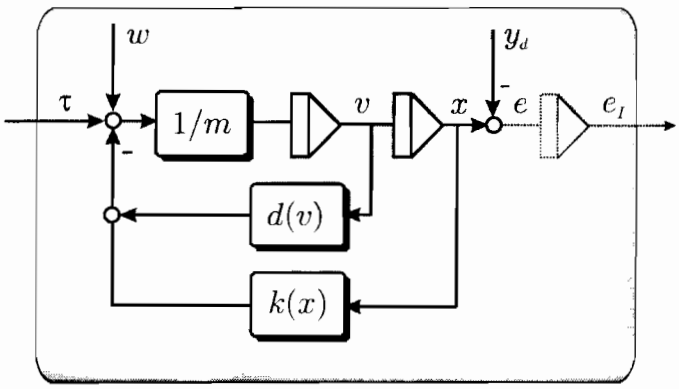


Figure 7.19: Augmentation of an additional integrator.

Nonlinear PID-Control

Augmentation of an additional integrator $\dot{e}_I = e$ to the 2nd-order plant (7.312)–(7.313), yields:

$$\begin{aligned} \dot{e}_I &= e && (7.314)\\ \dot{e} &= v - \dot{y}_d && (7.315)\\ m\dot{v} + d(v)v + k(x)x &= \tau + w && (7.316) \end{aligned}$$

For simplicity let us first assume that $w = 0$. Hence, backstepping with $z_1 = e_I$ results in three steps:

Step 1:

$$\begin{aligned} \dot{z}_1 &= e \\ &= \alpha_1 + z_2 && (7.317) \end{aligned}$$

Choosing the stabilizing function $\alpha_1 = -k_1 z_1$, yields:

$$\dot{z}_1 = -k_1 z_I + z_2 \qquad (7.318)$$

Hence:

$$\begin{aligned} V_1 &= \frac{1}{2} z_1^2 && (7.319)\\ \dot{V}_1 &= z_1 \dot{z}_1 \\ &= -k_1 z_1^2 + z_1 z_2 && (7.320) \end{aligned}$$

Step 2:

$$\begin{aligned} \dot{z}_2 &= \dot{e} - \dot{\alpha}_1 \\ &= v - \dot{y}_d - \dot{\alpha}_1 \\ &= (\alpha_2 + z_3) - \dot{y}_d - \dot{\alpha}_1 \end{aligned} \tag{7.321}$$

Hence:

$$V_2 = V_1 + \frac{1}{2} z_2^2 \tag{7.322}$$

$$\begin{aligned} \dot{V}_2 &= -k_1 z_1^2 + z_1 z_2 + z_2 \dot{z}_2 \\ &= -k_1 z_1^2 + z_2(z_1 + \alpha_2 + z_3 - \dot{y}_d - \dot{\alpha}_1) \end{aligned} \tag{7.323}$$

Choosing the stabilizing function $\alpha_2 = \dot{\alpha}_1 + \dot{y}_d - k_2 z_2 - z_1$, yields:

$$\dot{z}_2 = -z_1 - k_2 z_2 + z_3 \tag{7.324}$$

$$\dot{V}_2 = -k_1 z_1^2 - k_2 z_2^2 + z_2 z_3 \tag{7.325}$$

Step 3:

$$\begin{aligned} m\dot{z}_3 &= m\dot{v} - m\dot{\alpha}_2 \\ &= \tau + w - d(v)v - k(x)x - m\dot{\alpha}_2 \\ &= \tau - d(v)\alpha_2 - d(v)z_3 - k(x)x - m\dot{\alpha}_2 \end{aligned} \tag{7.326}$$

Let:

$$V_3 = V_2 + \frac{1}{2} m z_3^2 \tag{7.327}$$

$$\begin{aligned} \dot{V}_3 &= -k_1 z_1^2 - k_2 z_2^2 + z_3(z_2 + m\dot{z}_3) \\ &= -k_1 z_1^2 - k_2 z_2^2 + z_3(z_2 + \tau - d(v)\alpha_2 - d(v)z_3 - k(x)x - m\dot{\alpha}_2) \end{aligned} \tag{7.328}$$

Choosing the control law as:

$$\tau = m\dot{\alpha}_2 + d(v)\alpha_2 + k(x)x - z_2 - k_3 z_3 \tag{7.329}$$

yields:

$$\dot{V}_3 = -k_1 z_1^2 - k_2 z_2^2 - k_3 z_3^2 < 0, \ \forall z_1 \neq 0, z_2 \neq 0, z_3 \neq 0 \tag{7.330}$$

and:

$$m\dot{z}_3 = -[d(v) + k_3]z_3 - z_2 \tag{7.331}$$

Error Dynamics

For the undisturbed case $w = 0$, the error dynamics takes the form:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & m \end{bmatrix} \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = - \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & d(v) + k_3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \quad (7.332)$$

Hence, the equilibrium point $(z_1, z_2, z_3) = (0, 0, 0)$ is GES and therefore the tracking error e converges to zero. If $w =$ constant, the error dynamics takes the form:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & m \end{bmatrix} \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = - \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & d(v) + k_3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w$$

Hence, in steady-state ($\dot{\mathbf{z}} = \mathbf{0}$ and $d(v) = 0$) such that:

$$z_2 = k_1 z_1 = e - \alpha_1 = e + k_1 z_1 \quad \Rightarrow \quad \boldsymbol{e = 0} \quad (7.333)$$

The equilibrium point for $w =$ constant is:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} k_1 & -1 & 0 \\ 1 & k_2 & -1 \\ 0 & 1 & k_3 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w = \frac{1}{k_1 k_2 k_3 + k_1 + k_3} \begin{bmatrix} 1 \\ k_1 \\ 1 + k_1 k_2 \end{bmatrix} w \quad (7.334)$$

Therefore it can be concluded that for the case $w =$ constant the equilibrium point (z_1, z_2, z_3) is GES but (z_1, z_2, z_3) will converge to the constant non-zero values given by (7.334) even though $e = 0$. This shows that augmentation of an additional integrator when performing backstepping leads to zero steady-state errors in the case of regulation under the assumption of a constant disturbance w.

Implementations Considerations

The integrator augmentation technique is particular interesting for implementation on mechanical systems since the integral term is computed by integrating $z_1 = y - y_d$ which for a mechanical system is the position tracking error. This corresponds to applying a PID-controller on a 2nd-order system. On the contrary, when using *constant parameter adaptation* the integral term will be the integral of a linear combination of the state tracking errors, see (7.301). For a mechanical system this implies that both the position and velocity tracking errors are used to provide integral action. In many cases it is difficult to measure the velocity with the same accuracy as the position. This implies that the adaptive method will be more sensitive to measurement noise than the *integrator augmentation technique*.

7.4.6 Backstepping of MIMO Mass-Damper-Spring Systems

In this section *vectorial backstepping* is applied to mechanical systems and ship control (see Fossen and Berge 1997, Fossen and Grøvlen 1998). Consider a MIMO nonlinear mass-damper-spring system in the form:

$$\dot{\mathbf{x}} = \mathbf{v} \tag{7.335}$$

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{K}(\mathbf{x})\mathbf{x} = \mathbf{B}\mathbf{u} \tag{7.336}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the position vector, $\mathbf{v} \in \mathbb{R}^n$ is the velocity vector, $\mathbf{u} \in \mathbb{R}^r$ ($r \geq n$) is the control input vector, $\mathbf{D}(\mathbf{v}) \in \mathbb{R}^{n \times n}$ represents a matrix of damping coefficients, $\mathbf{K}(\mathbf{x}) \in \mathbb{R}^{n \times n}$ is a matrix of spring coefficients, $\mathbf{M} \in \mathbb{R}^{n \times n}$ is the inertia matrix, and $\mathbf{B} \in \mathbb{R}^{n \times r}$ is the input matrix. Hence, backstepping can be performed in *two vectorial steps*.

Step 1:

For the first system (7.335) consider **v as the control** and let:

$$\mathbf{v} = \mathbf{s} + \boldsymbol{\alpha}_1 \tag{7.337}$$

where:

$\mathbf{s} = \tilde{\mathbf{v}} + \boldsymbol{\Lambda}\tilde{\mathbf{x}}$ New state vector used for tracking control

$\boldsymbol{\alpha}_1$ Stabilizing vector field to be defined later

Here $\tilde{\mathbf{v}} = \mathbf{v} - \mathbf{v}_d$ and $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d$ are the velocity and position tracking errors, respectively, and $\boldsymbol{\Lambda} > 0$ is a diagonal matrix of positive elements. The definition of the s-vector is motivated by Slotine and Li (1987) who introduced s as a measure of tracking when designing their adaptive robot controller. It turns out that this transformation has the nice property of transforming the nonlinear state-space model to the form:

$$\begin{aligned} \mathbf{M}\dot{\mathbf{s}} + \mathbf{D}(\mathbf{v})\mathbf{s} &= \mathbf{M}\dot{\mathbf{v}} + \mathbf{D}(\mathbf{v})\mathbf{v} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r \\ &= \mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} \end{aligned} \tag{7.338}$$

where $\mathbf{v}_r$ can be interpreted as a *"virtual" reference trajectory:*

$$\begin{aligned} \mathbf{v}_r &= \mathbf{v} - \mathbf{s} \\ &= \mathbf{v}_d - \boldsymbol{\Lambda}\tilde{\mathbf{x}} \end{aligned} \tag{7.339}$$

The position error dynamics of Step 1 can therefore be written:

$$\begin{aligned} \dot{\tilde{\mathbf{x}}} &= \mathbf{v} - \mathbf{v}_d \\ &= \mathbf{s} + \boldsymbol{\alpha}_1 - \mathbf{v}_d \quad (\boldsymbol{\alpha}_1 = \mathbf{v}_r = \mathbf{v} - \mathbf{s}) \\ &= -\boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s} \end{aligned} \tag{7.340}$$

Hence:

$$V_1 = \frac{1}{2}\tilde{\mathbf{x}}^\top \mathbf{K}_p \tilde{\mathbf{x}}, \qquad \mathbf{K}_p = \mathbf{K}_p^\top > 0 \tag{7.341}$$

and

$$\begin{aligned}\dot{V}_1 &= \tilde{\mathbf{x}}^\top \mathbf{K}_p \dot{\tilde{\mathbf{x}}} \\ &= \tilde{\mathbf{x}}^\top \mathbf{K}_p(-\boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}) \\ &= -\tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}^\top \mathbf{K}_p \tilde{\mathbf{x}} \end{aligned} \tag{7.342}$$

Step 2:

In the second step we choose a CLF motivated by *"pseudo" kinetic energy*, that is:

$$V_2 = \frac{1}{2}\mathbf{s}^\top \mathbf{M}\mathbf{s} + V_1, \qquad \mathbf{M} = \mathbf{M}^\top > 0 \tag{7.343}$$

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{M}\dot{\mathbf{s}} + \dot{V}_1 \\ &= \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} + \mathbf{s}^\top \mathbf{K}_p \tilde{\mathbf{x}} \\ &= \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s} + \mathbf{K}_p\tilde{\mathbf{x}}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} \end{aligned} \tag{7.344}$$

Hence, we are ready to propose a control law e.g.:

$$\mathbf{B}\mathbf{u} = \mathbf{M}\dot{\mathbf{v}}_r + \mathbf{D}(\mathbf{v})\mathbf{v}_r + \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{K}_p\tilde{\mathbf{x}} - \mathbf{K}_d\mathbf{s}, \qquad \mathbf{K}_d > 0 \tag{7.345}$$

which results in:

$$\dot{V}_2 = -\mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}}$$

Since V_2 is positive definite and $\dot{V}_2$ is negative definite it follows from Theorem A.3 that the equilibrium point $(\tilde{\mathbf{x}}, \mathbf{s}) = (\mathbf{0}, \mathbf{0})$ is GES. Moreover, convergence of $\mathbf{s} \to \mathbf{0}$ and $\tilde{\mathbf{x}} \to \mathbf{0}$ implies that $\tilde{\mathbf{v}} \to \mathbf{0}$. When implementing the control law (7.345) it is assumed that $\mathbf{B}$ has an inverse:

$$\mathbf{B}^\dagger = \mathbf{B}^\top(\mathbf{B}\mathbf{B}^\top)^{-1} \tag{7.346}$$

or simply $\mathbf{B}^{-1}$ for the square case $r = n$.

Nonlinear Mass-Damper-Spring System with Actuator Dynamics

Consider the mass-damper-spring system of the previous section with actuator dynamics:

$$\dot{\mathbf{x}} = \boldsymbol{\nu} \tag{7.347}$$

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{K}(\mathbf{x})\mathbf{x} = \mathbf{B}\mathbf{u} \tag{7.348}$$

$$\mathbf{T}\dot{\mathbf{u}} + \mathbf{u} = \mathbf{u}_c \tag{7.349}$$

where $\mathbf{T} \in \mathbb{R}^{r\times r}$ is a diagonal matrix of actuator time constants and $\mathbf{u}_c \in \mathbb{R}^r$ is a vector of actuator commands. Instead of choosing the controller $\mathbf{u}$ in Step 2, $\mathbf{u}_c$ is treated as the control input to be specified in Step 3. Recall that:

$$\dot{V}_2 = \mathbf{s}^\top(\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\mathbf{v}}_r - \mathbf{D}(\mathbf{v})\mathbf{v}_r - \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{D}(\mathbf{v})\mathbf{s} + \mathbf{K}_p\tilde{\mathbf{x}}) - \tilde{\mathbf{x}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\mathbf{x}} \tag{7.350}$$

Step 3:

Let $\mathbf{Bu}$ be the virtual control vector of Step 3. Hence:

$$\mathbf{Bu} = \mathbf{z} + \boldsymbol{\alpha}_2 \tag{7.351}$$

$$\boldsymbol{\alpha}_2 = \mathbf{M}\dot{\mathbf{v}}_r + \mathbf{D}(\mathbf{v})\mathbf{v}_r + \mathbf{K}(\mathbf{x})\mathbf{x} - \mathbf{K}_p\tilde{\mathbf{x}} - \mathbf{K}_d\mathbf{s} \tag{7.352}$$

where $\mathbf{z}$ is a new state variable. This results in:

$$\dot{V}_2 = \mathbf{s}^\top\mathbf{z} - \mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top\mathbf{K}_p\boldsymbol{\Lambda}\tilde{\mathbf{x}} \tag{7.353}$$

Choosing:

$$V_3 = \frac{1}{2}\mathbf{z}^\top\mathbf{z} + V_2 \tag{7.354}$$

$$\begin{aligned}\dot{V}_3 &= \mathbf{z}^\top\mathbf{K}\dot{\mathbf{z}} + \dot{V}_2 \\ &= \mathbf{z}^\top(\mathbf{B}\dot{\mathbf{u}} - \dot{\boldsymbol{\alpha}}_2) + \mathbf{s}^\top\mathbf{z} - \mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top\mathbf{K}_p\boldsymbol{\Lambda}\tilde{\mathbf{x}} \\ &= \mathbf{z}^\top(\mathbf{B}\mathbf{T}^{-1}(\mathbf{u}_c - \mathbf{u}) - \dot{\boldsymbol{\alpha}}_2 + \mathbf{s}) - \mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top\mathbf{K}_p\boldsymbol{\Lambda}\tilde{\mathbf{x}}\end{aligned} \tag{7.355}$$

The control law:

$$\mathbf{u}_c = \mathbf{u} + \mathbf{T}\mathbf{B}^\dagger(\dot{\boldsymbol{\alpha}}_2 - \mathbf{s} - \mathbf{K}_z\mathbf{z}) \tag{7.356}$$

yields

$$\dot{V}_3 = -\mathbf{z}^\top\mathbf{K}_z\mathbf{z} - \mathbf{s}^\top(\mathbf{D}(\mathbf{v}) + \mathbf{K}_d)\mathbf{s} - \tilde{\mathbf{x}}^\top\mathbf{K}_p\boldsymbol{\Lambda}\tilde{\mathbf{x}} \tag{7.357}$$

Again GES is guaranteed. The main drawback of including the actuator dynamic is that $\dot{\boldsymbol{\alpha}}_2$ must be computed. The expression for $\dot{\boldsymbol{\alpha}}_2$ will not depend of the state derivatives since:

$$\dot{\boldsymbol{\alpha}}_2 = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\alpha}_2}{\partial(\text{state})_i}\overset{\bullet}{(\text{state})}_i$$

$$\overset{\bullet}{(\text{state})}_i = \text{“system equation depending on the states only”}$$

Example 7.8 (MIMO Backstepping of Robots)
This example is based on the results of Fossen and Berge (1997). Consider the nonlinear robot model (Sciavicco and Siciliano 1996):

$$\dot{\mathbf{q}} = \mathbf{v} \tag{7.358}$$

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{v}} + \mathbf{C}(\mathbf{q},\mathbf{v})\mathbf{v} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \tag{7.359}$$

where $\mathbf{M}(\mathbf{q}) = \mathbf{M}^\top(\mathbf{q}) > 0$ *is the inertia matrix,* $\mathbf{C}(\mathbf{q},\mathbf{v})$ *is a matrix of Coriolis and centripetal terms defined in terms of the* Christoffel symbols *and* $\mathbf{g}(\mathbf{q})$ *is a vector of gravitational forces and moments.* $\mathbf{q} \in \mathbf{R}^n$ *is a vector of joint angles,* $\mathbf{v} \in \mathbf{R}^n$ *is a vector of joint angular*

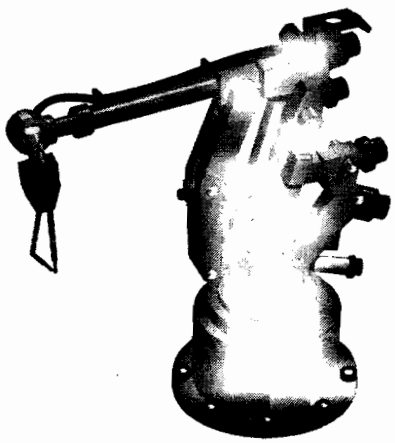


Figure 7.20: Robot manipulator.

rates and $\boldsymbol{\tau} \in \mathbf{R}^n$ *is a vector of control torques. Vectorial backstepping of a robot manipulator can be done in two steps:*

Step 1: *Define the* virtual control vector*:*

$$\dot{\mathbf{q}} = \mathbf{v} = \mathbf{s} + \boldsymbol{\alpha}_1 \tag{7.360}$$

where $\mathbf{s}$ *is a new state variable and* $\boldsymbol{\alpha}_1$ *is stabilizing function which can be chosen as:*

$$\boldsymbol{\alpha}_1 = \mathbf{v}_r, \quad \mathbf{v}_r = \nu_d - \boldsymbol{\Lambda}\tilde{\mathbf{q}} \tag{7.361}$$

where $\boldsymbol{\Lambda} > 0$ *is a diagonal design matrix and* $\tilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_d$ *is the tracking error. Combining (7.360) and (7.361) yields:*

$$\tilde{\mathbf{v}} = -\boldsymbol{\Lambda}\tilde{\mathbf{q}} + \mathbf{s} \tag{7.362}$$

where $\dot{\tilde{\mathbf{q}}} = \tilde{\mathbf{v}}$.

Step 2: *Consider the CLF:*

$$V = \frac{1}{2}\left(\mathbf{s}^\top \mathbf{M}(\mathbf{q})\mathbf{s} + \tilde{\mathbf{q}}^\top \mathbf{K}_q \tilde{\mathbf{q}}\right) > 0, \qquad \forall \mathbf{s} \neq \mathbf{0}, \tilde{\mathbf{q}} \neq \mathbf{0} \tag{7.363}$$

$$\begin{aligned} \dot{V} &= \mathbf{s}^\top \mathbf{M}(\mathbf{q})\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{M}}(\mathbf{q})\mathbf{s} + \tilde{\mathbf{q}}^\top \mathbf{K}_q \tilde{\mathbf{v}} \\ &= \mathbf{s}^\top \mathbf{M}(\mathbf{q})\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{M}}(\mathbf{q})\mathbf{s} - \tilde{\mathbf{q}}^\top \mathbf{K}_q \boldsymbol{\Lambda}\tilde{\mathbf{q}} + \tilde{\mathbf{q}}^\top \mathbf{K}_q \mathbf{s} \end{aligned} \tag{7.364}$$

Equations (7.360) and (7.361) can be combined to give:

$$\begin{aligned} \mathbf{M}(\mathbf{q})\dot{\mathbf{s}} &= \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} - \mathbf{M}(\mathbf{q})\dot{\boldsymbol{\alpha}} \\ &= \boldsymbol{\tau} - \mathbf{M}(\mathbf{q})\dot{\mathbf{v}}_r - \mathbf{C}(\mathbf{q}, \mathbf{v})\mathbf{v}_r - \mathbf{g}(\mathbf{q}) - \mathbf{C}(\mathbf{q}, \mathbf{v})\mathbf{s} \end{aligned} \tag{7.365}$$

Substituting (7.365) into (7.364) yields:

$$\begin{aligned}\dot{V} &= \mathbf{s}^\top\left(\boldsymbol{\tau}-\mathbf{M}(\mathbf{q})\dot{\mathbf{v}}_r-\mathbf{C}(\mathbf{q},\mathbf{v})\mathbf{v}_r-\mathbf{g}(\mathbf{q})+\mathbf{K}_q\tilde{\mathbf{q}}\right)\\ &\quad+\mathbf{s}^\top\left(\frac{1}{2}\dot{\mathbf{M}}(\mathbf{q})-\mathbf{C}(\mathbf{q},\mathbf{v})\right)\mathbf{s}-\tilde{\mathbf{q}}^\top\mathbf{K}_q\boldsymbol{\Lambda}\tilde{\mathbf{q}}\\ &= \mathbf{s}^\top\left(\boldsymbol{\tau}-\mathbf{M}(\mathbf{q})\dot{\mathbf{v}}_r-\mathbf{C}(\mathbf{q},\mathbf{v})\mathbf{v}_r-\mathbf{g}(\mathbf{q})+\mathbf{K}_q\tilde{\mathbf{q}}\right)\\ &\quad-\tilde{\mathbf{q}}^\top\mathbf{K}_q\boldsymbol{\Lambda}\tilde{\mathbf{q}}\end{aligned} \tag{7.366}$$

Here the skew-symmetric property $\mathbf{s}^\top(\frac{1}{2}\dot{\mathbf{M}}(\mathbf{q})-\mathbf{C}(\mathbf{q},\mathbf{v}))\mathbf{s}=\mathbf{0},\forall\mathbf{s}$ *has been applied. This suggests that the control law can be chosen as:*

$$\boldsymbol{\tau}=\mathbf{M}(\mathbf{q})\dot{\mathbf{v}}_r+\mathbf{C}(\mathbf{q},\mathbf{v})\mathbf{v}_r+\mathbf{g}(\mathbf{q})-\mathbf{K}_d\mathbf{s}-\mathbf{K}_q\tilde{\mathbf{q}} \tag{7.367}$$

where $\mathbf{K}_d=\mathbf{K}_d^\top>0$ *and* $\mathbf{K}_q=\mathbf{K}_q^\top>0$ *are design matrices. This finally yields:*

$$\dot{V}=-\mathbf{s}^\top\mathbf{K}_d\mathbf{s}-\tilde{\mathbf{q}}^\top\mathbf{K}_q\boldsymbol{\Lambda}\tilde{\mathbf{q}}<0\quad\forall\mathbf{s}\neq\mathbf{0},\tilde{\mathbf{q}}\neq\mathbf{0} \tag{7.368}$$

and GES follows. The control law (7.367) is equivalent with the control law of Slotine and Li (1987) with perfectly known parameters (non-adaptive case) except for the additional feedback term $\mathbf{K}_q\tilde{\mathbf{q}}$ *which is necessary to obtain GES and* $\boldsymbol{\Lambda}$ *which replaces the scalar weight* λ.

7.4.7 MIMO Backstepping of Ships

Conventional ship control systems are designed under the assumption that the kinematic and dynamic equations of motion can be linearized such that gain-scheduling techniques and optimal control theory can be applied (see Fossen 1994). This is not a good assumption for tracking applications where the surge and sway positions (n,e) and yaw angle ψ must be controlled simultaneously. The main reason for this, is that the rotation matrix in yaw, typically must be linearized about 36 operating points (steps of 10 degrees) to cover the whole circle arc with adequate accuracy. In addition to this, assumptions like linear damping and negligible Coriolis and centripetal forces are only good for low-speed applications, that is station-keeping and dynamic positioning (DP). These limitations clearly motivate a nonlinear design. MIMO nonlinear backstepping designs can be used for this purpose by exploiting nonlinear system properties like symmetry of the inertia matrix, dissipative damping and skew-symmetry of the Coriolis and centripetal matrix (see Fossen and Fjellstad 1995).

A MIMO nonlinear backstepping technique for marine vessels where the nonlinear system properties are exploited is presented below (Fossen and Strand 1998). An alternative reference is Fossen and Strand (1999*a*).

Vectorial Backstepping of Marine Vessels in 6 DOF

Consider a vessel described by the following model class:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{7.369}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{7.370}$$

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{u} \tag{7.371}$$

This model describes the motion of a vessel in n=6 DOF. It is assumed that $r \geq n$ control inputs are available.

The system (7.369)–(7.371) satisfies the following properties:

(i) $\mathbf{M} = \mathbf{M}^\top > 0 \Rightarrow \mathbf{x}^\top \mathbf{M}\mathbf{x} > 0, \forall \mathbf{x} \neq \mathbf{0}$

(ii) $\mathbf{C}(\boldsymbol{\nu}) = -\mathbf{C}^\top(\boldsymbol{\nu}) \in SS(6) \Rightarrow \mathbf{x}^\top \mathbf{C}(\boldsymbol{\nu})\mathbf{x} = \mathbf{0}, \forall \mathbf{x}$

(iii) $\mathbf{D}(\boldsymbol{\nu}) > 0 \Rightarrow \mathbf{x}^\top \mathbf{D}(\boldsymbol{\nu})\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(\mathbf{D}(\boldsymbol{\nu}) + \mathbf{D}^\top(\boldsymbol{\nu}))\mathbf{x} > 0, \forall \mathbf{x} \neq \mathbf{0}$

(iv) $\mathbf{B}\mathbf{B}^\top$ and $\mathbf{T}$ are non-singular

(v) $\mathbf{J}(\boldsymbol{\eta})$ =Euler angle transformation matrix (not defined for $\theta = \pm 90^o$)

Assumption (i) can be relaxed to systems with nonsymmetric inertia $\mathbf{M} \neq \mathbf{M}^\top$ by using acceleration feedback; see Section 7.1.5 or Fossen *et al.* (2002) for details.

New State Variables

Assume that the reference trajectory given by $\boldsymbol{\eta}_d^{(3)}, \ddot{\boldsymbol{\eta}}_d, \dot{\boldsymbol{\eta}}_d$, and $\boldsymbol{\eta}_d$ is smooth and bounded. The virtual reference trajectories in b- and n-frame coordinates are defined as:

$$\dot{\boldsymbol{\eta}}_r = \dot{\boldsymbol{\eta}}_d - \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} \tag{7.372}$$

$$\boldsymbol{\nu}_r = \mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r, \qquad \theta \neq \pm 90^o \tag{7.373}$$

where $\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \boldsymbol{\eta}_d$ is the tracking error and $\boldsymbol{\Lambda} > 0$ is a diagonal design matrix. Furthermore, let:

$$\mathbf{s} = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_r = \dot{\tilde{\boldsymbol{\eta}}} + \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} \tag{7.374}$$

The marine vessel dynamics (7.369)–(7.370) can be written (Fossen 1993*a*):

$$\mathbf{M}^*(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} + \mathbf{C}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \mathbf{D}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\dot{\boldsymbol{\eta}} + \mathbf{g}^*(\boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})\boldsymbol{\tau} \tag{7.375}$$

where:

$$\mathbf{M}^*(\boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})$$

$$\mathbf{C}^*(\boldsymbol{\nu}, \boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})[\mathbf{C}(\boldsymbol{\nu}) - \mathbf{M}\mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\mathbf{J}}(\boldsymbol{\eta})]\mathbf{J}^{-1}(\boldsymbol{\eta})$$

$$\mathbf{D}^*(\boldsymbol{\nu}, \boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{D}(\boldsymbol{\nu})\mathbf{J}^{-1}(\boldsymbol{\eta})$$

$$\mathbf{g}^*(\boldsymbol{\eta}) = \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{g}(\boldsymbol{\eta})$$

Hence, the marine vessel dynamics can be written in the following form:

$$\begin{aligned}\mathbf{M}^*(\boldsymbol{\eta})\dot{\mathbf{s}} = {} & -\mathbf{C}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\mathbf{s} - \mathbf{D}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\mathbf{s} + \mathbf{J}^{-\top}(\boldsymbol{\eta})\mathbf{B}\mathbf{u} \\ & -\mathbf{M}^*(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}}_r - \mathbf{C}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r - \mathbf{D}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\dot{\boldsymbol{\eta}}_r - \mathbf{g}^*(\boldsymbol{\eta})\end{aligned} \tag{7.376}$$

or equivalently:

$$\begin{aligned}\mathbf{M}^*(\boldsymbol{\eta})\dot{\mathbf{s}} = {} & -\mathbf{C}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\mathbf{s} - \mathbf{D}^*(\boldsymbol{\nu}, \boldsymbol{\eta})\mathbf{s} \\ & + \mathbf{J}^{-\top}(\boldsymbol{\eta})[\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\boldsymbol{\nu}}_r - \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{g}(\boldsymbol{\eta})]\end{aligned} \tag{7.377}$$

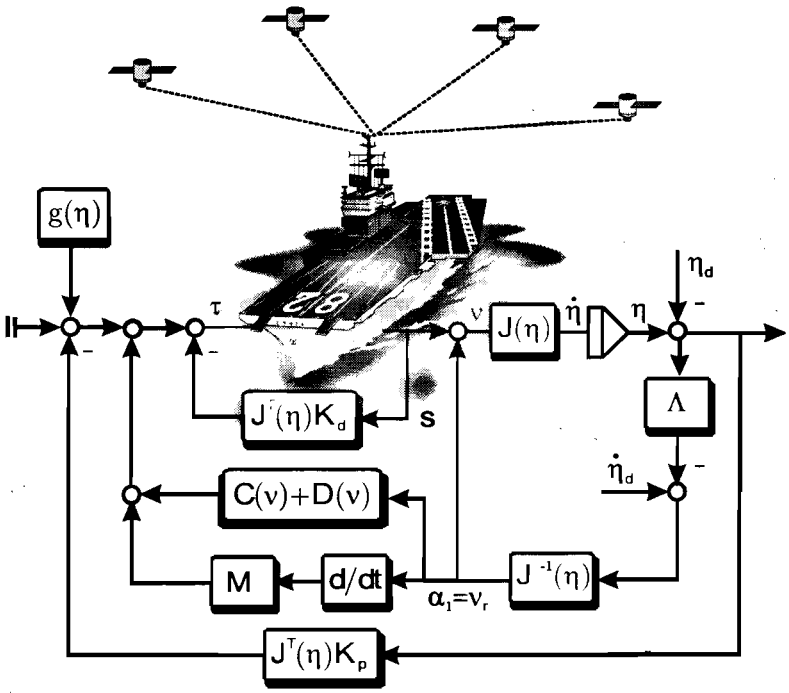


Figure 7.21: Nonlinear MIMO backstepping controller for 6 DOF control.

Step 1:

Consider the error dynamics:

$$\dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d = \mathbf{J}(\boldsymbol{\eta})(\boldsymbol{\nu} - \boldsymbol{\nu}_d) \tag{7.378}$$

Let $\boldsymbol{\nu}$ be the virtual control vector:

$$\boldsymbol{\nu} = \mathbf{s} + \boldsymbol{\alpha}_1 \tag{7.379}$$

The position error dynamics can therefore be written:

$$\begin{aligned} \dot{\tilde{\boldsymbol{\eta}}} &= \mathbf{J}(\boldsymbol{\eta})(\boldsymbol{\nu} - \boldsymbol{\nu}_d) \\ &= \mathbf{J}(\boldsymbol{\eta})(\mathbf{s} + \boldsymbol{\alpha}_1 - \boldsymbol{\nu}_d) \quad \{\boldsymbol{\alpha}_1 = \boldsymbol{\nu}_r = \mathbf{J}^{-1}(\boldsymbol{\eta})(\dot{\boldsymbol{\eta}}_\mathbf{d} - \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}})\} \\ &= \mathbf{J}(\boldsymbol{\eta})(\mathbf{s} + \mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}_d - \mathbf{J}^{-1}(\boldsymbol{\eta})\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} - \boldsymbol{\nu}_d) \\ &= -\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{J}(\boldsymbol{\eta})\mathbf{s} \end{aligned} \tag{7.380}$$

Hence, a CLF is:

$$V_1 = \frac{1}{2}\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}}, \quad \mathbf{K}_p = \mathbf{K}_p^\top > 0 \tag{7.381}$$

resulting in:

$$\begin{aligned}\dot{V}_1 &= \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \dot{\tilde{\boldsymbol{\eta}}} \\ &= \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p(-\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{J}(\boldsymbol{\eta})\mathbf{s}) \\ &= -\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{s}^\top \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}} \end{aligned} \tag{7.382}$$

Step 2:

In the second step a CLF motivated by the *"pseudo" kinetic energy* is chosen, that is:

$$V_2 = \frac{1}{2}\mathbf{s}^\top \mathbf{M}^*(\boldsymbol{\eta})\mathbf{s} + V_1, \quad \mathbf{M}^* = (\mathbf{M}^*)^\top > 0 \tag{7.383}$$

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{M}^*(\boldsymbol{\eta})\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{M}}^*(\boldsymbol{\eta})\mathbf{s} + \dot{V}_1 \\ &= -\mathbf{s}^\top[\mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta}) + \mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta})]\mathbf{s} \\ &\quad + \mathbf{s}^\top \mathbf{J}^{-\top}(\boldsymbol{\eta})[\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\boldsymbol{\nu}}_r - \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{g}(\boldsymbol{\eta})] \\ &\quad + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{M}}^*(\boldsymbol{\eta})\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{s}^\top \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}} \end{aligned} \tag{7.384}$$

Using the skew-symmetric property:

$$\mathbf{s}^\top \left(\dot{\mathbf{M}}^*(\boldsymbol{\eta}) - 2\mathbf{C}^*(\boldsymbol{\nu},\boldsymbol{\eta})\right)\mathbf{s} = 0, \qquad \forall \boldsymbol{\nu}, \boldsymbol{\eta}, \mathbf{s} \tag{7.385}$$

yields

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{J}^{-\top}(\boldsymbol{\eta})\left[\mathbf{B}\mathbf{u} - \mathbf{M}\dot{\boldsymbol{\nu}}_r - \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{g}(\boldsymbol{\eta}) + \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}}\right] \\ &\quad - \mathbf{s}^\top \mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta})\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} \end{aligned} \tag{7.386}$$

Hence, the control law can be chosen as (see Figure 7.21):

$$\boldsymbol{\tau} = \mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{g}(\boldsymbol{\eta}) - \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\tilde{\boldsymbol{\eta}} - \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_d\mathbf{s} \tag{7.387}$$

$$\mathbf{u} = \mathbf{B}^\dagger \boldsymbol{\tau} \tag{7.388}$$

where $\mathbf{K}_d > 0$. This results in:

$$\dot{V}_2 = -\mathbf{s}^\top(\mathbf{D}^*(\boldsymbol{\nu},\boldsymbol{\eta}) + \mathbf{K}_d)\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}}$$

Since V_2 is positive definite and $\dot{V}_2$ is negative definite it follows from Theorem A.3 that the equilibrium point $(\tilde{\boldsymbol{\eta}}, \mathbf{s}) = (\mathbf{0}, \mathbf{0})$ is GES. It follows from convergence of $\mathbf{s} \to \mathbf{0}$ and $\tilde{\boldsymbol{\eta}} \to \mathbf{0}$ that $\dot{\tilde{\boldsymbol{\eta}}} \to \mathbf{0}$.

Vectorial Backstepping in 3 DOF

Vectorial backstepping in 3 DOF (surge, sway, and yaw) is a special case of the general 6 DOF solution which can be applied for surface (floating) vessels. Typical applications are station-keeping and maneuvering of ships, semi-submersibles, and high-speed craft.

In this case the Euler angle transformation matrix $\mathbf{J}(\boldsymbol{\eta})$ reduces to (see (2.38) in Section 2.2):

$$\mathbf{J}(\boldsymbol{\eta}) \in \mathbb{R}^{6\times 6} \rightarrow \mathbf{R}(\psi) \in SO(3) \tag{7.389}$$

which is the rotation matrix in yaw. This implies that:

$$\mathbf{R}^{-1}(\psi) = \mathbf{R}^{\top}(\psi) \tag{7.390}$$

The dynamic equation (7.375) therefore becomes:

$$\mathbf{M}^{*}(\psi)\ddot{\boldsymbol{\eta}} + \mathbf{C}^{*}(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}} + \mathbf{D}^{*}(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\tau} \tag{7.391}$$

where the gravitational and buoyancy forces are assumed to outbalance each other such that $\mathbf{g}(\boldsymbol{\eta}) = \mathbf{0}$, and:

$$\begin{aligned} \mathbf{M}^{*}(\psi) &= \mathbf{R}(\psi)\mathbf{M}\mathbf{R}^{\top}(\psi) \\ \mathbf{C}^{*}(\boldsymbol{\nu},\psi) &= \mathbf{R}(\psi)[\mathbf{C}(\boldsymbol{\nu}) - \mathbf{M}\mathbf{R}^{\top}(\psi)\dot{\mathbf{R}}(\psi)]\mathbf{R}^{\top}(\psi) \\ \mathbf{D}^{*}(\boldsymbol{\nu},\psi) &= \mathbf{R}(\psi)\mathbf{D}(\boldsymbol{\nu})\mathbf{R}^{\top}(\psi) \end{aligned}$$

7.4.8 MIMO Backstepping Design with Acceleration Feedback

The results of the previous section can be extended to include acceleration feedback. A ship in surge, sway, and yaw will be considered to illustrate the design procedure. Recall from Sections 3.2.1 and 7.1.5 that $\mathbf{M}_A \neq \mathbf{M}_A^{\top}$ for ships moving at positive speed (maneuvering situations). However, acceleration feedback can be be used to symmetrize the system inertia matrix (Fossen *et al.* 2002). For simplicity a PD control law will be designed. Integral action can easily be included by using adaptive backstepping (Fossen *et al.* 2001); see Section 7.4.5.

The 3 DOF maneuvering model (surge, sway, and yaw) is:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{7.392}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau} \tag{7.393}$$

where:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{7.394}$$

Notice that $M_{23} \neq M_{32}$ (nonsymmetric). Acceleration feedback from only $\dot{u}$ and $\dot{v}$, implies that:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\mathrm{PD}} - \mathbf{K}_m\dot{\boldsymbol{\nu}} - \mathbf{C}_m(\boldsymbol{\nu})\boldsymbol{\nu} \tag{7.395}$$

where:

$$\mathbf{K}_m = \begin{bmatrix} K_{11} & K_{12} & 0 \\ K_{21} & K_{22} & 0 \\ K_{31} & K_{32} & 0 \end{bmatrix} \tag{7.396}$$

$$\mathbf{C}_m(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -K_{21}u - K_{22}v \\ 0 & 0 & K_{11}u + K_{12}v \\ K_{21}u + K_{22}v & -K_{11}u - K_{12}v & 0 \end{bmatrix} \tag{7.397}$$

The expression for $\mathbf{C}_m(\boldsymbol{\nu})$ in (7.395) is based on (3.100). Consequently, the system inertia matrix after acceleration feedback becomes:

$$\mathbf{H} = \mathbf{M} + \mathbf{K}_m = \begin{bmatrix} m - X_{\dot{u}} + K_{11} & K_{12} & 0 \\ K_{21} & m - Y_{\dot{v}} + K_{22} & mx_g - Y_{\dot{r}} \\ K_{31} & mx_g - N_{\dot{v}} + K_{32} & I_z - N_{\dot{r}} \end{bmatrix} \tag{7.398}$$

while

$$\mathbf{C}_H(\boldsymbol{\nu}) = \mathbf{C}(\boldsymbol{\nu}) + \mathbf{C}_m(\boldsymbol{\nu}) \tag{7.399}$$

The feedback term $\mathbf{C}_m(\boldsymbol{\nu})\boldsymbol{\nu}$ is necessary for:

$$\mathbf{s}^\top [\dot{\mathbf{H}}^*(\psi) - 2\mathbf{C}_H^*(\boldsymbol{\nu}, \psi)]\mathbf{s} = 0, \qquad \mathbf{s} \neq 0 \tag{7.400}$$

to hold for:

$$\mathbf{H}^*(\psi) = \mathbf{R}(\psi)\mathbf{H}\mathbf{R}^\top(\psi) \tag{7.401}$$

$$\mathbf{C}_H^*(\boldsymbol{\nu}, \psi) = \mathbf{R}(\psi)[\mathbf{C}_H(\boldsymbol{\nu}) - \mathbf{H}\mathbf{R}^\top(\psi)\dot{\mathbf{R}}(\psi)]\mathbf{R}^\top(\psi) \tag{7.402}$$

The control law (7.395) gives us some flexibility since the acceleration feedback terms K_{11}, K_{12}, K_{21}, K_{22}, K_{31}, and K_{32} can be chosen such that $\mathbf{H} = \mathbf{H}^\top > 0$. A symmetric expression independent of hydrodynamic added mass terms is obtained by choosing:

$$\mathbf{K}_m = \begin{bmatrix} K_{11} & K_{12} & 0 \\ K_{21} & K_{22} & 0 \\ K_{31} & K_{32} & 0 \end{bmatrix} = \begin{bmatrix} X_{\dot{u}} + \Delta K_{11} & 0 & 0 \\ 0 & Y_{\dot{v}} + \Delta K_{22} & 0 \\ 0 & N_{\dot{v}} - Y_{\dot{r}} & 0 \end{bmatrix} \tag{7.403}$$

where ΔK_{11} and ΔK_{22} can be treated as additional design parameters for the mass in the x- and y-directions. The resulting expression is:

$$\mathbf{H} = \begin{bmatrix} m + \Delta K_{11} & 0 & 0 \\ 0 & m + \Delta K_{22} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{7.404}$$

If $\Delta K_{11} = \Delta K_{22}$ (the mass in the x- and y-directions is equal) the PID controller will be independent of the heading angle which is advantageous when tuning a ship dynamic positioning system for instance.

The resulting dynamics after acceleration feedback is:

$$\mathbf{H}\dot{\boldsymbol{\nu}} + \mathbf{C}_H(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_{\mathrm{PD}} \tag{7.405}$$

Notice that $\mathbf{H}$ replaces $\mathbf{M}$ such that $\mathbf{H} = \mathbf{H}^\top > 0$ even though $\mathbf{M} \neq \mathbf{M}^\top$. Consider the CLF:

$$V_1 = \frac{1}{2}\mathbf{z}_1^\top \mathbf{K}_p \mathbf{z}_1, \qquad \mathbf{z}_1 = \boldsymbol{\eta}_d - \boldsymbol{\eta} \tag{7.406}$$

$$V_2 = V_1 + \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{H}\boldsymbol{\nu} \tag{7.407}$$

where V_1 and V_2 represent the "pseudo" potential and kinetic energy, respectively.

New State Variables
Assume that the reference trajectory, $\boldsymbol{\eta}_d^{(3)}$, $\ddot{\boldsymbol{\eta}}_d$, $\dot{\boldsymbol{\eta}}_d$, and $\boldsymbol{\eta}_d$, is smooth and bounded. A virtual reference trajectory is defined as:

$$\dot{\boldsymbol{\eta}}_r = \dot{\boldsymbol{\eta}}_d - \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}}, \qquad \boldsymbol{\nu}_r = \mathbf{R}^\top(\psi)\dot{\boldsymbol{\eta}}_r \tag{7.408}$$

where $\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \boldsymbol{\eta}_d$ is the tracking error and $\boldsymbol{\Lambda} > 0$ is a diagonal design matrix. Furthermore let:

$$\mathbf{s} = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_r = \dot{\tilde{\boldsymbol{\eta}}} + \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} \tag{7.409}$$

The vessel dynamics (7.392)–(7.405) can be transformed to:

$$\mathbf{H}^*(\psi)\ddot{\boldsymbol{\eta}} + \mathbf{C}_H^*(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}} + \mathbf{D}^*(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\tau}_{\mathrm{PD}}$$

Hence:

$$\begin{aligned} \mathbf{H}^*(\psi)\dot{\mathbf{s}} &= -\mathbf{C}_H^*(\boldsymbol{\nu},\psi)\mathbf{s} - \mathbf{D}^*(\boldsymbol{\nu},\psi)\mathbf{s} + \mathbf{R}(\psi)\boldsymbol{\tau}_{\mathrm{PD}} \\ &\quad -\mathbf{H}^*(\psi)\ddot{\boldsymbol{\eta}}_r - \mathbf{C}_H^*(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}}_r - \mathbf{D}^*(\boldsymbol{\nu},\psi)\dot{\boldsymbol{\eta}}_r \end{aligned}$$

or equivalently:

$$\mathbf{H}^*(\psi)\dot{\mathbf{s}} = -\mathbf{C}_H^*(\boldsymbol{\nu},\psi)\mathbf{s} - \mathbf{D}^*(\boldsymbol{\nu},\psi)\mathbf{s} + \mathbf{R}(\psi)[\boldsymbol{\tau}_{\mathrm{PD}} - \mathbf{H}\dot{\boldsymbol{\nu}}_r - \mathbf{C}_H(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r]$$

Step 1:
Consider the error dynamics:

$$\dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d = \mathbf{R}(\psi)[\boldsymbol{\nu} - \boldsymbol{\nu}_d] \tag{7.410}$$

Let $\boldsymbol{\nu}$ be the virtual control vector $\boldsymbol{\nu} = \mathbf{s} + \boldsymbol{\alpha}_1$. The position error dynamics can therefore be written:

$$\begin{aligned} \dot{\tilde{\boldsymbol{\eta}}} &= \mathbf{R}(\psi)[\boldsymbol{\nu} - \boldsymbol{\nu}_d] \\ &= \mathbf{R}(\psi)[\mathbf{s} + \boldsymbol{\alpha}_1 - \boldsymbol{\nu}_d], \qquad \{\boldsymbol{\alpha}_1 = \boldsymbol{\nu}_r = \mathbf{R}^\top(\psi)[\dot{\boldsymbol{\eta}}_d - \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}}]\} \\ &= \mathbf{R}(\psi)[\mathbf{s} + \mathbf{R}^\top(\psi)\dot{\boldsymbol{\eta}}_d - \mathbf{R}^\top(\psi)\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} - \boldsymbol{\nu}_d] \\ &= -\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{R}(\psi)\mathbf{s} \end{aligned} \tag{7.411}$$

Hence:

$$V_1 = \frac{1}{2}\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}}, \quad \mathbf{K}_p = \mathbf{K}_p^\top > 0 \tag{7.412}$$

and

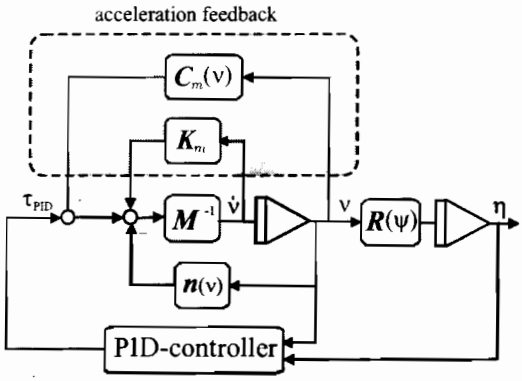


Figure 7.22: **Acceleration feedback and PID-controller.**

$$\begin{aligned}\dot{V}_1 &= \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \dot{\tilde{\boldsymbol{\eta}}} \\ &= \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p(-\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{R}(\psi)\mathbf{s}) \\ &= -\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p \boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{s}^\top \mathbf{R}^\top(\psi)\mathbf{K}_p\tilde{\boldsymbol{\eta}}\end{aligned} \tag{7.413}$$

Step 2:
In the second step we choose a CLF motivated by *"pseudo" kinetic energy*, that is:

$$V_2 = \frac{1}{2}\mathbf{s}^\top \mathbf{H}^*(\psi)\mathbf{s} + V_1 \tag{7.414}$$

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{H}^*(\psi)\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{H}}^*(\psi)\mathbf{s} + \dot{V}_1 \\ &= \mathbf{s}^\top \left[-\mathbf{C}_H^*(\boldsymbol{\nu},\psi)\mathbf{s} - \mathbf{D}^*(\boldsymbol{\nu},\psi)\mathbf{s} + \mathbf{R}(\psi)\left[\boldsymbol{\tau}_{\rm PD} - \mathbf{H}\dot{\boldsymbol{\nu}}_r - \mathbf{C}_H(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r\right]\right] \\ &\quad + \frac{1}{2}\mathbf{s}^\top \dot{\mathbf{H}}^*(\psi)\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}} + \mathbf{s}^\top \mathbf{R}^\top(\psi)\mathbf{K}_p\tilde{\boldsymbol{\eta}}\end{aligned}$$

Using the skew-symmetric property $\mathbf{s}^\top[\dot{\mathbf{H}}^*(\psi) - 2\mathbf{C}_H^*(\boldsymbol{\nu},\psi)]\mathbf{s} = 0$, yields:

$$\begin{aligned}\dot{V}_2 &= \mathbf{s}^\top \mathbf{R}(\psi)[\boldsymbol{\tau}_{\rm PD} - \mathbf{H}\dot{\boldsymbol{\nu}}_r - \mathbf{C}_H(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{R}^\top(\psi)\mathbf{K}_p\tilde{\boldsymbol{\eta}}] \\ &\quad - \mathbf{s}^\top \mathbf{D}^*(\boldsymbol{\nu},\psi)\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}}\end{aligned}$$

Consequently, the 3 DOF control law:

$$\boldsymbol{\tau}_{\rm PD} = \mathbf{H}\dot{\boldsymbol{\nu}}_r + \mathbf{C}_H(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r - \mathbf{R}^\top(\psi)[\mathbf{K}_p\tilde{\boldsymbol{\eta}} + \mathbf{K}_d\mathbf{s}] \tag{7.415}$$

results in:

$$\dot{V}_2 = -\mathbf{s}^\top(\mathbf{D}^*(\boldsymbol{\nu},\psi) + \mathbf{K}_d)\mathbf{s} - \tilde{\boldsymbol{\eta}}^\top \mathbf{K}_p\boldsymbol{\Lambda}\tilde{\boldsymbol{\eta}}$$

Since V_2 is positive definite and $\dot{V}_2$ is negative definite it follows that the equilibrium point $(\tilde{\boldsymbol{\eta}}, \mathbf{s}) = (\mathbf{0}, \mathbf{0})$ is GES. Moreover, convergence of $\mathbf{s} \to \mathbf{0}$ and $\tilde{\boldsymbol{\eta}} \to \mathbf{0}$ implies that $\dot{\tilde{\boldsymbol{\eta}}} \to \mathbf{0}$. The PD controller can easily be replaced with a PID controller by using the adaptive backstepping technique of Fossen *et al.* (2001) which is referred to as *backstepping with integral action;* see Section 7.4.5. In this case only UGAS is guaranteed.

7.5 Control Allocation

For marine vessels in n DOF it is necessary to distribute the generalized control forces $\boldsymbol{\tau} \in \mathbb{R}^n$ to the actuators in terms of control inputs $\mathbf{u} \in \mathbb{R}^r$ as shown in Figure 7.23. If $r > n$ this is an *overactuated control* problem while $r < n$ is referred to as *underactuated control*. The input matrix is square for $r = n$, that is, the number of actuators is equal to n DOF.

Computation of $\mathbf{u}$ from $\boldsymbol{\tau}$ is a model-based optimization problem which in its simplest form is unconstrained while physical limitations like input amplitude and rate saturations imply that a constrained optimization problem must be solved. Another complication is actuators that can be rotated at the same time as they produce control forces. An example is azimuth thrusters on an offshore supply vessel. This increases the number of available controls from r to $r + p$, where p denotes the number of rotatable actuators for which additional nonlinearities are introduced.

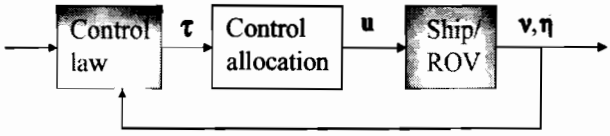


Figure 7.23: Block diagram showing the control allocation block in a feedback control system.

7.5.1 Actuator Models

The control force due to a propeller, a rudder, or a fin can be written (assuming linearity):

$$F = k\,u \tag{7.416}$$

where k is the force coefficient and u is the control input depending on the actuator considered; see Table 7.1. The linear model $F = ku$ can also be used to describe nonlinear monotonic control forces. For instance, if the rudder force F is quadratic in rudder angle δ, that is $F = k\,\delta\,|\delta|$, the choice $u = \delta\,|\delta|$ which has a unique inverse $\delta = \text{sign}(u)\sqrt{|u|}$ satisfies (7.416).

For marine vessels the most common actuators are:

- **Main propellers:** the main propellers of the vessel are mounted aft of the hull usually in conjunction with rudders. They produce the necessary force F_x in the x-direction needed for transit.
- **Tunnel thrusters:** transverse thrusters going through the hull of the vessel. The propeller unit is mounted inside a transverse tube and it produces a force F_y in the y-direction. Tunnel thrusters are only effective at low speed which limits their use to low-speed maneuvering and dynamic positioning.
- **Azimuth thrusters:** thruster units that can be rotated an angle α about the z-axis and produce two force components (F_x, F_y) in the horizontal plane are usually referred to

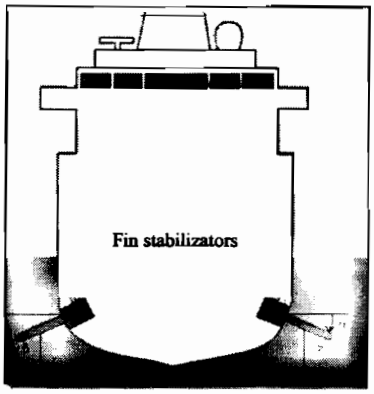


Figure 7.24: Fin stabilized ship where the vertical force $F = ku$ is proportional to the angle u for small deflections.

as azimuthing thrusters. They are usually mounted under the hull of the vessel and the most sophisticated units are retractable. Azimuth thrusters are attractive in dynamic positioning systems since they can produce forces in different directions leading to an overactuated control problem that can be optimized with respect to power and possible failure situations.

- **Aft rudders:** rudders are the primary steering device for conventional vessels. They are located aft of the vessel and the rudder force F_y will be a function of the rudder deflection. A rudder force in the y-direction will produce a yaw moment which can be used for steering control.

- **Stabilizing fins**: stabilizing fins are used for damping of vertical vibrations and roll motions. They produce a force F_z in the z-directions which is a function of the fin deflection. For small angles this relationship is linear. Fin stabilizators can be retractable allowing for selective use in bad weather. The lift forces are small at low speed so the most effective operating condition is in transit.

- **Control surfaces**: control surfaces can be mounted at different locations to produce lift and drag forces. For underwater vehicles these could be fins for diving, rolling, and pitching, rudders for steering, etc.

- **Water jets:** water jets is an alternative to main propellers aft of the ship. They are usually used for high-speed craft.

Table 7.1 implies that the forces and moments in 6 DOF corresponding to the force vector

$\mathbf{f} = [F_x, F_y, F_z]^\top$ can be written:

$$\boldsymbol{\tau} = \begin{bmatrix} \mathbf{f} \\ \mathbf{r} \times \mathbf{f} \end{bmatrix} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ F_z l_y - F_y l_z \\ F_x l_z - F_z l_x \\ F_y l_x - F_x l_y \end{bmatrix} \overset{\text{4 DOF}}{\Longrightarrow} \boldsymbol{\tau} = \begin{bmatrix} F_x \\ F_y \\ F_x l_y - F_y l_z \\ F_y l_x - F_x l_y \end{bmatrix} \tag{7.417}$$

where $\mathbf{r} = [l_x, l_y, l_z]^\top$ are the moment arms. For rotatable (azimuth) thrusters the control force F will be a function of the rotation angle α; see Figure 7.25. Consequently, an azimuth thruster in the horizontal plane will have two force components $F_x = F\cos\alpha$ and $F_y = F\sin\alpha$, while the main propeller aft of the ship only produces a longitudinal force $F_x = F$, see Table 7.1.

Table 7.1: Definition of actuators and control variables.

Actuator	u (control input)	α (control input)	$\mathbf{f}^\top$ (force vector)
main propellers (longitudinal)	pitch and rpm	–	$[F, 0, 0]$
tunnel thrusters (transverse)	pitch and rpm	–	$[0, F, 0]$
azimuth (rotatable) thruster	pitch and rpm	angle	$[F\cos\alpha, F\sin\alpha, 0]$
aft rudders	angle	–	$[0, F, 0]$
stabilizing fins	angle	–	$[0, 0, F]$

A unified representation of control forces and moments is:

$$\boldsymbol{\tau} = \mathbf{T}(\alpha)\underbrace{\mathbf{K}\mathbf{u}}_{\mathbf{f}} \tag{7.418}$$

where $\mathbf{u} \in \mathbb{R}^r$ and $\boldsymbol{\alpha} \in \mathbb{R}^p$ are control inputs defined by the vectors:

$$\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_p]^\top, \qquad \mathbf{u} = [u_1, ..., u_r]^\top \tag{7.419}$$

and $\mathbf{f} = \mathbf{K}\mathbf{u} \in \mathbb{R}^r$ is a vector of control forces.

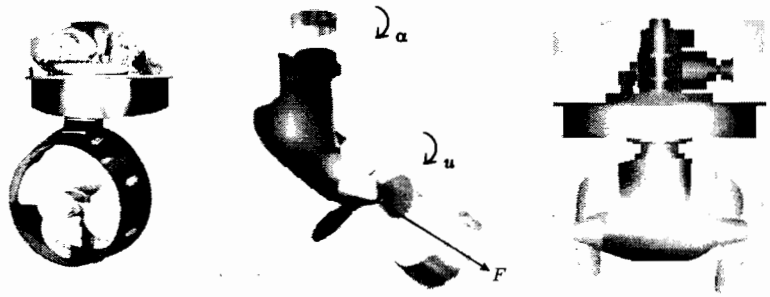
Force Coefficient Matrix

The force coefficient matrix $\mathbf{K} \in \mathbb{R}^{r \times r}$ is always diagonal such that:

$$\mathbf{K} = \text{diag}\{K_1, ..., K_r\}, \qquad \mathbf{K}^{-1} = \text{diag}\left\{\frac{1}{K_1}, ..., \frac{1}{K_r}\right\} \tag{7.420}$$

Actuator Configuration Matrix

The actuator configuration matrix $\mathbf{T}(\boldsymbol{\alpha}) \in \mathbb{R}^{n \times r}$ is defined in terms of a set of column vectors $\mathbf{t}_i \in \mathbb{R}^n$:



Azimuth thruster | Podded propeller | Contra-rotating propeller

Figure 7.25: Propellers that can be rotated an angle α to produce a force F in an arbitrary direction.

$$\mathbf{T}(\boldsymbol{\alpha}) = [\mathbf{t}_1, ..., \mathbf{t}_r] \tag{7.421}$$

In 4 DOF (*surge, sway, roll,* and *yaw*) the column vectors take the following form:

$$\underbrace{\mathbf{t}_i = \begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \\ l_{y_i}\cos\alpha_i - l_{z_i}\sin\alpha_i \\ l_{x_i}\sin\alpha_i - l_{y_i}\cos\alpha_i \end{bmatrix}}_{\text{azimuth thruster}}, \quad \underbrace{\mathbf{t}_i = \begin{bmatrix} 1 \\ 0 \\ 0 \\ l_{y_i} \end{bmatrix}}_{\text{main propeller}}, \quad \underbrace{\mathbf{t}_i = \begin{bmatrix} 0 \\ 1 \\ -l_{z_i} \\ l_{x_i} \end{bmatrix}}_{\text{tunnel thruster and aft rudder}}, \quad \underbrace{\mathbf{t}_i = \begin{bmatrix} 0 \\ 0 \\ l_{y_i} \\ 0 \end{bmatrix}}_{\text{stabilizing fin}}$$

Examples using this representation are found in Section 9.3 (fin and rudder control systems) and 11.2 (dynamic positioning systems). The 3 DOF representation (surge, sway and yaw) is found by deleting the 3rd row (roll) in the expressions for $\mathbf{t}_i$.

7.5.2 Unconstrained Control Allocation (Nonrotatable Actuators)

The simplest allocation problem is the one where all control forces are produced by thrusters in fixed directions alone or in combination with rudders and control surfaces. This implies that:

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 = \text{constant}, \qquad \mathbf{T} = \mathbf{T}(\boldsymbol{\alpha}_0) \tag{7.422}$$

It will be assumed that the allocation problem is *unconstrained*–i.e., there are no bounds on the vector elements f_i, α_i, and u_i, and their time derivatives. Saturating control and constrained control allocation are discussed in Sections 7.5.3–7.5.4.

For marine craft where the configuration matrix $\mathbf{T}$ is square or non-square ($r \geq n$), that is there are equal or more control inputs than controllable DOF, it is possible to find an "optimal"

distribution of control forces $\mathbf{f}$, for each DOF by using an explicit method. Consider the unconstrained least-squares (LS) optimization problem (Fossen and Sagatun 1991*a*):

$$J = \min_{\mathbf{f}} \left\{ \mathbf{f}^\top \mathbf{W} \mathbf{f} \right\} \quad \text{subject to:} \quad \boldsymbol{\tau} - \mathbf{T}\mathbf{f} = \mathbf{0} \tag{7.423}$$

Here $\mathbf{W}$ is a positive definite matrix, usually diagonal, weighting the control forces. For marine craft which have both control surfaces and propellers, the elements in $\mathbf{W}$ should be selected such that using the control surfaces is much more inexpensive than using the propellers.

Explicit Solution to the LS Optimization Problem using Lagrange Multipliers

Define the Lagrangian (Fossen 1994):

$$L(\mathbf{f}, \boldsymbol{\lambda}) = \mathbf{f}^\top \mathbf{W} \mathbf{f} + \boldsymbol{\lambda}^\top (\boldsymbol{\tau} - \mathbf{T}\mathbf{f}) \tag{7.424}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^r$ is a vector of Lagrange multipliers. Consequently, differentiating the Lagrangian L with respect to $\mathbf{f}$, yields:

$$\frac{\partial L}{\partial \mathbf{f}} = 2\mathbf{W}\mathbf{f} - \mathbf{T}^\top \boldsymbol{\lambda} = \mathbf{0} \Rightarrow \mathbf{f} = \frac{1}{2}\mathbf{W}^{-1}\mathbf{T}^\top \boldsymbol{\lambda} \tag{7.425}$$

Next, assume that $\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top$ is non-singular such that:

$$\boldsymbol{\tau} = \mathbf{T}\mathbf{f} = \frac{1}{2}\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top \boldsymbol{\lambda} \Rightarrow \boldsymbol{\lambda} = 2(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top)^{-1}\boldsymbol{\tau} \tag{7.426}$$

Substituting the Lagrange multipliers $\boldsymbol{\lambda} = 2(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top)^{-1}\boldsymbol{\tau}$ into (7.425) yields:

$$\mathbf{f} = \underbrace{\mathbf{W}^{-1}\mathbf{T}^\top(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top)^{-1}}_{\mathbf{T}_w^\dagger}\boldsymbol{\tau} \tag{7.427}$$

where the matrix:

$$\mathbf{T}_w^\dagger = \mathbf{W}^{-1}\mathbf{T}^\top(\mathbf{T}\mathbf{W}^{-1}\mathbf{T}^\top)^{-1} \tag{7.428}$$

is recognized as the *generalized inverse*. For the case $\mathbf{W} = \mathbf{I}$, that is equally weighted control forces, (7.428) reduces to the *Moore–Penrose pseudo inverse*:

$$\mathbf{T}^\dagger = \mathbf{T}^\top(\mathbf{T}\mathbf{T}^\top)^{-1} \tag{7.429}$$

Since:

$$\mathbf{f} = \mathbf{T}_w^\dagger \boldsymbol{\tau} \tag{7.430}$$

the control input vector $\mathbf{u}$ can be computed from (7.418) as:

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{T}_w^{\dagger}\boldsymbol{\tau} \tag{7.431}$$

Notice that this solution is valid for all $\boldsymbol{\alpha}_0$ but not optimal with respect to a time-varying $\boldsymbol{\alpha}_0$ (only $\mathbf{f}$). Optimality with respect to $\boldsymbol{\alpha}$ in addition to (7.425) is discussed in Section 7.5.4.

Matlab:
The generalized inverse for the case $\mathbf{T} = \mathbf{T}(\boldsymbol{\alpha}_0)$=constant is implemented in the GNC Toolbox as:

```
u=ucalloc(K,T,W,tau)
```

7.5.3 Constrained Control Allocation (Nonrotatable Actuators)

In industrial systems it is important to minimize the power consumption by taking advantage of the additional control forces in an overactuated control problem. From a safety critical point of view it is also important to take into account actuator limitations like saturation, tear and wear as well as other constraints. In general this leads to a *constrained* optimization problem.

Explicit Solution using Piecewise Linear Functions

Recently, an explicit solution approach for parametric quadratic programming has been developed by Tøndel *et al.* (2001). An on-line algorithm is presented in Tøndel *et al.* (2003) while applications to marine vessels are presented by Johansen *et al.* (2002). In this work the constrained optimization problem is formulated as:

$$\begin{aligned} J = \min_{\mathbf{f},\mathbf{s},\bar{f}} & \{\mathbf{f}^\top\mathbf{W}\mathbf{f} + \mathbf{s}^\top\mathbf{Q}\mathbf{s} + \beta\bar{f}\} \\ \text{subject to:} & \\ & \mathbf{T}\mathbf{f} = \boldsymbol{\tau} + \mathbf{s} \\ & \mathbf{f}_{\min} \le \mathbf{f} \le \mathbf{f}_{\max} \\ & -\bar{f} \le f_1, f_2, \ldots f_r \le \bar{f} \end{aligned} \tag{7.432}$$

where $\mathbf{s} \in \mathbb{R}^n$ is a vector of *slack variables*. The first term of the criterion corresponds to the LS criterion (7.423), while the second term is introduced to minimize the largest force $\bar{f} = \max_i |f_i|$ among the actuators. The constant $\beta \ge 0$ controls the relative weighting of the two criteria. This formulation ensures that the constraints $f_i^{\min} \le f_i \le f_i^{\max}$ $(i = 1, ..., r)$ are satisfied, if necessary by allowing the resulting generalized force $\mathbf{T}\mathbf{f}$ to deviate from its specification $\boldsymbol{\tau}$. To achieve accurate generalized force, the slack variable should be close to zero. This is obtained by choosing the weighting matrix $\mathbf{Q} \gg \mathbf{W} > 0$. Moreover, saturation is handled in an optimal manner by minimizing the combined criterion (7.432).

Letting $\mathbf{z} = [\mathbf{f}^\top, \mathbf{s}^\top, \bar{f}]^\top \in \mathbb{R}^{r+n+1}$ and $\mathbf{p} = [\boldsymbol{\tau}^\top, \mathbf{f}_{\min}^\top, \mathbf{f}_{\max}^\top, \beta]^\top \in \mathbb{R}^{n+2r+1}$ denotes the parameter vector, it is straightforward to see that the optimization problem (7.432) can be reformulated as a QP problem:

$$J = \min_{\mathbf{z}} \left\{ \mathbf{z}^\top \mathbf{\Phi} \mathbf{z} + \mathbf{z}^\top \mathbf{R} \mathbf{p} \right\}$$
$$\text{subject to:} \tag{7.433}$$
$$\mathbf{A}_1 \mathbf{z} = \mathbf{C}_1 \mathbf{p}$$
$$\mathbf{A}_2 \mathbf{z} \leq \mathbf{C}_2 \mathbf{p}$$

where:

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{W} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{n\times r} & \mathbf{Q} & \mathbf{0}_{n\times 1} \\ \mathbf{0}_{1\times r} & \mathbf{0}_{1\times n} & 0 \end{bmatrix}, \qquad \mathbf{R} = \begin{bmatrix} \mathbf{0}_{(r+n+1)\times(n+2r)} & \begin{bmatrix} \mathbf{0}_{(r+n)\times 1} \\ 1 \end{bmatrix} \end{bmatrix}$$

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{T} & -\mathbf{I}_{n\times n} & \mathbf{0}_{n\times 1} \end{bmatrix}, \qquad \mathbf{C}_1 = \begin{bmatrix} \mathbf{I}_{n\times n} & \mathbf{0}_{n\times(2r+1)} \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} -\mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \mathbf{0}_{r\times 1} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \\ \mathbf{I}_{r\times r} & \mathbf{0}_{r\times n} & -\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \end{bmatrix}, \qquad \mathbf{C}_2 = \begin{bmatrix} \mathbf{0}_{r\times n} & -\mathbf{I}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{I}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \\ \mathbf{0}_{r\times n} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times r} & \mathbf{0}_{r\times 1} \end{bmatrix}$$

Since $\mathbf{W} > 0$ and $\mathbf{Q} > 0$ this is a convex quadratic program in $\mathbf{z}$ parametrized by $\mathbf{p}$. Convexity guarantees that a global solution can be found. The optimal solution $\mathbf{z}^*(\mathbf{p})$ to this problem is a continuous piecewise linear function $\mathbf{z}^*(\mathbf{p})$ defined on any subset:

$$\mathbf{p}_{\min} \leq \mathbf{p} \leq \mathbf{p}_{\max} \tag{7.434}$$

of the parameter space. Moreover, an exact representation of this piecewise linear function can be computed off-line using multi-parametric QP algorithms (Tøndel *et al.* 2001). Consequently, it is not necessary to solve the QP (7.432) in real time for the current value of τ, and the parameters $f_{\min}$, $f_{\max}$, and β, if they are allowed to vary. In fact it suffices to evaluate the known piecewise linear function $\mathbf{z}^*(\mathbf{p})$ as a function of the given parameter vector $\mathbf{p}$ which can be done efficient with a small amount of computations. For details on the implementation aspects of the mp-QP algorithm; see Johansen *et al.* (2002) and references therein.

Explicit Solutions based on Minimum Norm and Null-Space Methods

In flight and aerospace control systems, the problems of control allocation and saturating control have been addressed by Durham (1993, 1994a, 1994b). They also propose an explicit solution to avoid saturation referred to as the "direct method". By noticing that there are infinite combinations of admissible controls that generates control forces on the boundary of the closed subset of attainable controls, the "direct method" calculates admissible controls in the interior of the attainable forces as scaled down versions of the unique solutions for

force demands. Unfortunately it is not possible to minimize the norm of the control forces on the boundary or some other constraint since the solutions on the boundary are unique. The computational complexity of the algorithm is proportional to the square of the number of controls, which can be problematic in real-time applications.

In Bordignon and Durham (1995) the null space interaction method is used to minimize the norm of the control vector when possible, and still access the attainable forces to overcome the drawbacks of the "direct method". This method is also explicit but much more computational intensive. For instance 20 independent controls imply that up to 3.4 billon points have to be checked at each sample. In Durham (1999) a computationally simple and efficient method to obtain near-optimal solutions is described. The method is based on prior knowledge of the controls' effectiveness and limits such that precalculation of several generalized inverses can be done.

Iterative Solutions

An alternative to the explicit solution could be to use an iterative solution to solve the QP problem. The m-file function `quadprog.m` in the Matlab™ optimization toolbox can be used for computer simulations, while a stand-alone compiled QP solver must be implemented in a real-time application. The drawback with the iterative solution is that several iterations may have to be performed at each sample in order to find the optimal solution. An advantage of the iterative approach is that there is more flexibility for on-line reconfiguration, as for example a change in $\mathbf{W}$ may require that the explicit solutions are recalculated. Computational complexity is also greatly reduced by a "warm start"–i.e., the numerical solver is initialized with the solution of the optimization problem computed at the previous sample.

7.5.4 Constrained Control Allocation (Azimuthing Thrusters)

The control allocation problem for vessels equipped with azimuth thrusters is in general a *non-convex* optimization problem that is hard to solve. The primary constraint is:

$$\boldsymbol{\tau} = \mathbf{T}(\boldsymbol{\alpha})\mathbf{f} \tag{7.435}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^p$ denotes the azimuth angles. The azimuth angles must be computed at each sample together with the control inputs $\mathbf{u} \in \mathbb{R}^r$ which are subject to both amplitude and rate saturations. In addition, azimuthing thrusters can only operate in feasible sectors $\alpha_{i,\min} \leq \alpha_i \leq \alpha_{i,\max}$ at a limiting turning rate $\dot{\boldsymbol{\alpha}}$. Another problem is that the inverse:

$$\mathbf{T}_w^{\dagger}(\boldsymbol{\alpha}) = \mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})[\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})]^{-1} \tag{7.436}$$

can be singular for certain $\boldsymbol{\alpha}$-values. The consequence of such a singularity is that no force is produced in certain directions. This may greatly reduce dynamic performance and maneuverability as the azimuth angles can be changed slowly only. This suggests that the following criterion should be minimized (Johansen *et al.* 2003):

$$J = \min_{\mathbf{f},\boldsymbol{\alpha},\mathbf{s}} \left\{ \sum_{i=1}^{r} \bar{P}_i |f_i|^{3/2} + \mathbf{s}^\top \mathbf{Q}\mathbf{s} + (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^\top \boldsymbol{\Omega} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + \frac{\varrho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^\top(\boldsymbol{\alpha}))} \right\} \tag{7.437}$$

subject to:

$$\begin{aligned} &\mathbf{T}(\boldsymbol{\alpha})\mathbf{f} = \boldsymbol{\tau} + \mathbf{s} \\ &\mathbf{f}_{\min} \leq \mathbf{f} \leq \mathbf{f}_{\max} \\ &\boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}_{\max} \\ &\Delta\boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} - \boldsymbol{\alpha}_0 \leq \Delta\boldsymbol{\alpha}_{\max} \end{aligned}$$

where:

- $\sum_{i=1}^{r} \bar{P}_i |f_i|^{3/2}$ represents power consumption where $\bar{P}_i > 0$ $(i = 1, ..., r)$ are positive weights.
- $\mathbf{s}^\top \mathbf{Q}\mathbf{s}$ penalizes the error $\mathbf{s}$ between the commanded and achieved generalized force. This is necessary in order to guarantee that the optimization problem has a feasible solution for any $\boldsymbol{\tau}$ and $\boldsymbol{\alpha}_0$. The weight $\mathbf{Q} > 0$ is chosen so large that the optimal solution is $\mathbf{s} \approx 0$ whenever possible.
- $\mathbf{f}_{\min} \leq \mathbf{f} \leq \mathbf{f}_{\max}$ is used to limit the use of force (saturation handling).
- $\boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}_{\max}$ denotes the feasible sectors of the azimuth angles.
- $\Delta\boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} - \boldsymbol{\alpha}_0 \leq \Delta\boldsymbol{\alpha}_{\max}$ ensures that the azimuth angles do not move to much within one sample taking $\boldsymbol{\alpha}_0$ equal to the angles at the previous sample. This is equivalent to limiting $\dot{\boldsymbol{\alpha}}$–i.e. the turning rate of the thrusters.
- The term:

$$\frac{\varrho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^\top(\boldsymbol{\alpha}))}$$

  is introduced to avoid singular configurations given by $\det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^\top(\boldsymbol{\alpha}) = 0$. To avoid division by zero, $\varepsilon > 0$ is chosen as a small number, while $\varrho > 0$ is scalar weight. A large ϱ ensures high maneuverability at the cost of higher power consumption and vice versa.

The optimization problem (7.437) is a non-convex nonlinear program and it requires a significant amount of computations at each sample (Nocedal and Wright 1999). Consequently, the following two implementation strategies are attractive alternatives to nonlinear program efforts.

Iterative Solutions using Quadratic Programming

The problem (7.437) can be locally approximated with a *convex* QP problem by assuming that:

1. the power consumption can be approximated by a quadratic term in $\mathbf{f}$, near the last force $\mathbf{f}_0$ such that $\mathbf{f} = \mathbf{f}_0 + \Delta\mathbf{f}$.
2. the singularity avoidance penalty can be approximated by a linear term linearized about the last azimuth angle $\boldsymbol{\alpha}_0$ such that $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \Delta\boldsymbol{\alpha}$.

The resulting QP criterion is (Johansen *et al.* 2003):

$$J = \min_{\Delta\mathbf{f},\Delta\boldsymbol{\alpha},\mathbf{s}} \Big\{ (\mathbf{f}_0 + \Delta\mathbf{f})^\top \mathbf{P}(\mathbf{f}_0 + \Delta\mathbf{f}) + \mathbf{s}^\top \mathbf{Q}\mathbf{s} + \Delta\boldsymbol{\alpha}^\top \boldsymbol{\Omega}\Delta\boldsymbol{\alpha} + \frac{\partial}{\partial\boldsymbol{\alpha}} \left(\frac{\varrho}{\varepsilon + \det(\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^\top(\boldsymbol{\alpha}))} \right) \Big|_{\boldsymbol{\alpha}_0} \Delta\boldsymbol{\alpha} \Big\} \tag{7.438}$$

subject to:

$$\begin{aligned} &\mathbf{s} + \mathbf{T}(\boldsymbol{\alpha}_0)\Delta\mathbf{f} + \tfrac{\partial}{\partial\boldsymbol{\alpha}} (\mathbf{T}(\boldsymbol{\alpha}_0)\mathbf{f})|_{\boldsymbol{\alpha}_0,\mathbf{f}_0} \Delta\boldsymbol{\alpha} = \boldsymbol{\tau} - \mathbf{T}(\boldsymbol{\alpha}_0)\mathbf{f}_0 \\ &\mathbf{f}_{\min} - \mathbf{f}_0 \le \mathbf{f} \le \mathbf{f}_{\max} - \mathbf{f}_0 \\ &\boldsymbol{\alpha}_{\min} - \boldsymbol{\alpha}_0 \le \Delta\boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max} - \boldsymbol{\alpha}_0 \\ &\Delta\boldsymbol{\alpha}_{\min} \le \Delta\boldsymbol{\alpha} \le \Delta\boldsymbol{\alpha}_{\max} \end{aligned}$$

The convex QP problem (7.438) can be solved by using standard software for numerical optimization for instance the m-file function `quadprog.m` in the Matlab™ optimization toolbox.

Iterative Solutions using Linear Programming

Linear approximations to the thrust allocation problem has been discussed by Webster and Sousa (1999) and Lindfors (1993) . In Lindfors (1993) the azimuth thrust constraints:

$$|f_i| = \sqrt{(f_i \cos\alpha_i)^2 + (f_i \sin\alpha_i)^2} \le f_i^{\max} \tag{7.439}$$

are represented as circles in the $(f_i \cos\alpha_i, f_i \sin\alpha_i)$-plane. The nonlinear program is transformed to a linear programming (LP) problem by approximating the azimuth thrust constraints by straight lines forming a polygon. If 8 lines are used to approximate the circles (octagons), the worst case errors will be less than ± 4.0 %. The criterion to be minimized is a linear combination of $|\mathbf{f}|$, that is magnitude of force in the x- and y-directions, weighted against the magnitudes $|\sqrt{(f_i \cos\alpha_i)^2 + (f_i \sin\alpha_i)^2}|$ representing azimuth thrust. Hence, singularities and azimuth rate limitations are not weighted in the cost function. If these are important, the QP formulation should be used.

Explicit Solution using the Singular Value Decomposition and Filtering Techniques

An alternative method to solve the constrained control allocation problem is to use the singular value decomposition (SVD) and a filtering scheme to control the azimuth directions such that they are aligned with the direction where most force is required, paying attention to singularities (Sørdalen 1997*b*). Results from sea trials have been presented in Sørdalen (1997*a*). A similar technique using the damped-least squares algorithm has been reported in Berge and Fossen (1997) where the results are documented by controlling a scale model of a supply vessel equipped with four azimuth thrusters.

7.6 Exercises

Exercise 7.1 *Consider a marine craft in 6 DOF where:*

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}$$

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

Here $\boldsymbol{\nu} = [u, v, w, p, q, r]^\top$ *and* $\boldsymbol{\eta} = [n, e, d, \phi, \theta, \psi]^\top$. *Let* $V(\boldsymbol{\eta}, \boldsymbol{\nu})$ *be a Lyapunov function:*

$$V(\boldsymbol{\eta}, \boldsymbol{\nu}) = \underbrace{\frac{1}{2}\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu}}_{\textit{kinetic energy}} + \underbrace{\frac{1}{2}\boldsymbol{\eta}^\top \mathbf{K}_p\boldsymbol{\eta}}_{\textit{potential energy}}$$

where $\mathbf{M}^\top = \mathbf{M} > 0$ *and* $\mathbf{K}_p^\top = \mathbf{K}_p > 0$.

a) Show that:

$$\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = \boldsymbol{\nu}^\top \left[\boldsymbol{\tau} - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{g}(\boldsymbol{\eta}) + \mathbf{J}^\top(\boldsymbol{\eta})\mathbf{K}_p\boldsymbol{\eta}\right]$$

b) Assume that $\boldsymbol{\nu}^T\mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} > 0$ *and find a control law* $\boldsymbol{\tau}$ *such that:*

$$\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = -\boldsymbol{\nu}^\top[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d]\boldsymbol{\nu} \leq 0, \quad \mathbf{K}_d > 0$$

c) The assumption $\mathbf{K}_p^\top = \mathbf{K}_p > 0$ *is relaxed to* $\mathbf{K}_p > 0$ *– i.e.* $\mathbf{x}^\top\mathbf{K}_p\mathbf{x} > 0$, $\mathbf{x} \neq \mathbf{0}$. *Is it possible to find a control* $\boldsymbol{\tau}$ *for this case satisfying* $\dot{V}(\boldsymbol{\eta}, \boldsymbol{\nu}) = -\boldsymbol{\nu}^\top[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d]\boldsymbol{\nu} \leq 0$.
Hint: $\mathbf{x}^\top\mathbf{A}\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\mathbf{x} + \frac{1}{2}\mathbf{x}^\top(\mathbf{A} - \mathbf{A}^\top)\mathbf{x} = \frac{1}{2}\mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$ *since* $\mathbf{x}^\top(\mathbf{A} - \mathbf{A}^\top)\mathbf{x} = 0$.

Exercise 7.2 *Consider the attitude dynamics of a satellite (*Euler's equations*):*

$$\dot{\boldsymbol{\Theta}} = \mathbf{T}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\omega}_{nb}^b$$

$$\mathbf{I}_o\dot{\boldsymbol{\omega}}_{nb}^b + \boldsymbol{\omega}_{nb}^b \times (\mathbf{I}_o\boldsymbol{\omega}_{nb}^b) = \mathbf{m}_o^b$$

where $\mathbf{m}_o^b$ *is the control input vector. Next let* $V(\boldsymbol{\omega}_{nb}^b, \boldsymbol{\Theta})$ *be a CLF:*

$$V(\boldsymbol{\omega}_{nb}^b, \boldsymbol{\Theta}) = \frac{1}{2}(\boldsymbol{\omega}_{nb}^b)^\top \mathbf{I}_o\boldsymbol{\omega}_{nb}^b + \frac{1}{2}\boldsymbol{\Theta}^\top\mathbf{K}_p\boldsymbol{\Theta}$$

where $\mathbf{I}_o^\top = \mathbf{I}_o > 0$ *and* $\mathbf{K}_p^\top = \mathbf{K}_p > 0$.

a) Show that:

$$\dot{V}(\boldsymbol{\omega}_{nb}^b, \boldsymbol{\Theta}) = (\boldsymbol{\omega}_{nb}^b)^\top \left[\mathbf{m}_o^b + \mathbf{S}(\mathbf{I}_o\boldsymbol{\omega}_{nb}^b)\boldsymbol{\omega}_{nb}^b + \mathbf{T}_{\boldsymbol{\Theta}}^\top(\boldsymbol{\Theta})\mathbf{K}_p\boldsymbol{\Theta}\right]$$

Hint: $\boldsymbol{\omega}_{nb}^b \times (\mathbf{I}_o\boldsymbol{\omega}_{nb}^b) = -(\mathbf{I}_o\boldsymbol{\omega}_{nb}^b) \times \boldsymbol{\omega}_{nb}^b = -\mathbf{S}(\mathbf{I}_o\boldsymbol{\omega}_{nb}^b)\boldsymbol{\omega}_{nb}^b$

b) Use the fact that $\mathbf{x}^T\mathbf{S}\mathbf{x} = 0, \forall \mathbf{x} \in \mathbb{R}^3$ *if* $\mathbf{S} = -\mathbf{S}^T$. *Find a moment control law* $\mathbf{m}_o^b$ *such that:*

$$\dot{V}(\boldsymbol{\omega}_{nb}^b, \boldsymbol{\Theta}) = -(\boldsymbol{\omega}_{nb}^b)^\top\mathbf{K}_d\boldsymbol{\omega}_{nb}^b \leq 0, \quad \mathbf{K}_d > 0$$

Is this a P, PD or PID control law? (Explain why).

c) Explain why this solution is local *alternatively* global.

Exercise 7.3 *Consider a* torpedo *where the forward speed u is given by:*

$$(m - X_{\dot{u}})\dot{u} - X_{|u|u}\,|u|\,u = \tau$$

and where τ represents the control input. Assume that the reference speed u_d and acceleration $\dot{u}_d$ are smooth known signals. We will now construct a torpedo speed tracking controller using Lyapunov analysis.

a)Assume that m and $-X_{\dot{u}}$ are positive constants. Show that the nonlinear speed controller:

$$\tau = -\underbrace{K_m\tilde{u}}_{\substack{\text{acceleration}\\ \text{feedback}}} \underbrace{+(m - X_{\dot{u}})\dot{u}_d - X_{|u|u}\,|u|\,u_d)}_{\substack{\text{reference}\\ \text{feedforward}}} - \underbrace{K_d\tilde{u}}_{\text{P-controller}}$$

with $K_m > 0$ and $K_d > 0$ results in a GES error system. Use the following Lyapunov function:

$$V(\tilde{u}) = \frac{1}{2}(m - X_{\dot{u}} + K_m)\tilde{u}^2$$

where $\tilde{u} = u - u_d$ in your analysis.

b) Assume that the mass m of the torpedo is reduced during speed control due to consumption of fuel while hydrodynamic added mass $X_{\dot{u}}$ is constant. Let:

$$m(t) = m_o + \Delta m(t)$$

where m_o =constant is the mass of the torpedo without fuel and $\Delta m(t)$ denotes the mass of the fuel. The fuel consumption dynamics can be described a first-order system:

$$T\Delta\dot{m}(t) + \Delta m(t) = 0, \quad T > 0$$

where $\Delta m(0) = \Delta m_o$ =constant is the initial mass of the fuel and $\lim_{t\to\infty} \Delta m(t) = 0$. Let the acceleration feedback gain $K_m(t)$ be a function of time and use the speed controller and Lyapunov function in 1a) to find a differential equation for $K_m(t)$ with initial condition $K_m(0) = 0$ such that:

$$\dot{V}(\tilde{u}) = -(K_d - X_{|u|u}\,|u|)\tilde{u}^2 < 0, \quad \forall\tilde{u} \neq 0, u \in \mathbb{R}$$

c) Find exact solutions for $K_m(t)$ and $\Delta m(t)$ by solving the ODEs and sketch the solutions as functions of time (in the same plot). Comment the results.

Exercise 7.4 *Consider the model:*

$$\begin{aligned} \dot{x} &= v \\ m\dot{v} + dv + kx &= \tau + w \\ y &= x + v_m \end{aligned}$$

where w and v_m are zero mean white noise signals with variance $q = E(w^2)$ and $r = E(v_m^2)$.

a) Under what conditions are this model observable? The conditions should only depend on

the model parameters that is m, d and k.

b) Construct a continious-time Kalman filter:

$$\frac{d}{dt}\hat{\mathbf{x}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{b}u + \mathbf{k}(y - \mathbf{c}^T\hat{\mathbf{x}})$$

and compute the steady-state Kalman filter gains $\mathbf{k} = [k_1, k_2]^T$ as a function of the model parameters (m, d, k) and noise variances (q, r). (Hint: solve the steady-state Riccati equation for $\mathbf{P}$ by setting $\dot{\mathbf{P}} = \mathbf{0}$).

c) Explain why $\hat{\mathbf{x}} \to \mathbf{x}$ when using the estimator 2b). (Hint: analyze the estimation error dynamics)

Exercise 7.5 *Consider the ship autopilot model:*

$$\begin{aligned} \dot{\psi} &= r \\ T\dot{r} + H(r) &= K\delta \\ \dot{\delta} &= f(\delta) + \delta_c \end{aligned}$$

where ψ is the yaw angle, r is the yaw rate and δ is the rudder angle. The input to the steering machine is δ_c while $H(r)$ and $f(\delta)$ are two nonlinear functions describing the nonlinear maneuvering characteristic and steering dynamics, respectively.

a) Construct a backstepping controller with nonlinear damping for tracking of a smooth reference signal given by ψ_d, r_d and $\dot{r}_d$ using δ_c as input. The solution should be presented in three recursive steps using Lyapunov function analysis.

b) Write the error dynamics in the standard form:

$$\mathbf{M}\dot{\mathbf{z}} = -\mathbf{K}(\mathbf{z})\mathbf{z} + \mathbf{S}\mathbf{z}$$

where $\mathbf{S} = -\mathbf{S}^T$. The expressions for $\mathbf{M}, \mathbf{K}(\mathbf{z})$ and $\mathbf{S}$ should be derived.

Part III

Ship and Rig Applications

Chapter 8

Course Autopilots

This chapter discusses model-based autopilot design, including PID-control, LQ design techniques, feedback linearization, and backstepping designs. Both full state and output feedback are discussed. The main motivation for using model-based control and not neural networks, fuzzy systems, genetic algorithms etc. is that the course dynamics of the ship is quite easy to model with good accuracy. Consequently, it is advantageous to exploit the ship dynamics when designing the autopilot. Experimental results with both model ships and full-scale vessels are used in the case studies to illustrate the different design techniques.

The principal blocks of a heading angle autopilot system, shown in Figure 8.1, are:

Control System: The control system provides the necessary feedback signal to track the desired yaw angle ψ_d. The output is the yaw moment τ_N.

Control Allocation: This module distributes the output from the feedback control system, usually the yaw moment τ_N, to the actuators (rudders and in some cases propellers, thrusters, etc.) in an optimal manner; see Section 7.5. For single screw ships the controller yaw moment τ_N will simply be a function of the rudder command δ_c.

Reference Model: The autopilot reference model computes smooth trajectories ψ_d, r_d, and $\dot{r}_d$ needed for *course-changing* maneuvers. *Course-keeping* is the special case then ψ_d = constant and $r_d = \dot{r}_d = 0$; see Section 5.1

Compass/Yaw Gyro: The compass measures the yaw angle ψ which is needed for feedback. In some cases a yaw gyro is available for yaw rate feedback, that is feedback from $r = \dot{\psi}$; see Sections 6.1 and 6.2.

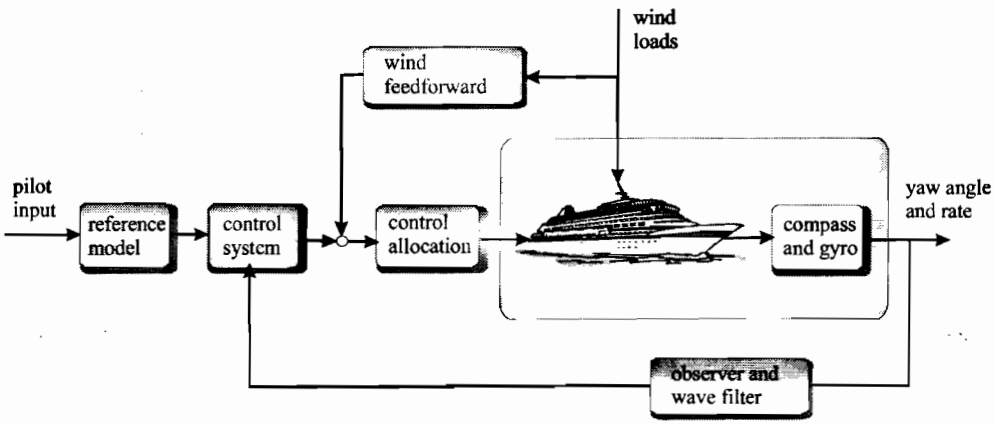


Figure 8.1: Block diagram of autopilot system.

Observer/Wave Filter: In its simplest form the 1st-order wave-induced motion components ψ_w and r_w are filtered out from the measurements $y_1 = \psi + \psi_w$ and $y_2 = r + r_w$, and consequently prevented from entering the feedback loop. This is known as *wave filtering* where the output of the filter is the low-frequency motion components ψ and r. This is necessary to avoid excessive rudder action. In cases where y_2 is not measured the wave filter must be constructed as a state observer so that r can be estimated from the yaw angle measurement y_1; see Sections 6.1.3 and 6.1.4

Wind Feedforward: In cases where a wind sensor is available for *wind speed* and *direction*, a wind model can be used for wind feedforward. This is often advantageous since the integral action term in the PID-controller does not have to integrate up the wind disturbance term. However, an accurate model of the wind force and moment as a function of ship speed and wind direction is needed to implement wind feedforward; see Section 4.1.

8.1 Autopilot Models

Before discussing autopilot design the most important autopilot models from the literature are reviewed.

8.1.1 Rigid-Body Ship Dynamics

In Section 3.5.1 the 3 DOF model (surge, sway, and yaw) was written:

$$\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_{RB} \tag{8.1}$$

where $\boldsymbol{\tau}_{RB} = [X, Y, N]^\top$ and:

$$\mathbf{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix}, \mathbf{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \tag{8.2}$$

This is based on the assumptions that:

- the body-fixed coordinate origin is set in the center line of the vessel, that is $y_g = 0$.
- the vessel has a homogeneous mass distribution and the hull is symmetrical about the xz-plane such that $I_{xy} = I_{yz} = 0$; see Section 3.4.2.
- the heave, roll, and pitch modes can be neglected for normal operations, that is, $w = p = q = \dot{w} = \dot{p} = \dot{q} = 0$.

Nonlinear Rigid-Body Equations of Motion

Expanding (8.1), yields the horizontal plane model:

$$\begin{array}{lrcl} \text{Surge:} & m(\dot{u} - vr - x_g r^2) & = & X \\ \text{Sway:} & m(\dot{v} + ur + x_g \dot{r}) & = & Y \\ \text{Yaw:} & I_z \dot{r} + m x_g (\dot{v} + ur) & = & N \end{array} \tag{8.3}$$

Linear Rigid-Body Equations of Motion

Linear theory is based on the assumption that the mean forward speed $U \approx u_0$ is constant (or at least slowly-varying) and that $v \approx 0$ and $r \approx 0$, such that:

$$U = \sqrt{u^2 + v^2} \approx u_0 \tag{8.4}$$

Linearization of (8.1) about $(u, v, r) = (u_0, 0, 0)$, yields:

$$\begin{array}{lrcl} \text{Speed equation:} & m\Delta\dot{u} & = & \Delta X \\ \text{Steering equations:} & m(\Delta\dot{v} + u_0 \Delta r + x_g \Delta\dot{r}) & = & \Delta Y \\ & I_z \Delta\dot{r} + m x_g (\Delta\dot{v} + u_0 \Delta r) & = & \Delta N \end{array} \tag{8.5}$$

where $\Delta(\cdot)$ denotes the perturbation away from the nominal value. Notice that linearization implies that the speed equation (*surge equation*) is decoupled from the steering equations (*sway-yaw subsystem*). The assumption that the mean forward speed is constant implies that this model is only valid for small rudder angles.

Hydrodynamic Control Forces and Moments

For ships with one single-screw propeller, thrust and rudder deflection are denoted by T and δ, respectively. The force and moment terms X, Y, and N are nonlinear functions of (δ, T);

the velocities (u, v, r); and the accelerations $(\dot{u}, \dot{v}, \dot{r})$, that is

$$\begin{aligned} X &= X(u, v, r, \dot{u}, \delta, T) \\ Y &= Y(v, r, \dot{v}, \dot{r}, \delta) \\ N &= N(v, r, \dot{v}, \dot{r}, \delta) \end{aligned} \tag{8.6}$$

Decoupling the Speed Equation from the Steering Equations

The speed equation relates the propeller thrust T to the velocity u and it is obtained by combining (8.3) and (8.6) as:

$$m(\dot{u} - vr - x_g r^2) = X(u, v, r, \dot{u}, \delta, T) \tag{8.7}$$

Here X is a nonlinear function describing the hydrodynamic surge force. An example is the model by Blanke (1981):

$$X = X_{\dot{u}}\dot{u} + X_{vr}vr + X_{|u|u}|u|u + X_{rr}r^2 + (1-t)T + X_{cc\delta\delta}c^2\delta^2 + X_{\text{ext}} \tag{8.8}$$

The hydrodynamic coefficients in this expression are:

$X_{\dot{u}}$	-	added mass in surge
$X_{\|u\|u}$	-	drag force coefficient in surge (resistance)
t	-	thrust deduction number
$X_{cc\delta\delta}$	-	resistance due to rudder deflection
T	-	propeller thrust
c	-	flow velocity past the rudder
T_{loss}	-	loss term or added resistance
$(m + X_{vr})$	-	excessive drag force due to combined sway-yaw motions
$(X_{rr} + mx_g)$	-	excessive drag force in yaw
X_{ext}	-	external force due to wind and waves

Substituting (8.8) into (8.7), yields:

$$(m - X_{\dot{u}})\dot{u} = X_{|u|u}|u|u + (1-t)T + T_{\text{loss}} \tag{8.9}$$

where:

$$T_{\text{loss}} = (m + X_{vr})vr + \underbrace{X_{cc\delta\delta}c^2}_{X_{\delta\delta}}\delta^2 + (X_{rr} + mx_g)r^2 + X_{\text{ext}} \tag{8.10}$$

It should be noted that the resistance and the propeller thrust will outbalance each other in steady state if the loss term $T_{\text{loss}} = 0$, that is:

$$\dot{u} = 0 \Rightarrow -X_{|u|u}|u|u = (1-t)T \tag{8.11}$$

The flow past the rudder is of course strongly influenced by the propeller-induced flow. A theoretical framework showing this relationship is included in Blanke (1981). This is based on the experiments of Van Berlekom (1975). In many cases it is assumed that the effect of the flow velocity c can be neglected such that:

$$X_{cc\delta\delta}c^2 = X_{\delta\delta} = \text{constant} \tag{8.12}$$

Linearization of (8.9) about $u = u_0$, yields:

$$(m - X_{\dot{u}})\Delta\dot{u} = X_u \Delta u + (1-t)\Delta T + \Delta T_{\text{loss}} \tag{8.13}$$

where $X_u = 2u_0 X_{u|u|}$ is the linear damping derivative in surge. The cruise speed u_o as a function of steady thrust T_o for $(T_{\text{loss}})_0 = 0$ is given by (8.11). Consequently:

$$u_0 = \text{sgn}(T_0)\sqrt{\frac{(1-t)\,|T_0|}{-X_{|u|u}}} \tag{8.14}$$

8.1.2 The Linear Ship Steering Equations

The ship steering equations of motion are obtained by considering the sway-yaw subsystem corresponding to the state variables v, r, ψ, and the control input δ.

The Model of Davidson and Schiff (1946)

Consider the linear steering dynamics (8.5) in the form:

$$\mathbf{M}_{RB}\dot{\boldsymbol{\nu}} + \mathbf{C}_{RB}(u_0)\boldsymbol{\nu} = \boldsymbol{\tau}_{RB} \tag{8.15}$$

where $\boldsymbol{\nu} = [v, r]^\top$ is the state vector and:

$$\mathbf{M}_{RB} = \begin{bmatrix} m & mx_g \\ mx_g & I_z \end{bmatrix}, \quad \mathbf{C}_{RB}(u_0) = \begin{bmatrix} 0 & mu_0 \\ 0 & mx_g u_0 \end{bmatrix} \tag{8.16}$$

Notice that $\mathbf{C}_{RB}(u_0)$ is not skew-symmetric for the reduced order model (8.15) with constant forward speed u_0. Linear theory suggests that the hydrodynamic force and moment can be modeled as (Davidson and Schiff 1946):

$$\boldsymbol{\tau}_{RB} = -\mathbf{M}_A\dot{\boldsymbol{\nu}} - \mathbf{D}\boldsymbol{\nu} - \mathbf{b}\delta_R \tag{8.17}$$

where δ_R is the rudder angle, $\mathbf{b} = [-Y_\delta, -N_\delta]^\top$ and:

$$\mathbf{M}_A = \begin{bmatrix} -Y_{\dot{v}} & -Y_{\dot{r}} \\ -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} -Y_v & -Y_r \\ -N_v & -N_r \end{bmatrix} \tag{8.18}$$

Notice that Davidson and Schiff (1946) assume linear damping $\mathbf{D}\boldsymbol{\nu}$ and that the hydrodynamic added mass $\mathbf{C}_A(\boldsymbol{\nu})\boldsymbol{\nu}$ (quadratic terms in velocity) can be neglected.

For notational convenience, it is common to define:

$$\delta = -\delta_R \tag{8.19}$$

such that a positive rudder angle δ results in a positive yaw rate r. The resulting model then becomes:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_0)\boldsymbol{\nu} = \mathbf{b}\delta \tag{8.20}$$

where:

$$\mathbf{M} = \begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} -Y_\delta \\ -N_\delta \end{bmatrix} \tag{8.21}$$

$$\mathbf{N}(u_0) = \mathbf{C}(u_0) + \mathbf{D} = \begin{bmatrix} -Y_v & mu_0 - Y_r \\ -N_v & mx_g u_0 - N_r \end{bmatrix} \tag{8.22}$$

For positive speed $u_0 \gg 0$, the system inertia matrix $\mathbf{M} \neq \mathbf{M}^\top$ since $Y_{\dot{r}} \neq N_{\dot{v}}$.

State-Space Modeling

The corresponding state-space model is obtained by letting $\mathbf{x} = [v, r]^\top$ be the state vector and $u = \delta$. Consequently:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}_1 u \tag{8.23}$$

where:

$$\mathbf{A} = -\mathbf{M}^{-1}\mathbf{N} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \qquad \mathbf{b}_1 = \mathbf{M}^{-1}\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{8.24}$$

The coefficients are obtained from:

$$\begin{aligned}
a_{11} &= \frac{-(I_z - N_{\dot{r}})Y_v + (mx_g - Y_{\dot{r}})N_v}{\det(\mathbf{M})} \\
a_{12} &= \frac{(I_z - N_{\dot{r}})(mu_0 - Y_r) - (mx_g - Y_{\dot{r}})(mx_g u_0 - N_{\dot{r}})}{\det(\mathbf{M})} \\
a_{21} &= \frac{-(m - Y_{\dot{v}})N_v + (mx_g - N_{\dot{v}})Y_v}{\det(\mathbf{M})} \\
a_{22} &= \frac{(m - Y_{\dot{v}})(mx_g u_0 - N_r) - (mx_g - N_{\dot{v}})(mu_0 - Y_r)}{\det(\mathbf{M})} \\
b_1 &= \frac{-(I_z - N_{\dot{r}})Y_\delta + (mx_g - Y_{\dot{r}})N_\delta}{\det(\mathbf{M})} \\
b_2 &= \frac{-(m - Y_{\dot{v}})N_\delta + (mx_g - N_{\dot{v}})Y_\delta}{\det(\mathbf{M})}
\end{aligned} \tag{8.25}$$

where $\det(\mathbf{M}) = (m - Y_{\dot{v}})(I_z - N_{\dot{r}}) - (mx_g - N_{\dot{v}})(mx_g - Y_{\dot{r}}) > 0$ is the determinant of the system inertia matrix.

The Models of Nomoto (1957)

An alternative representation of the model of Davidson and Schiff (1946) was proposed by Nomoto *et al.* (1957). This model is obtained by eliminating the sway velocity v from (8.20). The result is Nomoto's 2nd-order model which simply is the transfer function between r and δ, that is:

$$\frac{r}{\delta}(s) = \frac{K(1 + T_3 s)}{(1 + T_1 s)(1 + T_2 s)} \tag{8.26}$$

where T_i $(i = 1, \ldots, 3)$ are time constants and K is the gain constant.

Matlab:
This Nomoto transfer function is computed numerically in Matlab™ from the state-space model (8.23) by:

```
A      = -inv(M)*N
b1     = inv(M)*b
[n,d]  = ss2tf(A,b1)
```

A similar expression is obtained for sway:

$$\frac{v}{\delta}(s) = \frac{K_v(1+T_v s)}{(1+T_1 s)(1+T_2 s)} \tag{8.27}$$

where K_v and T_v are the gain and time constants in sway, respectively.

A 1st-order approximation to (8.26) is obtained by defining the *effective time constant*:

$$T = T_1 + T_2 - T_3 \tag{8.28}$$

such that:

$$\frac{r}{\delta}(s) = \frac{K}{(1+Ts)} \tag{8.29}$$

where T and K are known as the Nomoto time and gain constants, respectively. Neglecting the roll and pitch modes ($\phi = \theta = 0$) such that:

$$\dot{\psi} = r \tag{8.30}$$

finally yields:

$$\begin{aligned} \frac{\psi}{\delta}(s) &= \frac{K(1+T_3 s)}{s(1+T_1 s)(1+T_2 s)} \\ &\approx \frac{K}{s(1+Ts)} \end{aligned} \tag{8.31}$$

This model is the most popular model for ship autopilot design due to its simplicity and accuracy.

Time-Domain Representations of the 1st- and 2nd-Order Nomoto Models

Combining Nomoto's 2nd-order model (8.26) with (8.30), yields:

$$T_1 T_2 \psi^{(3)} + (T_1 + T_2)\ddot{\psi} + \dot{\psi} = K(\delta + T_3 \dot{\delta}) \tag{8.32}$$

A similar expression is obtained for (8.29) and (8.30):

$$T\ddot{\psi} + \dot{\psi} = K\delta \tag{8.33}$$

Matlab:
The accuracy of the 1st-order Nomoto model when compared to the 2nd-order model is illustrated in Example 8.1 where a course stable cargo ship and a course unstable oil tanker are considered.

```
function nomoto(T1,T2,T3,K)
% NOMOTO(T1,T2,T3,K) generates the Bode plots for
%
%            K                      K (1+T3s)
% H1(s) = --------     H2(s) = -------------
%         (1+Ts)s                s(1+T1s)(1+T2s)
%
% Author:      Thor I. Fossen
% Date:        19th June 2001
% Revisions:

T = T1+T2-T3;
d1 = [T 1 0];                n1 = K;
d2 = [T1*T2 T1+T2 1 0];      n2 = K*[T3 1];
[mag1,phase1,w] = bode(n1,d1);
[mag2,phase2]     = bode(n2,d2,w);

% shift ship phase with 360 deg for course unstable ship
if K < 0,
phase1 = phase1-360;
phase2 = phase2-360;
end

clf,subplot(211),semilogx(w,20*log10(mag1)),grid
xlabel('Frequency [rad/s]'),title('Gain [dB]')
hold on,semilogx(w,20*log10(mag2),'-'),hold off
subplot(212),semilogx(w,phase1),grid
xlabel('Frequency [rad/s]'),title('Phase [deg]')
hold on,semilogx(w,phase2,'-'),hold off
```

Example 8.1 (Frequency Response for Nomoto 1st- and 2nd-Order Models)
Consider a Mariner class cargo ship (Chislett and Strøm-Tejsen 1965a) and a fully loaded tanker (Dyne and Trägårdh 1975) given by the parameters in Table 8.1. The Bode diagram is generated by using the GNC Toolboc commands:

```
T1=118;T2=7.8;T3=18.5;K=0.185;
nomoto(T1,T2,T3,K)

T1=-124.1;T2=16.4;T3=46.0;K=-0.019;
nomoto(T1,T2,T3,K)
```

It is seen from Figure 8.2 that the 1st-order approximation is quite accurate up to 0.1 rad/s for the cargo ship and the tanker. A small deviation in the phase around 0.5 rad/s is observed. This is due to the cancellation of the sway dynamics.

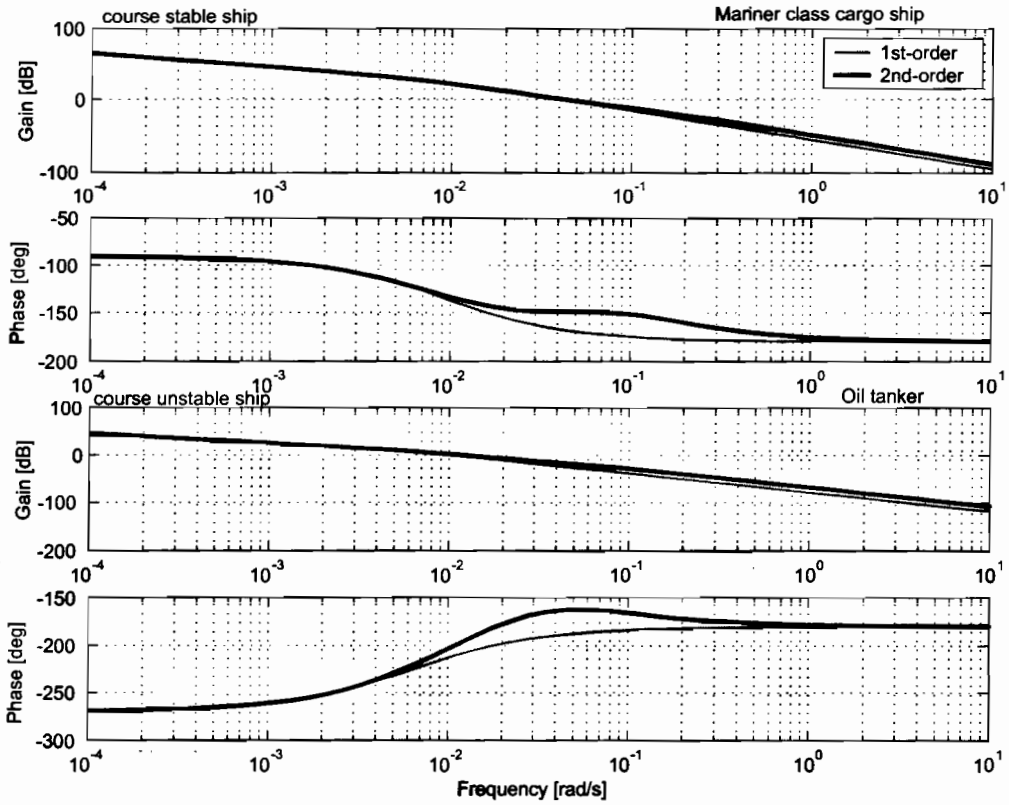


Figure 8.2: 1st-order and 2nd-order Nomoto transfer functions for a course stable Mariner class cargo ship and a course unstable oil tanker.

Table 8.1: Parameters for a cargo ship and a fully loaded oil tanker.

	L (m)	u_0 (m/s)	∇ (dwt)	K (1/s)	T_1 (s)	T_2 (s)	T_3 (s)
cargo ship	161	7.7	16622	0.185	118.0	7.8	18.5
Oil tanker	350	8.1	389100	−0.019	−124.1	16.4	46.0

8.1.3 Non-Dimensional Autopilot Models

When designing course autopilots it is often convenient to normalize the ship steering equations of motion such that the model parameters can be treated as constants with respect to the instantaneous speed U defined by:

$$U = \sqrt{u^2 + v^2} = \sqrt{(u_0 + \Delta u)^2 + \Delta v^2} \tag{8.34}$$

where u_0 is the *service speed* and Δu and Δv are small perturbations in the surge and sway velocities, respectively. Hence,

$$U \approx u_0 \tag{8.35}$$

During course changing maneuvers the instantaneous speed will decrease due to increased resistance during the turn.

The most commonly used normalization forms for marine vessels are the *Prime-system* of SNAME (1950), and the *Bis-system* of Norrbin (1970).

Prime-System: This system uses the ship's instantaneous speed U, the length $L = L_{pp}$ (the length between the fore and aft perpendiculars), the time unit L/U, and the mass unit $12\rho L^3$ or $12\rho L^2 T$ as normalization variables. The latter is inspired by wing theory where the reference area $A = LT$ is used instead of $A = L^2$. The prime system cannot be used for low-speed applications like dynamic ship positioning, since normalization of the velocities u, v, w implies dividing by the cruise speed U, which can be zero for a dynamically positioned ships. As a consequence, the prime system is mostly used in ship maneuvering.

Bis-System: This system can be used for zero speed as well as high speed applications since division of speed U is avoided. The Bis-system is based on the use of the length $L = L_{pp}$, the time unit $\sqrt{L/g}$ such that speed becomes $\sqrt{Lg} > 0$. In addition the body mass density ratio $\mu = m/\rho\nabla$, where m is the mass unit and ∇ is the hull contour displacement, is applied. The density ratio μ takes the following values:

$\mu < 1$	Underwater vehicles (ROVs, AUVs, submarines etc.)
$\mu = 1$	Floating ships/rigs and neutrally buoyant underwater vehicles
$\mu > 1$	Heavy torpedoes (typically $\mu = 1.3$–1.5)

The normalization variables for the Prime- and Bis-systems are given in Table 8.2. The non-dimensional quantities in the Prime- and Bis-systems will be distinguished from those with dimension by applying the notation $(\cdot)'$ for the Prime-system and $(\cdot)''$ for the Bis-system.

Example 8.2 (Normailzation of Parameters)
The hydrodynamic coefficient Y_r can be made non-dimensional by using the Prime- and Bis-systems. First, let us determine the dimension of Y_r. Consider:

$$\underbrace{Y}_{[N=kgm/s^2]} = \underbrace{Y_r}_{[unknown]} \underbrace{r}_{[rad/s]}$$

Hence, the unknown dimension must be kgm/s since rad is a non-dimensional unit. The non-dimensional values Y_r' and Y_r'' are found by using kg, m, and s from Table 8.2. Consequently:

$$Y_r' = \frac{Y_r}{\frac{[\frac{1}{2}\rho L^3][L]}{[L/U]}} = \frac{1}{\frac{1}{2}\rho L^3 U} Y_r \tag{8.36}$$

$$Y_r'' = \frac{Y_r}{\frac{[\mu\rho\nabla][L]}{\sqrt{L/g}}} = \frac{1}{\mu\rho\nabla\sqrt{Lg}} Y_r \tag{8.37}$$

For a floating ship Y_r'' can be further simplified since $\mu = 1$ and $m = \rho\nabla$. Hence:

$$Y_r'' = \frac{1}{m\sqrt{Lg}} Y_r \tag{8.38}$$

Table 8.2: Normalization variables used for the Prime-system and Bis-system.

Unit	Prime-system I	Prime-system II	Bis-system
Length	L	L	L
Mass	$\frac{1}{2}\rho L^3$	$\frac{1}{2}\rho L^2 T$	$\mu\rho\nabla$
Inertia moment	$\frac{1}{2}\rho L^5$	$\frac{1}{2}\rho L^4 T$	$\mu\rho\nabla L^2$
Time	$\frac{L}{U}$	$\frac{L}{U}$	$\sqrt{L/g}$
Reference area	L^2	LT	$\mu\frac{2\nabla}{L}$
Position	L	L	L
Angle	1	1	1
Linear velocity	U	U	$\sqrt{Lg}$
Angular velocity	$\frac{U}{L}$	$\frac{U}{L}$	$\sqrt{g/L}$
Linear acceleration	$\frac{U^2}{L}$	$\frac{U^2}{L}$	g
Angular acceleration	$\frac{U^2}{L^2}$	$\frac{U^2}{L^2}$	$\frac{g}{L}$
Force	$\frac{1}{2}\rho U^2 L^2$	$\frac{1}{2}\rho U^2 LT$	$\mu\rho g\nabla$
Moment	$\frac{1}{2}\rho U^2 L^3$	$\frac{1}{2}\rho U^2 L^2 T$	$\mu\rho g\nabla L$

Example 8.3 (Normalization of States and Parameters)
Consider the model of Davidson and Schiff (1946). Normalization of (8.20) according to the Prime-system suggests:

$$\mathbf{M}'\dot{\boldsymbol{\nu}}' + \mathbf{N}'(u_0')\boldsymbol{\nu}' = \mathrm{b}'\delta' \tag{8.39}$$

where $\boldsymbol{\nu}' = [v', r']^\top$ *and:*

$$\mathbf{M}' = \begin{bmatrix} m' - Y_{\dot{v}}' & m'x_g' - Y_{\dot{r}}' \\ m'x_g' - N_{\dot{v}}' & I_z' - N_{\dot{r}}' \end{bmatrix}, \quad \mathrm{b}' = \begin{bmatrix} -Y_\delta' \\ -N_\delta' \end{bmatrix}$$

$$\mathbf{N}'(u_0') = \begin{bmatrix} -Y_v' & m'u_0' - Y_r' \\ -N_v' & m'x_g'u_0' - N_r' \end{bmatrix}$$

where

$$u_0' = \frac{u_0}{U} \approx 1, \quad \textit{for } \Delta u \approx 0 \textit{ and } \Delta v \approx 0 \tag{8.40}$$

The non-dimensional velocities and control input can be transformed to its dimensional values by:

$$v = U\,v', \qquad r = \frac{U}{L}r', \qquad \delta = \delta' \tag{8.41}$$

6 DOF Normalization Procedure

A systematic procedure for 6 DOF normalization is found by defining a transformation matrix:

$$\mathbf{T} = \mathrm{diag}\left\{1, 1, 1, \frac{1}{L}, \frac{1}{L}, \frac{1}{L}\right\} \tag{8.42}$$

$$\mathbf{T}^{-1} = \mathrm{diag}\{1, 1, 1, L, L, L\} \tag{8.43}$$

Table 8.3: 6 DOF normalization variables.

	Prime-system	**Bis-system**
acceleration	$\dot{\boldsymbol{\nu}} = \frac{U^2}{L}\,\mathbf{T}\dot{\boldsymbol{\nu}}'$	$\dot{\boldsymbol{\nu}} = g\,\mathbf{T}\dot{\boldsymbol{\nu}}''$
velocity	$\boldsymbol{\nu} = U\,\mathbf{T}\boldsymbol{\nu}'$	$\boldsymbol{\nu} = \sqrt{Lg}\,\mathbf{T}\boldsymbol{\nu}''$
position/attitude	$\boldsymbol{\eta} = L\,\mathbf{T}\boldsymbol{\eta}'$	$\boldsymbol{\eta} = L\,\mathbf{T}\boldsymbol{\eta}''$
control forces/moments	$\boldsymbol{\tau} = \frac{1}{2}\rho U^2 L^2\,\mathbf{T}^{-1}\boldsymbol{\tau}'$	$\boldsymbol{\tau} = \mu\rho g\nabla\,\mathbf{T}^{-1}\boldsymbol{\tau}''$

Consider the non-dimensional MIMO model (see Example 8.3):

$$\mathbf{M}'\dot{\boldsymbol{\nu}}' + \mathbf{D}'\boldsymbol{\nu}' + \mathbf{G}'\boldsymbol{\eta}' = \boldsymbol{\tau}' \tag{8.44}$$

When designing vessel simulators and gain-scheduled controllers it is convenient to perform the numerical integration in real-time using dimensional time. Consequently, it is convenient to use the non-dimensional hydrodynamic coefficients as input to the simulator or controller, while the states $\boldsymbol{\nu}, \boldsymbol{\eta}$, and input $\boldsymbol{\tau}$ should have their physical dimensions. For the *Prime-system* this is obtained by applying the following transformation to (8.44):

$$\mathbf{M}'\left(\frac{L}{U^2}\mathbf{T}^{-1}\dot{\boldsymbol{\nu}}\right) + \mathbf{D}'\left(\frac{1}{U}\mathbf{T}^{-1}\boldsymbol{\nu}\right) + \mathbf{G}'\left(\frac{1}{L}\mathbf{T}^{-1}\boldsymbol{\eta}\right) = \frac{1}{\frac{1}{2}\rho U^2 L^2}\mathbf{T}\boldsymbol{\tau} \tag{8.45}$$

such that:

$$(\mathbf{T}\mathbf{M}'\mathbf{T}^{-1})\dot{\boldsymbol{\nu}} + \left(\frac{U}{L}\right)(\mathbf{T}\mathbf{D}'\mathbf{T}^{-1})\boldsymbol{\nu} + \left(\frac{U}{L}\right)^2(\mathbf{T}\mathbf{G}'\mathbf{T}^{-1})\boldsymbol{\eta} = \frac{1}{\frac{1}{2}\rho L^3}\mathbf{T}^2\boldsymbol{\tau} \tag{8.46}$$

Notice that $\boldsymbol{\nu}, \boldsymbol{\eta}$, and the input vector $\boldsymbol{\tau}$ now have physical dimensions while $\mathbf{M}', \mathbf{D}'$ and $\mathbf{G}'$ are non-dimensional. Similarly, *Bis-system* scaling gives:

$$(\mathbf{T}\mathbf{M}''\mathbf{T}^{-1})\dot{\boldsymbol{\nu}} + \sqrt{\frac{g}{L}}(\mathbf{T}\mathbf{D}''\mathbf{T}^{-1})\boldsymbol{\nu} + \frac{g}{L}(\mathbf{T}\mathbf{G}''\mathbf{T}^{-1})\boldsymbol{\eta} = \frac{1}{\mu\rho\nabla}\mathbf{T}^2\boldsymbol{\tau} \tag{8.47}$$

The following example demonstrates this for the model of Davidson and Schiff (1946).

Example 8.4 (Normalization of Parameters while keeping the Actual States)
Consider the model in Example 8.3:

$$\mathbf{M}'\dot{\boldsymbol{\nu}}' + \mathbf{N}'(u_0')\boldsymbol{\nu}' = \mathbf{b}'\delta' \tag{8.48}$$

Transforming the states $\boldsymbol{\nu}'$ and control input δ' in (8.45) to dimensional quantities, yields:

$$(\mathbf{T}\mathbf{M}'\mathbf{T}^{-1})\dot{\boldsymbol{\nu}} + \frac{U}{L}(\mathbf{T}\mathbf{N}'(u_0')\mathbf{T}^{-1})\boldsymbol{\nu} = \frac{U^2}{L}\mathbf{T}\mathbf{b}'\delta \tag{8.49}$$

where

$$\mathbf{T} = diag\{1, 1, 1/L\} \tag{8.50}$$

Notice that $\delta = \delta'$. Expanding (8.49), yields:

$$\begin{bmatrix} m'_{11} & Lm'_{12} \\ \frac{1}{L}m'_{21} & m'_{22} \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} + \frac{U}{L} \begin{bmatrix} n'_{11} & Ln'_{12} \\ \frac{1}{L}n'_{21} & n'_{22} \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} = \frac{U^2}{L} \begin{bmatrix} b'_1 \\ \frac{1}{L}b'_2 \end{bmatrix} \delta \tag{8.51}$$

where m'_{ij}, d'_{ij} and b'_i are defined according to Prime systems I or II in Table 8.2

Example 8.5 (Normalization Procedure for the Nomoto Time and Gain Constants)
The gain and time constants in Nomoto's 1st- and 2nd-order models can be made invariant with respect to L and U by defining:

$$K' = (L/U)\,K, \qquad T' = (U/L)\,T \tag{8.52}$$

This suggests that the 1st-order ship dynamics can be expressed as:

$$(L/U)\,T'\,\dot{r} + r = (U/L)\,K'\,\delta \tag{8.53}$$

or

$$\dot{r} = -\left(\frac{U}{L}\right)\frac{1}{T'}r + \left(\frac{U}{L}\right)^2\frac{K'}{T'}\delta \tag{8.54}$$

This representation is quite useful since the non-dimensional gain and time constants will typically be in the range: $0.5 < K' < 2$ and $0.5 < T' < 2$ for most ships. An extension to Nomoto's 2nd-order model is obtained by writing:

$$(L/U)^2\,T'_1\,T'_2\,\psi^{(3)} + (L/U)\,(T'_1 + T'_2)\,\ddot{\psi} + \dot{\psi} = (U/L)\,K'\,\delta + K'\,T'_3\,\dot{\delta} \tag{8.55}$$

where the non-dimensional time constants T'_i are defined as: $T'_i = T_i\,(U/L)$ for $(i = 1, 2, 3)$ and the non-dimensional gain constant is $K' = (L/U)\,K$.

8.1.4 Nonlinear Models for Autopilot Design

The linear Nomoto models of Section 8.1.2 can be extended to include nonlinear effects by adding *static nonlinearities* referred to as *maneuvering characteristics*. Commonly used models are:

Nonlinear Extension of Nomoto's 1st-Order Model

In Norrbin (1963) the following 1st-order model was proposed:

$$T\dot{r} + H_N(r) = K\delta \tag{8.56}$$

$$H_N(r) = n_3 r^3 + n_2 r^2 + n_1 r + n_0 \tag{8.57}$$

where $H_N(r)$ is a nonlinear function describing the maneuvering characteristics. For $H_N(r) = r$, the linear model (8.33) is obtained.

Nonlinear Extension of Nomoto's 2nd-Order Model

Bech and Wagner Smith (1969) propose a 2nd-order model:

$$T_1T_2\ddot{r} + (T_1 + T_2)\dot{r} + KH_B(r) = K(\delta + T_3\dot{\delta}) \tag{8.58}$$

$$H_B(r) = b_3r^3 + b_2r^2 + b_1r + b_0 \tag{8.59}$$

where $H_B(r)$ can be found from Bech's reverse spiral maneuver. The linear equivalent (8.32) is obtained for $H_B(r) = r$.

The linear and nonlinear maneuvering characteristics are shown in Figure 8.13 in Section 8.3. They are generated by solving for r as an function of δ using the steady-state solutions of (8.56) or (8.58):

$$H_N(r) = K\delta, \qquad H_B(r) = \delta \tag{8.60}$$

The nonlinear maneuvering characteristics can also be generated from full-scale maneuvering tests. For stable ships both the *Bech* and *Dieudonne spiral tests* can be applied while the Bech spiral is the only one avoiding the hysteresis effect for course unstable ships; see Section 8.3 for details.

For a course-unstable ship, $b_1 < 0$ whereas a course-stable ship satisfies $b_1 > 0$. A single-screw propeller or asymmetry in the hull will cause a non-zero value of b_0. Similarly, symmetry in the hull implies that $b_2 = 0$. Since a constant rudder angle is required to compensate for constant steady-state wind and current disturbances, the bias term b_0 could conveniently be treated as an additional rudder off-set. This in turn implies that a large number of ships can be described with the simple polynomial:

$$H_B(r) = b_3r^3 + b_1r \tag{8.61}$$

The coefficients b_i $(i = 0, \ldots, 3)$ are related to those in Norrbin's model n_i $(i = 0, \ldots, 3)$ by:

$$n_i = \frac{b_i}{|b_1|} \tag{8.62}$$

resulting in:

$$H_N(r) = n_3r^3 + n_1r \tag{8.63}$$

This implies that, $n_1 = 1$ for a course-stable ship and $n_1 = -1$ for a course-unstable ship.

The Nonlinear Model of Abkowitz (1964)

One of the standard nonlinear ship models in the literature is that of Abkowitz (1964). Consider the nonlinear rigid-body equations of motion (8.3):

$$\begin{aligned} m(\dot{u} - vr - x_gr^2) &= X(\mathbf{x}) \\ m(\dot{v} + ur + x_g\dot{r}) &= Y(\mathbf{x}) \\ I_z\,\dot{r} + mx_g(\dot{v} + ur) &= N(\mathbf{x}) \end{aligned} \tag{8.64}$$

where

$$\mathbf{x} = [u, v, r, \dot{u}, \dot{v}, \dot{r}, \delta]^\top \tag{8.65}$$

Based on these equations, Abkowitz (1964) proposed a 3rd-order truncated *Taylor series* expansion of the functions $X(\mathbf{x})$, $Y(\mathbf{x})$ and $N(\mathbf{x})$ at:

$$\mathbf{x}_o = [u_o, 0, 0, 0, 0, 0, 0]^\top \tag{8.66}$$

This gives:

$$\begin{aligned}
X(\mathbf{x}) &\approx X(\mathbf{x}_0) + \sum_{i=1}^{n} \left(\frac{\partial X(\mathbf{x})}{\partial x_i}\bigg|_{\mathbf{x}_0} \Delta x_i + \frac{1}{2} \frac{\partial^2 X(\mathbf{x})}{(\partial x_i)^2}\bigg|_{\mathbf{x}_0} \Delta x_i^2 + \frac{1}{6} \frac{\partial^3 X(\mathbf{x})}{(\partial x_i)^3}\bigg|_{\mathbf{x}_0} \Delta x_i^3 \right) \\
Y(\mathbf{x}) &\approx Y(\mathbf{x}_0) + \sum_{i=1}^{n} \left(\frac{\partial Y(\mathbf{x})}{\partial x_i}\bigg|_{\mathbf{x}_0} \Delta x_i + \frac{1}{2} \frac{\partial^2 Y(\mathbf{x})}{(\partial x_i)^2}\bigg|_{\mathbf{x}_0} \Delta x_i^2 + \frac{1}{6} \frac{\partial^3 Y(\mathbf{x})}{(\partial x_i)^3}\bigg|_{\mathbf{x}_0} \Delta x_i^3 \right) \\
N(\mathbf{x}) &\approx Z(\mathbf{x}_0) + \sum_{i=1}^{n} \left(\frac{\partial N(\mathbf{x})}{\partial x_i}\bigg|_{\mathbf{x}_0} \Delta x_i + \frac{1}{2} \frac{\partial^2 N(\mathbf{x})}{(\partial x_i)^2}\bigg|_{\mathbf{x}_0} \Delta x_i^2 + \frac{1}{6} \frac{\partial^3 N(\mathbf{x})}{(\partial x_i)^3}\bigg|_{\mathbf{x}_0} \Delta x_i^3 \right)
\end{aligned}$$

where $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_0 = [\Delta x_1, \Delta x_2, ...\Delta x_n]^\top$. Unfortunately, a 3rd-order Taylor series expansion results in a large number of terms. By applying some physical insight, the complexity of these expressions can be reduced. Abkowitz (1964) makes the following assumptions:

1. *Most ship maneuvers can be described with a 3rd-order truncated Taylor expansion about the steady state condition $u = u_0$.*
2. *Only 1st-order acceleration terms are considered.*
3. *Standard port/starboard symmetry simplifications except terms describing the constant force and moment arising from single-screw propellers.*
4. *The coupling between the acceleration and velocity terms is negligible.*

Simulations of standard ship maneuvers show that these assumptions are quite good. Applying these assumptions finally yields:

$$\begin{aligned}
X &= X^* + X_{\dot{u}}\dot{u} + X_u\Delta u + X_{uu}\Delta u^2 + X_{uuu}\Delta u^3 + X_{vv}v^2 + X_{rr}r^2 + X_{\delta\delta}\delta^2 \\
&+ X_{rv}rv\delta + X_{r\delta}r + X_{v\delta}v\delta + X_{vvu}v^2\Delta u + X_{rru}r^2\Delta u + X_{\delta\delta u}\delta^2\Delta u \\
&+ X_{rvu}rvu + X_{r\delta u}r\delta\Delta u + X_{v\delta u}v\delta\Delta u \\
Y &= Y^* + Y_u\Delta u + Y_{uu}\Delta u^2 + Y_r r + Y_v v + Y_{\dot{r}}\dot{r} + Y_{\dot{v}}\dot{v} + Y_\delta\delta + Y_{rrr}r^3 + Y_{vvv}v^3 \\
&+ Y_{\delta\delta\delta}\delta^3 + Y_{rr\delta}r^2\delta + Y_{\delta\delta r}\delta^2 r + Y_{rrv}r^2 v + Y_{vvr}v^2 r + Y_{\delta\delta v}\delta^2 v + Y_{vv\delta}v^2\delta + Y_{\delta vr}\delta vr \\
&+ Y_{vu}v\Delta u + Y_{vuu}v\Delta u^2 + Y_{ru}r\Delta u + Y_{ruu}r\Delta u^2 + Y_{\delta u}\delta\Delta u + Y_{\delta uu}\delta\Delta u^2 \\
N &= N^* + N_u\Delta u + N_{uu}\Delta u^2 + N_r r + N_v v + N_{\dot{r}}\dot{r} + N_{\dot{v}}\dot{v} + N_\delta\delta + N_{rrr}r^3 + N_{vvv}v^3 \\
&+ N_{\delta\delta\delta}\delta^3 + N_{rr\delta}r^2\delta + N_{\delta\delta r}\delta^2 r + N_{rrv}r^2 v + N_{vvr}v^2 r + N_{\delta\delta v}\delta^2 v + N_{vv\delta}v^2\delta \\
&+ N_{\delta vr}\delta vr + N_{vu}v\Delta u + N_{vuu}v\Delta u^2 + N_{ru}r\Delta u + N_{ruu}r\Delta u^2 + N_{\delta u}\delta\Delta u \\
&+ N_{\delta uu}\delta\Delta u^2
\end{aligned} \tag{8.67}$$

The hydrodynamic derivatives are defined as:

$$\begin{aligned}
F^* &= F(\mathbf{x}_0), \qquad F_{x_i} = \frac{\partial F(\mathbf{x})}{\partial x_i}\bigg|_{\mathbf{x}_0} \\
F_{x_i x_j} &= \frac{1}{2}\frac{\partial^2 F(\mathbf{x})}{\partial x_i \partial x_j}\bigg|_{\mathbf{x}_0}, \quad F_{x_i x_j x_k} = \frac{1}{6}\frac{\partial^3 F(\mathbf{x})}{\partial x_i \partial x_j \partial x_k}\bigg|_{\mathbf{x}_0}
\end{aligned}$$

where $F \in \{X, Y, N\}$.

The Nonlinear Model of Norrbin (1970)

Norrbin (1970) developed a nonlinear mathematical model for ship maneuvering in deep and confined waters. This model is based on both experimental and analytical methods.

Similar to Abkowitz's model, Norrbin's model consists of three principal equations describing the axial and transverse forces (X and Y) and the yaw moment (N). Coefficients and parameters are made non-dimensional by applying the *Bis-system* (see Section 8.1.3). For deep water Norrbin's model takes the following form:

Speed equation:

$$(1 - X''_{\dot{u}})\dot{u} = \frac{1}{2}L^{-1}X''_{uu}u^2 + \frac{1}{24}L^{-2}g^{-1}X''_{uuuu}u^4 + g(1-t)T'' + (1 + X''_{vr})vr$$
$$+ L(x''_g + \frac{1}{2}X''_{rr})r^2 + \frac{1}{6}L^{-2}g^{-1}X''_{uvvv}u|v|v^2 + \frac{1}{4}L^{-1}X_{c|c|\delta\delta}|c|c\delta_e^2$$

Steering equations:

$$(1 - Y''_{\dot{v}})\dot{v} = L(Y''_{\dot{r}} - x''_g)\dot{r} + (Y''_{ur} - 1)ur + \frac{1}{2}(Lg)^{-1/2}Y''_{uur}u^2r$$
$$+ L^{-1}Y''_{uv}uv + \frac{1}{2}L^{-3/2}g^{-1/2}Y''_{uuv}u^2v + \frac{1}{2}L^{-1}Y''_{|v|v}|v|v + \frac{1}{2}LY''_{|r|r}|r|r$$
$$+ Y''_{|v|r}|v|r + Y''_{v|r|}v|r| + \frac{1}{2}L^{-1}Y''_{|c|c\delta}|c|c\delta_e + k_\gamma gT''$$

$$((k''_z)^2 - N''_{\dot{r}})\dot{r} = L^{-1}(N''_{\dot{v}} - x''_g)\dot{v} + L^{-1}(N''_{ur} - x''_g)ur$$
$$+ \frac{1}{2}L^{-3/2}g^{-1/2}N''_{uur}u^2r + L^{-2}N''_{uv}uv + \frac{1}{2}L^{-5/2}g^{-1/2}N''_{uuv}u^2v$$
$$+ \frac{1}{2}L^{-2}N''_{|v|v}|v|v + \frac{1}{2}N''_{|r|r}|r|r + L^{-1}N''_{|v|r}|v|r$$
$$+ L^{-1}N''_{v|r|}v|r| + \frac{1}{2}L^{-2}N''_{|c|c\delta}|c|c\delta_e + L^{-1}gk_NT''$$

where

δ_e	-	effective rudder angle ($\delta_e = \delta$ for $v = r = 0$)
c	-	flow velocity past rudder
T''	-	non-dimensional propeller thrust
t	-	thrust deduction factor
$(k''_z)^2 = I''_z$	-	non-dimensional squared radius of gyration
g	-	acceleration of gravity
L	-	length of hull

The *radius of gyration* with respect to the z-axis is defined as:

$$k_z = \sqrt{I_z m} \quad \Rightarrow \quad I_z = mk_z^2 \tag{8.68}$$

This number simply tells how far from the z-axis the entire mass m might be concentrated and still give the same I_z. Semi-empirical methods for estimation of the force and moment

derivatives are found in Norrbin (1970). A quasi-stationary approach can be used to model the effective rudder angle δ_e. Norrbin (1970) gives the following expression:

$$\delta_e = \delta + (k_v\, vc + k_r\, Lr2c)|\delta| \tag{8.69}$$

where δ is the rudder angle and typical values for k_v and k_r are $k_v = -0.5$ and $k_r = 0.5$. Norrbin (1970) suggests fitting the flow velocity past the rudder for positive thrust from the open water propeller diagram as:

$$c^2 = \frac{1}{2}c_{uu}^2 u^2 + c_{un}^2 un + \frac{1}{2}c_{|n|n}^2 |n|n + \frac{1}{2}c_{nn}^2 n^2 \tag{8.70}$$

where n is the propeller revolution. The four coefficients in this equation depend on the screw characteristics as well as the wake factors. Besides, the equation for the flow velocity c at the rudder, an auxiliary equation for the propeller thrust T is needed. This equation is written:

$$gT'' = \frac{1}{2}L^{-1}T''_{uu}u^2 + T''_{un}un + \frac{1}{2}LT_{|n|n}|n|n + \frac{1}{2}LT_{nn}n^2 \tag{8.71}$$

In Norrbin (1970) a more general version of this model describing large tankers in deep and confine waters is presented.

The Nonlinear Model of Blanke (1981)

A simplified form of Norrbin's nonlinear model which retains the most important terms for steering and propulsion loss assignment has been proposed by Blanke (1981). For convenience, this model is written in dimensional form as:

Speed Equation:

$$(m - X_{\dot{u}})\dot{u} = X_{|u|u}|u|u + (1-t)T + T_{\text{loss}} \tag{8.72}$$

where the loss term is:

$$T_{\text{loss}} = (m + X_{vr})vr + (mx_g + X_{rr})r^2 + X_{\delta\delta}\delta^2 + X_{\text{ext}} \tag{8.73}$$

In addition to this simplification, Blanke suggests that the terms $X_{\dot{u}}$ and $(mx_g + X_{rr})$ can be taken to be zero since these terms will be quite small for most ships. In fact, $X_{\dot{u}}$ will typically be less than 5% of the ship mass. The last term is multiplied with the square angular rate r^2, which will be less than 0.0003 (rad/s)^2 for a ship limited by a turning rate of $r_{\max} = 1$ (deg/s) $= 0.0175$ (rad/s).

Steering Equations:

$$\begin{aligned}(m - Y_{\dot{v}})\dot{v} + (mx_g - Y_{\dot{r}})\dot{r} &= -(m - Y_{ur})ur + Y_{uv}uv + Y_{|v|v}|v|v \\ &\quad + Y_{|v|r}|v|r + Y_\delta\delta + Y_{\text{ext}} \qquad (8.74)\\ (mx_g - N_{\dot{v}})\dot{v} + (I_z - N_{\dot{r}})\dot{r} &= -(mx_g - N_{ur})ur + N_{uv}uv + N_{|v|v}|v|v \\ &\quad + N_{|v|r}|v|r + N_\delta\delta + N_{\text{ext}} \qquad (8.75)\end{aligned}$$

It should be noted that all models discussed in this chapter are based on the assumption that the ship motion is restricted to the horizontal plane. In Section 9.1, it is shown how the roll motion can be included to describe the coupled ship motion in 4 DOF; that is *surge, sway, roll,* and *yaw.*

8.2 Open-Loop Stability Analysis of Ships

Stability of the uncontrolled ship can be defined as the ability to return to an equilibrium point after a disturbance, without any corrective action of the rudder. *Maneuverability,* on the other hand, is defined as the capability of the ship to carry out specific maneuvers. Excessive stability implies that the control effort will be excessive in a maneuvering situation whereas a marginally stable ship is easy to maneuver. Thus, a compromise between stability and maneuverability must be made.

8.2.1 Stability Considerations for Ship Steering and Positioning

For ships it is common to distinguish between three types of stability, namely *straight-line, directional* and *positional motion stability.* Consider the following test system:

$$\dot{x} = u\cos\psi - v\sin\psi \approx u_0\cos\psi \tag{8.76}$$

$$\dot{y} = u\sin\psi + v\cos\psi \approx u_0\sin\psi \tag{8.77}$$

$$\dot{\psi} = r \tag{8.78}$$

$$T\dot{r} + r = K\delta + w \tag{8.79}$$

where w is the external disturbances and $u_0 = \text{constant}$ is the cruise speed. Let the rudder control system be of PD type, that is:

$$\delta = -K_p\,(\psi - \psi_d) - K_d\,r \tag{8.80}$$

where $\psi_d = \text{constant}$ is used to denote the desired heading angle and K_p and K_d are two positive regulator gains. Substituting the control law (8.80) into Nomoto's 1st-order model yields the closed loop system:

$$\underbrace{T}_{m}\ddot{\psi} + \underbrace{(1 + KK_d)}_{d}\dot{\psi} + \underbrace{KK_p}_{k}\psi = \underbrace{KK_p\psi_d + w}_{f(t)} \tag{8.81}$$

The closed-loop system represents a 2nd-order mass-damper-spring system:

$$m\ddot{\psi} + d\dot{\psi} + k\psi = f(t) \tag{8.82}$$

with driving input $f(t) = k\psi_d + w$. The eigenvalues $\lambda_{1,2}$, the natural frequency ω_n, and the relative damping ratio ζ for the mass-damper-spring system are:

$$\lambda_{1,2} = \frac{-d \pm \sqrt{d^2 - 4km}}{2m}, \qquad \omega_n = \sqrt{\frac{k}{m}}, \qquad \zeta = \frac{d}{2}\frac{1}{\sqrt{km}} \tag{8.83}$$

Matlab:
The closed-loop system (8.81) is simulated in Matlab™ using the GNC Toolbox script:

```
StabDemo
```

The following stability considerations are made for the system (8.81):

Instability: For uncontrolled ships ($K_p = K_d = 0$) instability occurs when:

$$\lambda_1 = -\frac{d}{m} = -\frac{1}{T} > 0 \qquad \text{and} \qquad \lambda_2 = 0$$

which simply states that $T < 0$. This is common for large tankers.

Straigh-Line Stability: Consider an uncontrolled ship ($K_p = K_d = 0$) moving in a straight path. If the new path is straight after a disturbance w in yaw the ship is said to have straight-line stability. The direction of the new path will usually differ from the initial path because no restoring forces are present ($k = 0$). This corresponds to:

$$\lambda_1 = -\frac{d}{m} = -\frac{1}{T} < 0 \qquad \text{and} \qquad \lambda_2 = 0$$

Consequently, the requirement $T > 0$ implies straight-line stability for the uncontrolled ship ($\delta = 0$).

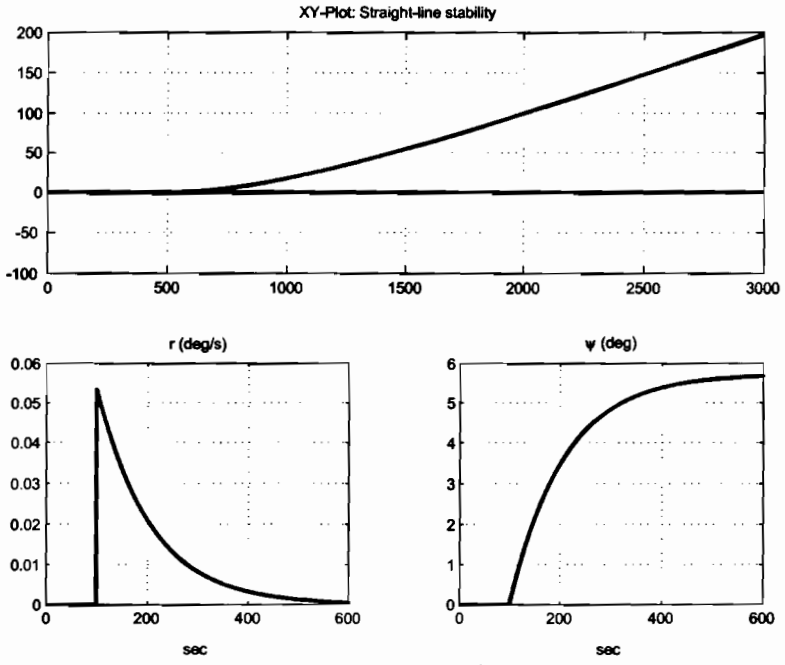


Figure 8.3: Straight-line stability for a ship when an impulse $w(t)$ is injected at $t = 100$ (s).

Directional Stability (Stability on Course): Directional stability is a much stronger requirement than straight-line stability. Directional stability requires the final path to be parallel to the initial path which is obtained for $K_p > 0 \Rightarrow k > 0$. Additional damping is added through $K_d > 0$, that is, PD-control. The ship is said to be directionally stable if both eigenvalues have negative real parts, that is:

$$\mathrm{Re}\{\lambda_{1,2}\} < 0$$

The following two types of directional stability are observed:

No oscillations ($d^2 - 4km \geq 0$): This implies that both eigenvalues are negative and real– i.e., $\zeta \geq 1$ such that:

$$\lambda_{1,2} = \frac{-d \pm \sqrt{d^2 - 4km}}{2m} = \left(-\zeta \pm \sqrt{\zeta^2 - 1}\right)\omega_n < 0$$

For a critically damped system $\zeta = 1.0$, such that $\lambda_{1,2} = -\frac{2}{2m} = -\omega_n$.

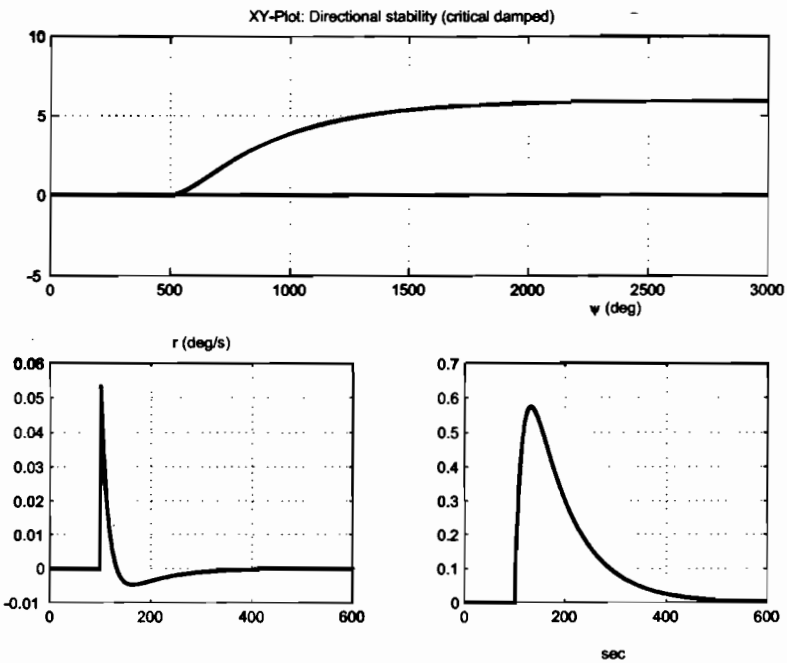


Figure 8.4: Directional stability for a critically damped ship ($\zeta = 1.0$) when an impulse $w(t)$ is injected at $t = 100$ (s).

Damped oscillator ($d^2 - 4km < 0$): This corresponds to two imaginary eigenvalues $\lambda_{1,2}$ with negative real parts ($\zeta < 1$), that is:

$$\lambda_{1,2} = \frac{-d \pm j\sqrt{4km - dm}}{2m} = \left(-\zeta \pm j\sqrt{1-\zeta^2}\right)\omega_n$$

Directional stability for a critically damped ($\zeta = 1.0$) and under damped ship ($\zeta = 0.1$) is shown in Figures 8.4–8.5. Notice the oscillations in both positions and yaw angle in Figure 8.5. Directional stability requires feedback control since there are no restoring forces in yaw. However, in heave, roll and pitch where metacentric restoring forces are present ($k > 0$) no feedback is required to damp out the oscillations.

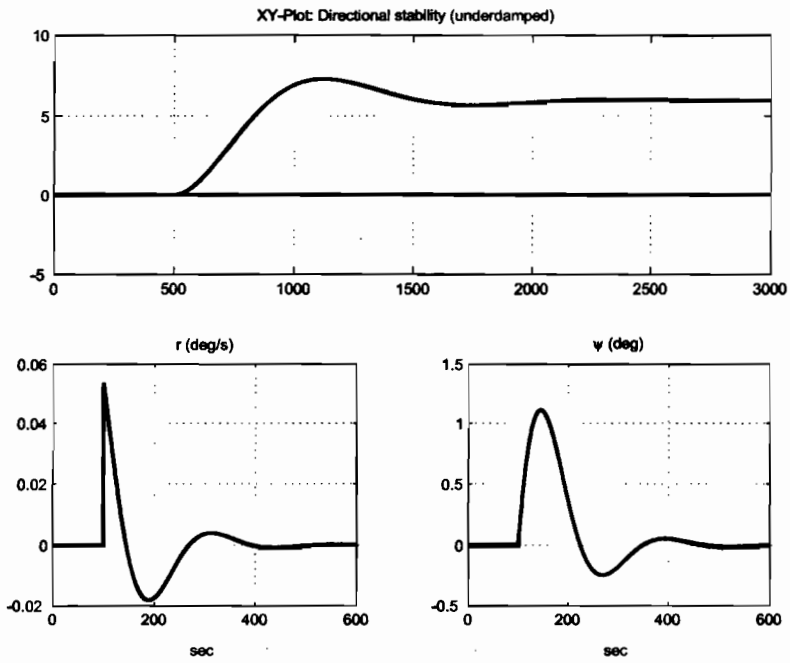


Figure 8.5: Directional stability for a under damped ship ($\zeta = 0.1$) when an impulse $w(t)$ is injected at $t = 100$ (s).

Positional Motion Stability: Positional motion stability implies that the ship should return to its original path after a disturbance. This can be achieved by including integral action in the controller. Hence, a PID-controller can be designed to compensate for the unknown disturbance term w while a PD-controller will generally result in a steady state offset.

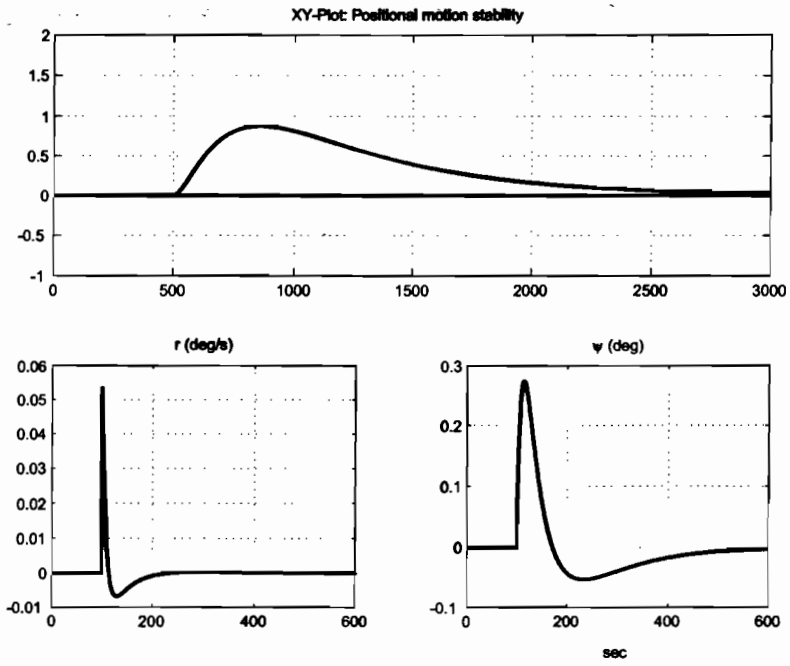


Figure 8.6: Positional motion stability for a PID-controlled ship when an impulse $w(t)$ is injected at $t = 100$ (s).

Example 8.6 (Straigh-Line Stability)
Consider the cargo ship and oil tanker of Example 8.1. Recall that the equivalent time constant in Nomoto's 1st-order model was defined as:

$$T = T_1 + T_2 - T_3$$

Hence, the uncontrolled cargo ship has an equivalent time constant:

$$T_{cargo\ ship} = 118.0 + 7.8 - 18.5 = 107.3\ (s) > 0$$

while the oil tanker has an equivalent time constant

$$T_{oil\ tanker} = -124.1 + 16.4 - 46.0 = -153.7\ (s) < 0$$

This implies that the cargo ship is straight-line stable while the oil tanker is unstable.

8.2.2 Criteria for Straight-Line Stability

Recall from Section 8.2.1 that a ship is said to be dynamically straight-line stable if it returns to a straight-line motion after a disturbance in yaw without any corrective action from the rudder. Consequently, instability refers to the case when the ship goes into a starboard or port turn without any rudder deflections. In the same section, Nomoto's 1st-order model was

used to find a simple criterion for straight-line motion. This leads to the requirement that the time constant T must be positive. Similarly, it is possible to derive a criterion for straight-line stability for the state-space model (8.20):

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_0)\boldsymbol{\nu} = \mathbf{b}\delta \tag{8.84}$$

where $\boldsymbol{\nu} = [v, r]^\top$. Applications of *Laplace's transformation* to the linear model (8.84) with $\boldsymbol{\nu}(0) = \mathbf{0}$, yields:

$$[\mathbf{M}s + \mathbf{N}(u_0)]\boldsymbol{\nu}(s) = \mathbf{b}\delta(s) \tag{8.85}$$

Consequently,

$$\boldsymbol{\nu}(s) = [\mathbf{M}s + \mathbf{N}(u_0)]^{-1}\mathbf{b}\delta(s) = \frac{\text{adj}(\mathbf{M}s + \mathbf{N}(u_0))}{\det(\mathbf{M}s + \mathbf{N}(u_0))}\mathbf{b}\delta(s) \tag{8.86}$$

The characteristic equation is:

$$\det(\mathbf{M}\sigma + \mathbf{N}(u_0)) = A\,\sigma^2 + B\,\sigma + C = 0 \tag{8.87}$$

where

$$\begin{aligned} A &= \det(\mathbf{M}) \\ B &= n_{11}m_{22} + n_{22}m_{11} - n_{12}m_{21} - n_{21}m_{12} \\ C &= \det(\mathbf{N}(u_0)) \end{aligned} \tag{8.88}$$

The two roots $\sigma_{1,2}$ of (8.87), both of which must have negative real parts for open-loop stability are:

$$\text{Re}\{\sigma_{1,2}\} = \text{Re}\left\{\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}\right\} < 0 \tag{8.89}$$

The quantities $\sigma_{1,2}$ are often referred to as the controls-fixed stability indexes for straight-line stability.

The Routh stability criterion was developed in the 1860s by the British scientist E. J. Routh.

Theorem 8.1 (The Routh Stability Criterion)
Consider the characteristic equation:

$$a_n\lambda^n + a_{n-1}\lambda^{n-1} + a_{n-2}\lambda^{n-2} + \ldots + a_0 = 0 \tag{8.90}$$

To apply the Routh criterion, the Routh array shown in Table 8.4) must be constructed. The coefficients a_i are the coefficients of the characteristic equation (8.90) and b_i, c_i, d_i, etc. are defined as:

$$\begin{aligned} b_1 &= a_{n-1}a_{n-2} - a_n a_{n-3} a_{n-1} & b_2 &= a_{n-1}a_{n-4} - a_n a_{n-5} a_{n-1} & \ldots \\ c_1 &= b_1 a_{n-3} - a_{n-1} b_2 b_1 & c_2 &= b_1 a_{n-5} - a_{n-1} b_3 b_1 & \ldots \\ d_1 &= c_1 b_2 - c_2 b_1 c_1 & \ldots & & \end{aligned}$$

Table 8.4: Routh array.

λ^n	a_n	a_{n-2}	a_{n-4}	...
λ^{n-1}	a_{n-1}	a_{n-3}	a_{n-5}	...
λ^{n-2}	b_1	b_2	b_3	...
λ^{n-3}	c_1	c_2	c_3	...
λ^{n-4}	d_1	d_2	d_3	...
$\vdots$	...			

Necessary and sufficient conditions for the system to be stable are:

1. All the coefficients of the characteristic equation must be non-zero and have the same sign.
2. All the coefficients of the first column of the Routh array must have the same sign.

If Condition 2 is violated, the number of sign changes will indicate how many roots of the characteristic equation which will have positive real parts. Hence, the system will be unstable.

Proof. See Routh (1877). ■

According to the Routh Stability criterion, necessary and sufficient conditions for the ship to be stable are:

$$A,\ B,\ C\ >\ 0 \tag{8.91}$$

The first condition $A = \det(\mathbf{M}) > 0$ is automatically satisfied since the vessels's inertia matrix $\mathbf{M}$ always is positive definite. Condition $B > 0$ implies that:

$$n_{11}m_{22} + n_{22}m_{11} > n_{12}m_{21} + n_{21}m_{12} \tag{8.92}$$

Consequently, the products of the diagonal elements of $\mathbf{M}$ and $\mathbf{N}(u_0)$ must be larger than the products of the off-diagonal elements. This is is satisfied for most ships. Consequently, condition (8.91) reduces to:

$$C = \det(\mathbf{N}(u_0)) > 0 \tag{8.93}$$

This condition has also been proved by Abkowitz (1964) who stated the following theorem.

Theorem 8.2 (Straight-Line Stability (Abkowitz 1964))
A ship is dynamically stable in straight-line motion if the hydrodynamic derivatives satisfy:

$$\begin{aligned} \det(\mathbf{N}(u_0)) &= \det\begin{bmatrix} -Y_v & mu_0 - Y_r \\ -N_v & mx_g u_0 - N_r \end{bmatrix} \\ &= Y_v(N_r - mx_g u_0) - N_v(Y_r - mu_0) > 0 \end{aligned} \tag{8.94}$$

Proof. This is seen as a consequence of (8.93) and (8.22). ■

It is interesting to notice that making C more positive will improve stability and thus reduce the ship's maneuverability, and the other way around. Straight-line stability implies that the new path of the ship will be a straight line after a disturbance in yaw. The direction of the new path will usually differ from the initial path. Contrary to this, unstable ships will go into a starboard or port turn without any rudder deflection. It should be noted that most modern large tankers are slightly unstable. For such ships, the criterion (8.94) corresponds to one of the poles being in the right half-plane.

Straight-Line Stability in Terms of Time Constants

The criterion (8.91) can be related to Nomoto's 2nd-order model (8.26) by noticing that:

$$T_1 T_2 = \frac{A}{C} > 0; \qquad T_1 + T_2 = \frac{B}{C} > 0 \tag{8.95}$$

Consequently, straight-line stability is guaranteed if $T_1 > 0$ and $T_2 > 0$. This can also be seen from:

$$\sigma_{1,2} = -\frac{1}{T_{1,2}} = \mathrm{Re}\left\{\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}\right\} < 0 \tag{8.96}$$

8.2.3 Criteria for Directional Stability

Dynamic stability on course, or directional stability, cannot be obtained without activating the rudder. Usually a PID-control system is used to generate the necessary rudder action to stabilize the ship. For simplicity, consider a PD-controller:

$$\delta = -K_p\,(\psi - \psi_d) - K_d r \tag{8.97}$$

which after substitution into Nomoto's 2nd-order model, yields the closed-loop dynamics:

$$T_1 T_2 \psi^{(3)} + (T_1 + T_2 + T_3 K K_d)\ddot{\psi} + (1 + K K_d + T_3 K K_p)\dot{\psi} + K K_p \psi = K K_p \psi_d \tag{8.98}$$

From this expression, the cubic characteristic equation:

$$A\,\sigma^3 + B\,\sigma^2 + C\,\sigma + D = 0 \tag{8.99}$$

is recognized, where:

$$A = T_1 T_2 \tag{8.100}$$
$$B = T_1 + T_2 + T_3 K K_d \tag{8.101}$$
$$C = 1 + K K_d + T_3 K K_p \tag{8.102}$$
$$D = K K_p \tag{8.103}$$

The requirement for directional stability is:

$$\mathrm{Re}\{\sigma_{1,2,3}\} < 0 \tag{8.104}$$

This can be checked by forming the Routh array:

$$\begin{array}{ll} A & C \\ B & D \\ \frac{BC-AD}{B} & 0 \\ D & \end{array} \tag{8.105}$$

Consequently, sufficient and necessary conditions for the ship to be dynamically stable on course are:

$$(i) \qquad A, B, C, D > 0 \tag{8.106}$$

$$(ii) \qquad BC - AD > 0 \tag{8.107}$$

This again implies that the controller gains K_p and K_d must be chosen such that the conditions (8.106) and (8.107) are satisfied.

8.3 Maneuverability

Several ship maneuvers can be used to evaluate the robustness, performance and limitations of a ship. This is usually done by defining a criterion in terms of a *maneuvering index* or by simply using a *maneuvering characteristic* to compare the maneuverability of the test ship with previously obtained results from other ships. A frequently used measure of maneuverability is the turning index of Norrbin (1965).

The Norrbin Measure of Maneuverability

Norrbin (1965) defines a *turning index* as:

$$P = \frac{\psi'(t' = 1)}{\delta'(t' = 1)} \tag{8.108}$$

where $t' = t(U/L)$ is the nondimensional time. P is a measure of turning ability or maneuverability since it can be interpreted as the heading change per unit rudder angle in one ship length traveled at $U = 1$ (m/s). An expression for P can be found by solving the ODE:

$$T'\ddot{\psi}' + \dot{\psi}' = K'\delta' \tag{8.109}$$

with $\delta' =$ constant. This results in:

$$\psi'(t') = K'[t' - T' + T' \exp(-(t'/T'))]\delta'(t') \tag{8.110}$$

A 2nd-order Taylor expansion of $\exp(-t'/T')$ is:

$$\exp(-t'/T') = 1 - \frac{t'}{T'} + \frac{(t')^2}{2(T')^2} + O(3) \tag{8.111}$$

such that:

$$\frac{\psi'(t')}{\delta'(t')} \approx K'\left[t' - T' + T'\left(1 - \frac{t'}{T'} + \frac{(t')^2}{2(T')^2}\right)\right] = K\frac{(t')^2}{2T'} \quad (8.112)$$

$$\frac{\psi'(t'=1)}{\delta'(t'=1)} \approx K'\left[\frac{(t')^2}{2T'}\right]_{t'=1} = \frac{K'}{2T'} \quad (8.113)$$

Consequently:

$$P \approx \frac{1}{2}\frac{K'}{T'} \quad (8.114)$$

The P-number is a good measure of maneuverability for course-stable ships. Norrbin concludes that $P > 0.3$ guarantees a reasonable standard of course change quality for most ships while $P > 0.2$ seems to be sufficient for large oil-tankers. For poorly stable ships it is recommended to use P together with another maneuverability index, for instance the slope $dr'/d\delta'$ or the width of the r'–δ' loop; see Section 8.3.4.

Maneuvering Characteristics

A maneuvering characteristic can be obtained by changing or keeping a predefined course and speed of the ship in a systematic manner by means of active controls. For most surface vessels these controls are rudders, fins, propellers and thrusters. However, since ship maneuverability depends on the water depth, environmental disturbances, ship speed and hydrodynamic derivatives etc. care must be taken when performing a full-scale maneuvering test. A guide for sea trials describing how these maneuvers should be performed is found in SNAME (1989) . The following standard ship maneuvers have been proposed by the International Towing Tank Conference (ITTC):

- **Turning Circle:** This trial is mainly used to calculate the ship's steady turning radius and to check how well the steering machine performs under course-changing maneuvers.
- **Kempf's Zig-Zag Maneuver:** The zig-zag test is a standard maneuver used to compare the maneuvering properties and control characteristic of a ship with those of other ships. Another feature is that the experimental results of the test can be used to calculate the K and T values of Nomoto's 1st-order model.
- **Pull-Out Maneuver:** The pull-out maneuver can be used to check whether the ship is straight-line stable or not. The maneuver can also be used to indicate the degree of stability.
- **Dieudonné's Spiral Maneuver:** The spiral maneuver is also used to check straight-line stability. The maneuver gives an indication of the range of validity of the linear theory.
- **Bech's Reverse Spiral Maneuver:** The reverse spiral maneuver can be used for unstable ships to produce a nonlinear maneuvering characteristic. The results from the test indicate which rudder corrections that are required to stabilize an unstable ship.

- **Stopping Trials:** Crash-stops and low-speed stopping trials can be used to determine the ship's head reach and maneuverability during emergency situations.

8.3.1 Turning Circle

This is probably the oldest maneuvering test. The test can be used as an indication on how well the steering machine and rudder control performs during course-changing maneuvers. It is also used to calculate standard measures of maneuverability like *tactical diameter, advance* and *transfer* shown in Figure 8.7); see Gertler and Hagen (1960) for a detailed description.

Matlab:
The turning circle for the Mariner class vessel is computed using the GNC Toolbox script ExTurnCircle.m where:

```
t_final = 700;              % final simulation time (sec)
t_rudderexecute = 100;      % time rudder is executed (sec)
h = 0.1;                    % sampling time (sec)

% Mariner class cargo ship, cruise speed U0 = 7.7 m/s
x = zeros(7,1);     % x=[u v r x y psi delta]' (initial values)
u i= -15*pi/180;    % delta_c=-delta_R at time t = t_rudderexecute

[t,u,v,r,x,y,psi,U] =...
       turncircle('mariner', x, ui, t_final, t_rudderexecute, h);
```

The results are shown in Figure 8.7. Similar results are obtained by replacing, mariner.m, with the container ship, container.m; see ExTurnCircle.m.

The maneuvering characteristics for the Mariner class vessel were computed to be

Rudder execute (x-coordinate):	769 m
Steady turning radius:	711 m
Maximum transfer:	1315 m
Maximum advance:	947 m
Transfer at 90 (deg) heading:	534 m
Advance at 90 (deg) heading:	943 m
Tactical diameter at 180 (deg) heading:	1311 m

The *steady turning radius* R is perhaps the most interesting quantity obtained from the turning trials. In the maneuvering trial code of the 14th ITTC (1975) it is proposed to turn the ship over at maximum speed and with a rudder angle of minimum 15 degrees to obtain the turning circle. The rudder angle δ should be held constant such that a constant rate of turn is reached (in practice a turning circle of 540 degrees may be necessary).

The output from a positioning system is used to calculate the tactical diameter, steady turning radius, maximum advance and maximum transfer. A typical turning circle corresponding to a negative rudder angle is shown in Figure 8.7.

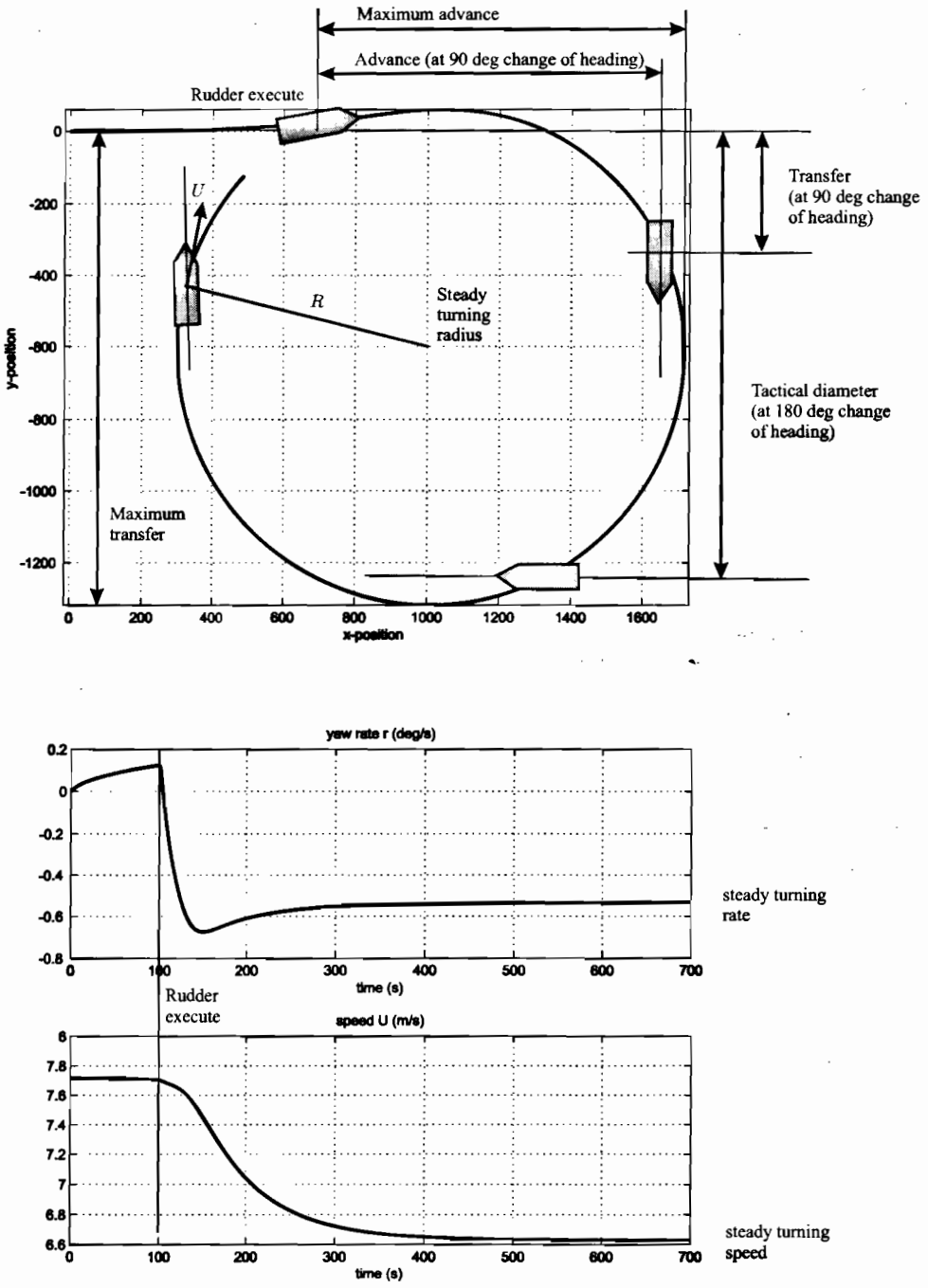


Figure 8.7: Turning circle, yaw rate and speed for the Mariner class vessel for a constant rudder angle $\delta_R = -15$ deg applied at $t = 100$ sec.

For a constant rudder angle δ, the ship will move in a circle with constant turning radius R and speed U in steady state, that is $\dot{\boldsymbol{\nu}} = \mathbf{0}$. Solving (8.20) for the steady-state solution of $\boldsymbol{\nu} = [v, r]^\top$, yields:

$$\mathbf{N}(u_0)\boldsymbol{\nu} = \mathbf{b}\delta \quad \Longrightarrow \quad \boldsymbol{\nu} = \mathbf{N}^{-1}(u_0)\mathbf{b}\delta \tag{8.115}$$

The equation for r in this expression becomes:

$$r = \frac{(Y_v N_\delta - N_v Y_\delta)}{Y_v(N_r - mx_g u_0) - N_v(Y_r - mu_0)}\,\delta \tag{8.116}$$

The ship's turning radius R is defined as:

$$R = \frac{U}{r} \qquad \text{where} \qquad U = \sqrt{u^2 + v^2} \tag{8.117}$$

Introducing the length $L = L_{pp}$ of the ship the following expression for the ratio (R/L) is obtained:

$$\left(\frac{R}{L}\right) = \left(\frac{U}{L}\right)\frac{C}{(Y_v N_\delta - N_v Y_\delta)}\frac{1}{\delta}, \qquad \delta \neq 0 \tag{8.118}$$

where

$$C = \det(\mathbf{N}(u_0) = Y_v(N_r - mx_g u_0) - N_v(Y_r - mu_0) > 0 \qquad \text{(stable ship)}$$

is recognized as one of the stability derivatives in the straight-line stability criterion discussed in Section 8.2.2. From (8.118) it is seen that increased stability (large C) implies that the turning radius will increase. Consequently, a highly stable ship requires more maneuvering effort than a marginally stable one. The ratio (R/L) can also be written in terms of non-dimensional quantities by:

$$\left(\frac{R}{L}\right) = \frac{Y'_v(N'_r - m'x'_g) - N'_v(Y'_r - m')}{Y'_v N'_\delta - N'_v Y'_\delta}\frac{1}{\delta}, \qquad \delta \neq 0 \tag{8.119}$$

This formula is independent of the ship speed. It should be noted that the formulas for the turning radius are based on linear theory which assumes that δ is small and accordingly that R is large.

Another feature of the turning test is that the Nomoto gain and time constant can be determined. This is illustrated in the following example where a cargo ship is considered.

Example 8.7 (Determination of the Nomoto Gain and Time Constants)
The Nomoto gain and time constants can be computed from a turning test by using nonlinear least-squares curve fitting, for instance. Solving the ODE:

$$T\dot{r} + r = K\delta \tag{8.120}$$

for a step input $\delta = \delta_0 = $ constant, yields:

$$r(t) = \exp(-t/T)r(0) + [1 - \exp(-t/T)]\,K\delta_0 \tag{8.121}$$

where K and T are unknowns. The Matlab™ GNC Toolbox script ExKT.m *fits this model to a simulated step response of the model* mariner.m *which is a nonlinear model of the Mariner*

class vessel.

The results for a step $\delta_0 = 5$ *(deg) and* $U = 7.7$ *(m/s)*=15 *(knots), are (see Figure 8.8):*

$$K = 0.09\ (s^{-1}), \quad T = 22.6\ (s) \tag{8.122}$$

The Norrbin measure of maneuverability becomes:

$$P = \frac{1}{2}\frac{K'}{T'} = \frac{1}{2}\frac{K}{T}\left(\frac{L}{U}\right)^2 = \frac{1}{2}\left(\frac{0.09}{22.6}\right)\left(\frac{160.9}{7.7}\right)^2 = 0.87 \tag{8.123}$$

which guarantees good maneuverability since $P > 0.3$. *The turning circle is shown in Figure 8.7 indicating that the steady-state turning radius is* 711 *(m).*

Matlab:

```
% ExKT Script for computation of Nomoto gain and time constants
% using nonlinear least-squares.The rudder input is 5 deg at t=0.

N = 2000;       % number of samples
h = 0.1;        % sample time

xout = zeros(N,2);
x = zeros(7,1);
delta_R = 5*(pi/180);              % rudder angle step input

for i=1:N,
   xout(i,:)  = [(i-1)*h ,x(3)];
   xdot = mariner(x,delta_R);      % nonlinear Mariner model
   x = euler2(xdot,x,h);           % Euler integration
end

% time-series
tdata = xout(:,1);
rdata = xout(:,2)*180/pi;

% nonlinear least-squares parametrization:  x(1)=1/T and x(2)=K
x0 = [0.01 0.1]'
F = inline('exp(-tdata*x(1))*0 +...
            x(2)*(1-exp(-tdata*x(1)))*5','x','tdata')
x = lsqcurvefit(F,x0, tdata, rdata);

plot(tdata,rdata,'g',tdata,exp(-tdata*x(1))*0 +...
      x(2)*(1-exp(-tdata*x(1)))*5,'r'),grid
title('NLS fit of Mariner model for \delta = 5 (deg)')
xlabel('time (s)')
legend('Nonlinear model','Estimated 1st-order Nomoto model')
```

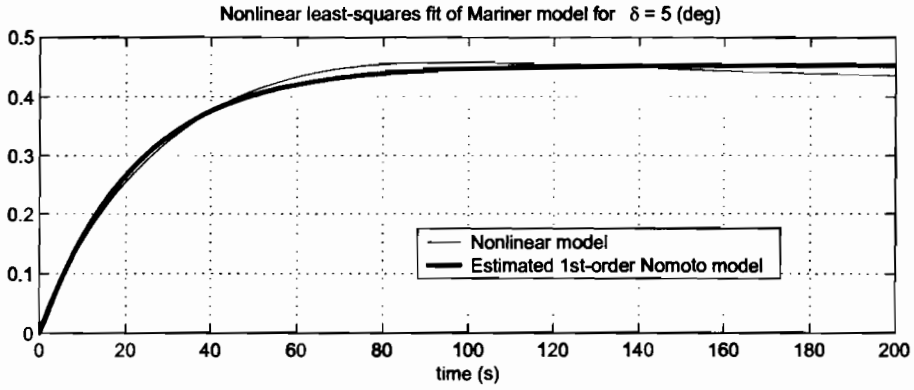


Figure 8.8: Plot showing the estimated linear model and the nonlinear Mariner model for a step $\delta = \delta_0 = 5$ (deg).

8.3.2 Kempf's Zig-Zag Maneuver

The zig-zag test was first proposed by Kempf (1932) . Comprehensive test results of 75 freighters are published in Kempf (1944).

The zig-zag time-response (see Figures 8.9–8.10) is obtained by moving the rudder to 20 degrees starboard from an initially straight course. The rudder setting is kept constant until the heading is changed 20 degrees, then the rudder is reversed 20 degrees to port. Again, this rudder setting is maintained until the ship's heading has reached 20 degrees in the opposite direction. This process continues until a total of 5 rudder step responses have been completed. This test is usually referred to as a 20^o–20^o maneuver (the first angle refers to the actual rudder settings while the second angle denotes how much the heading angle should change before the rudder is reversed) and was standardized by the International Towing Tank Conference (ITTC) in 1963.

For larger ships, ITTC has recommended the use of a 10^o–10^o or a 20^o–10^o maneuver to reduce the time and waterspace required. The only apparatus required to perform the test is a compass and a stopwatch. Alternatively a PC interfaced for real-time logging of compass data can be used. The results from the zig-zag maneuver can be used to compare the maneuvering properties of different ships

Example 8.8 (Zig-Zag Maneuvering Trials)
Both the Mariner class vessel (mariner.m) and the container ship (container.m) are simulated for a 20^o–20^o and a 20^o–10^o zig-zag maneuver, respectively, by using the Matlab™ script ExZigZag.m.

The simulation results for the two vessels are shown in Figure 8.9–8.10.

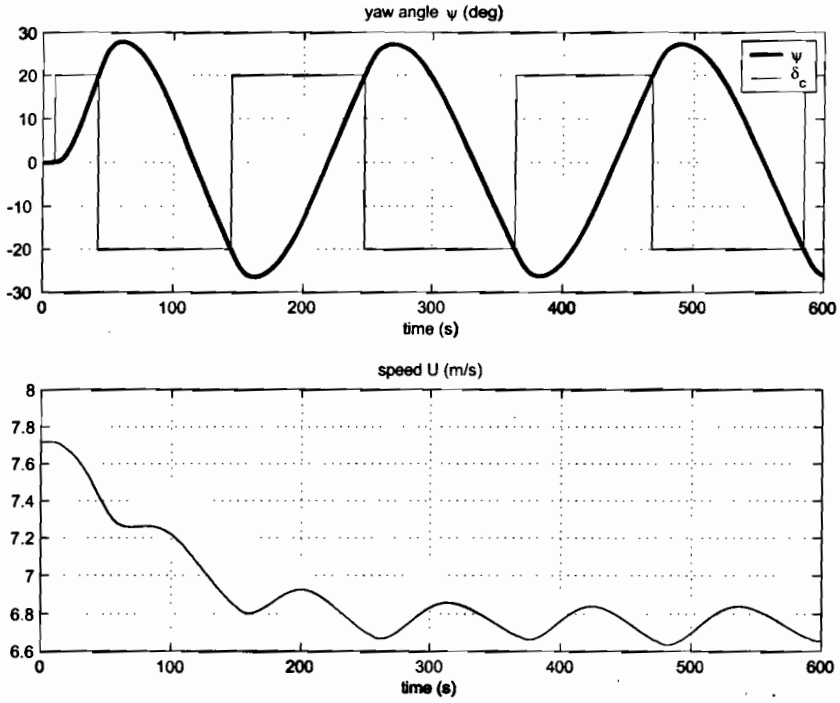


Figure 8.9: 20^o–20^o maneuver for the Mariner class vessel.

Matlab:

```
t_final = 600;              % final simulation time (sec)
t_rudderexecute = 10;       % time rudder is executed (sec)
h = 0.1;                    % sampling time (sec)

% 20-20 zig-zag maneuver for the Mariner class cargo ship
% cruise speed U0 = 7.7 m/s (see mariner.m)
x = zeros(7,1);    % x = [ u v r x y psi delta ]' (initial values)
ui = 0;            % delta_c = 0 for time t < t_rudderexecute
[t,u,v,r,x,y,psi,U] =...
       zigzag('mariner',x,ui,t_final,t_rudderexecute,h,[20,20]);

% 20-10 zig-zag maneuver for a container ship
% cruise speed 8.0 m/s see container.m)
x = [8.0 0 0 0 0 0 0 0 0 70]'; % x = [ u v r x y psi delta n ]'
delta_c = 0;        % delta_c = 0 for time t < t_rudderexecute
n_c = 80;           % n_c = propeller revolution in rpm
ui = [delta_c, n_c];
[t,u,v,r,x,y,psi,U] =...
      zigzag('container',x,ui,t_final,t_rudderexecute,h,[20,10]);
```

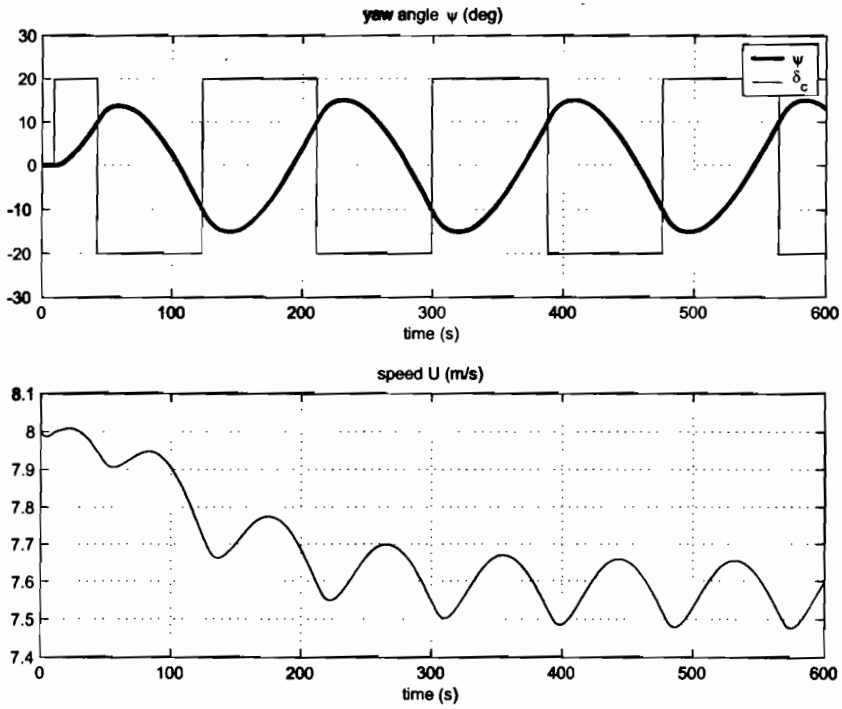


Figure 8.10: 20^o–10^o maneuver for the container ship.

8.3.3 Pull-Out Maneuver

In 1969 Roy Burcher proposed a simple test procedure to determine whether a ship is straight-line stable or not. This test is referred to as the pull-out maneuver (12th ITTC 1969a). The pull-out maneuver involves a pair of maneuvers in which a rudder angle of approximately 20 degrees is applied and returned to zero after steady turning has been attained. Both a port and starboard turn should be performed.

During the test the ship's rate of turn must be measured or at least calculated by numerical derivation of the measured compass heading. If the ship is straight-line stable the rate of turn will decay to the same value for both the starboard and port turn (see Figure 8.11). The ship is unstable if the steady rate of turn from the port and starboard turn differ (see Figure 8.12). The difference between these two steady rates of turn corresponds exactly to the height of the Dieudonné's spiral loop that we will discuss next.

Example 8.9 (Pullout Maneuver for a Stable and an Unstable Ship)
Both the Mariner class vessel (mariner.m) and the Esso Osaka tanker (tanker.m) are simulated under a pullout maneuver by using the Matlab™ script ExPullout.m.

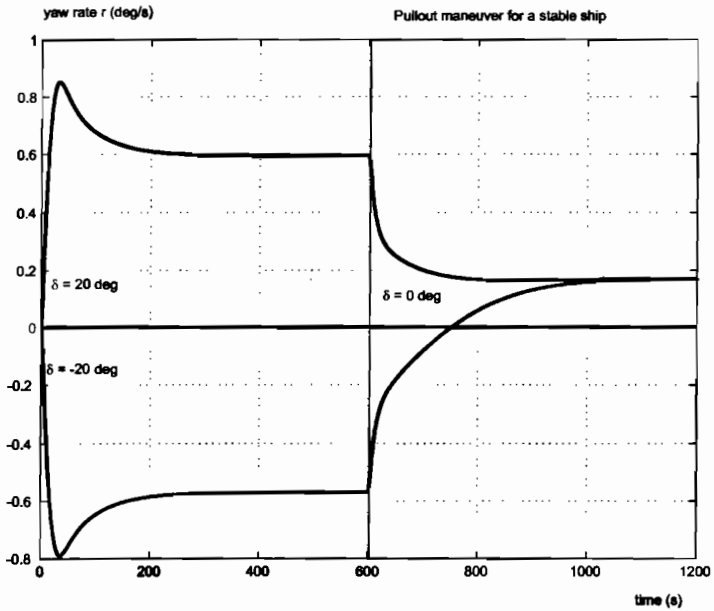


Figure 8.11: Pullout maneuver for the Mariner class vessels. Notice that the positive and negative curves meet for the stable ship.

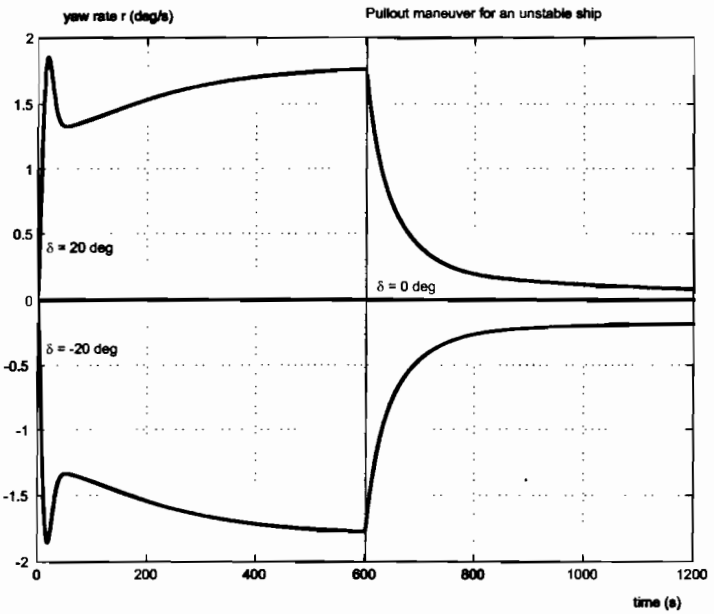


Figure 8.12: Pullout maneuver for the Esso Osaka tanker. Notice that the positive and negative curves do not meet.

Matlab:

```
delta_c = 20*pi/180;    % rudder angle for maneuver (rad)
h = 0.1;                % sampling time (sec)

% Mariner class cargo ship, speed U0 = 7.7 m/s (see mariner.m)
x = zeros(7,1);    % x = [ u v r x y psi delta ]' (initial values)
ui = delta_c;      % ui = delta_c
[t,r1,r2] = pullout('mariner',x,ui,h);

% The Esso Osaka tanker (see tanker.m)
n = 80;
U = 8.23;
x = [ U 0 0 0 0 0 0 n ]';    % x = [ u v r x y psi delta n ]'
n_c = 80;                    % n_c = propeller revolution in rpm
depth = 200;                 % water depth
ui = [delta_c, n_c, depth];
[t,r1,r2] = pullout('tanker',x,ui,h);
```

The results are shown in Figures 8.11–8.12 where the curves meet for the stable ship (Mariner class vessel) while there is an off-set between the curves for the unstable model of the Esso Osaka tanker.

8.3.4 Dieudonné's Spiral Maneuver

The direct spiral test was published first in 1949–1950 by the French scientist Jean Dieudonné. An English translation is found in Dieudonné (1953). The direct spiral maneuver is used to check straight-line stability. As seen from Figure 8.13, the maneuver also gives an indication of the degree of stability and the range of validity of the linear theory.

To perform the test the ship should initially be held on a straight course. The rudder angle is then put to 25 degrees starboard and held until steady yawing rate is obtained. After this the rudder angle is decreased in steps of 5 degrees and again held until constant yawing rates are obtained for all the rudder angles. The procedure is performed for all rudder angles between 25 degrees starboard and 25 degrees port. In the range around zero rudder angle the step of 5 degrees rudder should be reduced to obtain more precise values. The results are plotted in an r–δ diagram as shown in Figure 8.13. It should be noted that the spiral maneuver should be performed in still air and calm water to obtain the best results.

For straight-line unstable ships it is recommended to use Bech's reverse spiral maneuver.

8.3.5 Bech's Reverse Spiral Maneuver

For stable ships both Dieudonné's direct and Bech's reverse spiral tests can be used. For unstable ships within the limits indicated by the pull-out maneuver Bech's reverse spiral should be applied. The reverse spiral test was first published by Mogens Bech in 1966 at the Nordic ship technical meeting in Malmö, Sweden and later by Bech (1968). Since then

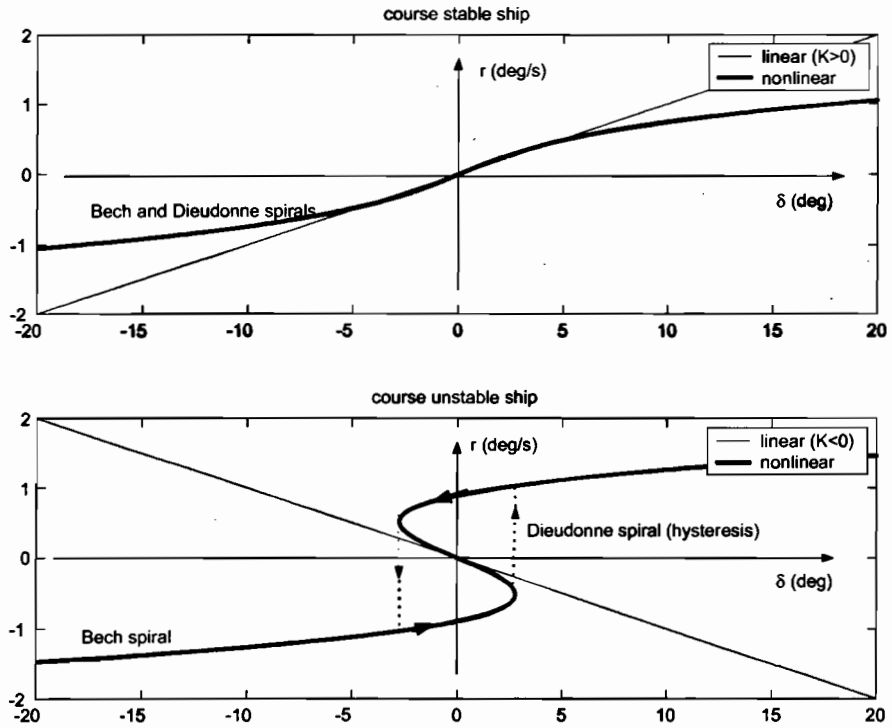


Figure 8.13: $r - \delta$ diagram showing the Dieudonne amd Bech spirals for both a stable and course stable ship. Notice the hysteresis loop in the Dieudonne spiral for the unstable ship.

the reverse spiral test has been quite popular, because of the simplicity and reliability of the method. The reverse spiral is particularly attractive since it is less time-consuming than Dieudonné's spiral test.

By observing that the ship steering characteristic is nonlinear outside a limited area, Bech (1968) suggested that one describe the *mean* value of the required rudder deflection δ_{ss} to steer the ship at a constant rate of turn r_{ss} as a nonlinear function:

$$\delta_{ss} = H_B(r_{ss}) \tag{8.124}$$

where $H_B(r_{ss})$ is a nonlinear function describing the maneuvering characteristic.

This can be understood by considering Nomoto's 2nd-order model:

$$T_1T_2\ddot{r} + (T_1 + T_2)\dot{r} + KH_B(r) = K(\delta + T_3\dot{\delta}) \tag{8.125}$$

where the linear term r has been replaced with a function $H_B(r)$. Assuming that $r = r_{ss}$ is constant in steady-state, that is, $\ddot{r} = \dot{r} = \dot{\delta} = 0$ directly gives (8.124). This implies that the r–δ curve will be a single-valued (one-to-one) function of r for both the stable and unstable ship; see Figure 8.13. If the conventional spiral test is applied to an unstable ship a hysteresis loop will be observed.

The full-scale test is performed by measuring the necessary rudder action required to bring the ship into a desired rate of turn. For an unstable ship this implies that the rudder

angle will oscillate about a mean rudder angle. The amplitude of the rudder oscillations should be kept to a minimum. After some time a "*balance condition*" is reached and both the mean rudder angle and rate of turn can be calculated. Care should be taken for large ships since they will require some more time to converge to their "balance condition".

8.4 Course-Keeping Autopilots and Turning Control

The different autopilot blocks of Figure 8.1 are designed in this section using the ship models of Section 8.1. We will first discuss reference models before the different feedback control strategies are applied.

8.4.1 Autopilot Reference Model

A modern autopilot must have both course-keeping and turning capabilities. This can be obtained in one design by using a reference model to compute the desired states ψ_d, r_d, and $\dot{r}_d$ needed for *course-changing* (turning) while *course-keeping*, that is:

$$\psi_d = \text{constant} \tag{8.126}$$

can be treated as a special case of turning. A simple 3rd-order filter for this purpose is; see Section 5.1:

$$\frac{\psi_d}{\psi_r}(s) = \frac{\omega_n^3}{(s+\omega_n)(s^2+2\zeta\omega_n s+\omega_n^2)} \tag{8.127}$$

where the reference ψ_r is the operator input, ζ is the relative damping ratio, and ω_n is the natural frequency. Notice that:

$$\lim_{t\to\infty} \psi_d(t) = \psi_r \tag{8.128}$$

and that $\dot{\psi}_d$ and $\ddot{\psi}_d$ are smooth and bounded for steps in ψ_r. This is the main motivation for choosing a 3rd-order model.

In many cases it is advantageous to limit the desired yaw rate $|r_d| \leq r_{\max}$ during turning. This can be done by including a saturating element in the reference model (see Van Amerongen 1982, Van Amerongen 1984). The yaw acceleration $a_d = \ddot{\psi}_d$ can also be limited such that $|a_d| \leq a_{\max}$ by using a second saturating element. The resulting state-space model including velocity and acceleration saturating elements becomes:

$$\dot{\psi}_d = \text{sat}(r_d) \tag{8.129}$$

$$\dot{r}_d = \text{sat}(a_d) \tag{8.130}$$

$$\dot{a}_d = -(2\zeta+1)\omega_n \text{sat}(a_d) - (2\zeta+1)\omega_n^2 \text{sat}(r_d) + \omega_n^3(\psi_r - \psi_d) \tag{8.131}$$

The saturating element is defined as:

$$\text{sat}(x) = \begin{cases} \text{sgn}(x)x_{\max} & \text{if } |x| \geq x_{\max} \\ x & \text{else} \end{cases} \tag{8.132}$$

The autopilot reference model has been simulated in Matlab™ with yaw rate limitation

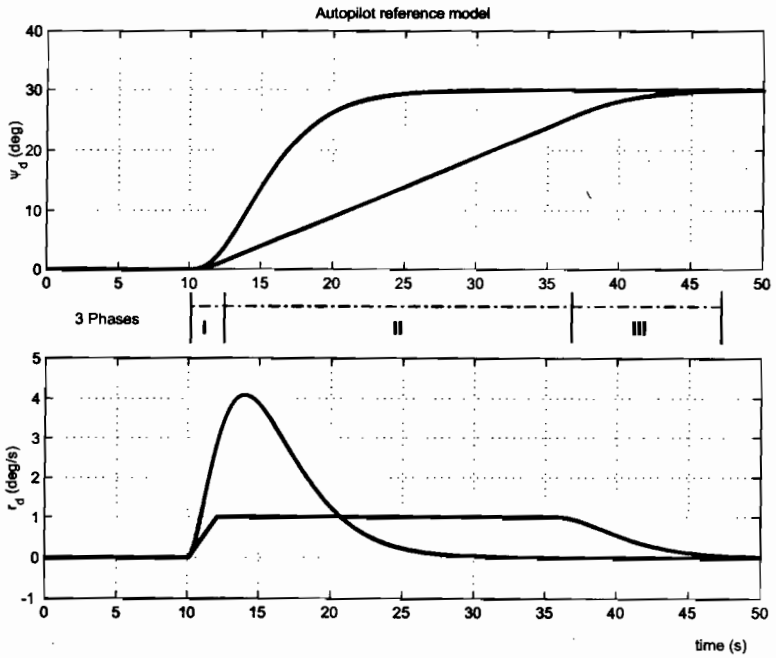


Figure 8.14: The plots show the effect of including a rate limitator of $r_{\max} = 1$ deg in a 3rd-order reference model for heading. Notice that r_d becomes very high in the linear case while ψ_d looks satisfactory in both cases.

$r_{\max} = 1.0$ (deg/s), acceleration limit $a_{\max} = 0.5$ (deg/s^2), and command $\psi_r = 30$ (deg). The results are shown in Figure 8.14. Notice that the unlimited (linear) case yields unsatisfactory high values for r_d.

The main motivation for using a rate limiting element in the reference model is that the course-changing maneuver will be described by three phases (positive turn):

I Start of turn, acceleration ($r_d > 0$ and $0 < \dot{r}_d \leq a_{\max}$)

II Steady turning ($r_d = r_{\max}$ and $\dot{r}_d = 0$)

III End of turn, deceleration ($r_d > 0$ and $-a_{\max} \leq \dot{r}_d < 0$)

The same three phases applies to negative turns but with opposite signs on r_d and a_d. The three phases are advantageous when performing a large change in course. The effect of a saturating element and nonlinear damping in a reference model are also demonstrated in Example 5.1 in Section 8.1.

A more sophisticated method for generating heading reference signals could be to use optimization techniques to compute the desired yaw angle, but then at the expense of a more complicated software algorithm to be implemented in real time.

8.4.2 Conventional PID-Control

The autopilot systems of *Sperry* and *Minorsky* were both single-input single-output (SISO) control systems where the heading (yaw angle) of the ship was measured by a gyro compass; see Section 1.1. Today, this signal is fed back to a computer in which a PID-control system (autopilot) is implemented in software. The autopilot compares the operator set-point (desired heading) with the measured heading and computes the rudder command which is then transmitted to the rudder servo for corrective action.

The main difference between the autopilot systems of Sperry and Minorsky and the modern autopilot is the increased functionality that has been added with sophisticated features like:

- Wave filtering; avoids 1st-order wave disturbances being fed back to the actuators; see Section 6.1.
- Adaptation to varying environmental conditions, shallow water effects and time-varying model parameters, e.g. changes in mass and centre of gravity.
- Wind feedforward for accurate and rapid course-changing maneuvers.
- Reference feedforward using a dynamic model, ψ_d, r_d, and $\dot{r}_d$, for course changing maneuvers. Course-keeping is obtained by using a constant reference signal, $\psi_d =$ constant, as input to the reference model; see Section 8.4.1.

Full State Feedback

Assume that both ψ and r are measured by using a compass and a rate gyro. A PID-controller for full state feedback is (see Section 7.1):

$$\tau_N(s) = \tau_{FF}(s) \underbrace{-K_p \left(1 + T_d s + \frac{1}{T_i s}\right) \tilde{\psi}(s)}_{\tau_{PID}} \tag{8.133}$$

where τ_N is the controller yaw moment, τ_{FF} is a feedforward term to be decided, $\tilde{\psi} = \psi - \psi_d$ is the heading error and:

$$\begin{aligned} K_p &> 0 \quad \text{proportional gain constant} \\ T_d &> 0 \quad \text{derivative time constant} \\ T_i &> 0 \quad \text{integral time constant} \end{aligned}$$

A continious-time representation of the controller is:

$$\tau_N(t) = \tau_{FF} - K_p \tilde{\psi} - \underbrace{K_p T_d}_{K_d} \tilde{r} - \underbrace{K_p / T_i}_{K_i} \int_0^t \tilde{\psi}(\tau) d\tau \tag{8.134}$$

where $\tilde{r} = r - r_d$ and $\tilde{\psi} = \psi - \psi_d$. The controller gains can be found by pole placement, see Section 7.1.3. This gives:

$$\begin{aligned} K_p &= T\omega_n^2 \\ K_d &= 2\zeta\omega_n T - 1 \\ K_i &= \frac{\omega_n}{10} K_p \end{aligned}$$

where ζ is the relative damping ratio and ω_n is the natural frequency of the closed-loop system.

Consider Nomoto's 1st-order model (8.29) in Section 8.1.2 with two rudder inputs δ_1 and δ_2:

$$T\dot{r} + r = \underbrace{K(\delta_1 + \delta_2) + \tau_{\text{wind}}}_{\tau_N} \tag{8.135}$$

where τ_{wind} represents the wind moment. In cases where wind is a measured term, τ_{wind} can be cancelled directly by the controller. If τ_{wind} is unknown, then the integral term in the controller must compensate for the slowly-varying wind loads.

Optimal *control allocation* for a ship equipped with two equal rudders implies that:

$$\begin{aligned} &\text{Wind feedforward:} && \delta_1 = \delta_2 = 0.5\left(\tfrac{\tau_N - \tau_{\text{wind}}}{K}\right) \\ &\text{Integral action compensates for wind:} && \delta_1 = \delta_2 = 0.5\left(\tfrac{\tau_N}{K}\right) \end{aligned} \tag{8.136}$$

A more general discussion on control allocation is found in Section 7.5.

The *feedforward* term τ_{FF} in (8.133) is determined such that perfect tracking during course-changing maneuvers is obtained. Using Nomoto's 1st-order model (8.135) as basis for feedforward, suggests that *reference feedforward* should be included according to:

$$\tau_{FF} = T\dot{r}_d + r_d \tag{8.137}$$

Substituting (8.137) and (8.133) into (8.135), the error dynamics becomes:

$$T\ddot{e} + \dot{e} = \tau_{\text{PID}} \tag{8.138}$$

where $e = \psi - \psi_d$. Since this system is linear, the closed system can be analyzed in the frequency plane by using *Bode* plots. Define the transfer function:

$$h(s) = \frac{e}{\tau_{\text{PID}}}(s) = \frac{1}{s(1+Ts)} \tag{8.139}$$

and let (see Figure 8.15):

$$\begin{aligned} h_{\text{PID}}(s) &= K_p\left(1 + T_d s + \frac{1}{T_i s}\right) \\ &= K_p \frac{T_i T_d s^2 + T_d s + 1}{T_i s} \end{aligned} \tag{8.140}$$

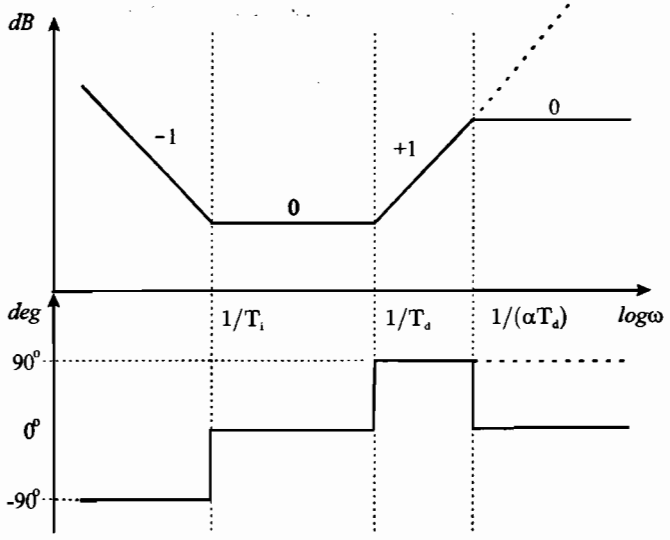


Figure 8.15: Bode plot showing asymptotic curves for the PID-controller with (solid) and without (dotted) limited differentiation.

Hence, the loop transfer function becomes:

$$\begin{aligned} l(s) &= h(s)h_{\text{PID}}(s) \\ &= \frac{K_p}{T_i}\frac{(T_iT_ds^2+T_ds+1)}{s^2(1+Ts)} \end{aligned} \tag{8.141}$$

A frequently used approximation for (8.140) is found by assuming that $T_i \gg T_d$ such that $T_i \approx T_i + T_d$. Hence:

$$\begin{aligned} h_{\text{PID}}(s) &= K_p\left(1+T_ds+\frac{1}{T_is}\right) \\ &\approx K_p\frac{1+(T_i+T_d)s+T_dT_is^2}{T_is} \\ &= K_p\frac{(1+T_is)(1+T_ds)}{T_is} \end{aligned} \tag{8.142}$$

This expression is particular useful when sketching a *Bode diagram* using asymptotic approximations, see Figure 8.15.

Matlab:
The transfer function (8.140) can be plotted by using the command:

```
bode(Kp.*[Ti*Td,Td,1],[Ti,0])
```

Output Feedback using Only Compass Measurements

In many cases ships are only equipped with a gyrocompass for feedback control. Hence, the D-term in the controller must be replaced with a limited differentiator:

$$r(s) \approx \frac{T_d s}{1 + \alpha T_d s} \psi(s), \quad 0 < \alpha \ll 1 \tag{8.143}$$

such that the high-frequency components of $\psi(s)$ are filtered out. If we apply the low-pass filter:

$$h_{\mathrm{LP}}(s) = \frac{1}{1 + \alpha T_d s} \tag{8.144}$$

to all terms in the PID-controller, (8.142) takes the form:

$$h_{\mathrm{PID}}(s) = K_p \frac{(1 + T_i s)(1 + T_d s)}{T_i s (1 + \alpha T_d s)}$$

The controller can be implemented in the time-domain as:

$$\tau_{\mathrm{N}} = \tau_{\mathrm{FF}} - K_p \tilde{\psi}_{\mathrm{LP}} - \underbrace{K_p T_d}_{K_d} \tilde{r}_{\mathrm{LP}} - \underbrace{K_p / T_i}_{K_i} \int_0^t \tilde{\psi}_{\mathrm{LP}}(\tau) d\tau \tag{8.145}$$

with two filters:

$$\tilde{\psi}_{\mathrm{LP}}(s) = \frac{1}{1 + \alpha T_d s} \tilde{\psi}(s), \qquad \tilde{r}_{\mathrm{LP}}(s) = \frac{T_d s}{1 + \alpha T_d s} \tilde{\psi}(s) \tag{8.146}$$

The parameter $0 < \alpha < 1$ is usually chosen to 0.1 while $T_i = 10\, T_d$, such that:

$$\frac{1}{T_i} \ll \frac{1}{T_d} \ll \frac{1}{\alpha T_d} \tag{8.147}$$

This is illustrated in Figure 8.15.

Example 8.10 (Experimental testing of Car Carrier Autopilot)
In Figure 8.16 a scale model of MV Autoprestige of the United European Car Carriers (UECC) is shown. UECC is transporting vehicles for the global automotive industry.

The mode scale is $S = 21.6$ *and the length of the ship is* $L_s = 128.8$ *(*m*). Hence, the length of the model becomes:*

$$L_m = L_s / S = 5.96(\,\mathrm{m}) \tag{8.148}$$

The maneuvering tests of the vessel were performed in the Ocean Basin at MARINTEK in Trondheim, see Figure 8.2. This is a large testing facility where irregular waves can be generated by using wave makers. Also current generators and wind fans can be used to produce realistic conditions. The experiments were performed at service speed $U_m = 2.3$ *(*m/s*) corresponding to:*

$$U_s = \sqrt{S} U_m = 10.7\ (\mathrm{m/s}) = 20.8\ (\textit{knots}) \tag{8.149}$$

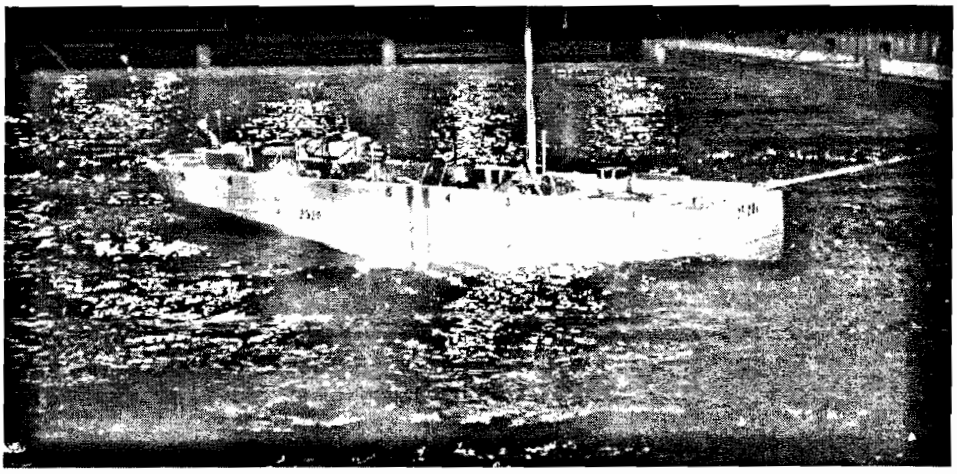


Figure 8.16: Model of the MV Autoprestige car carrier scale $S = 21.6$. Courtesy to MARINTEK and the United European Car Carriers (UECC).

in full scale. The wave makers were configured to produce regular waves corresponding to Sea State Codes 4 and 5 according to:

sea state	*wave direction* β	*significant wave height* $H_{1/3}$	*wave period* T_0	*wave frequency* $\omega_0 = \frac{2\pi}{T_0}$	*frequency of encounter* $\omega_e = \omega_0 - \frac{\omega_0^2}{g} U_s \cos\beta$
4 - bow sea	$-135°$	1.3 m	8.5 s	0.70 rad/s	1.08 rad/s
5 - quatering sea	$50°$	3.5 m	9.0 s	0.74 rad/s	0.36 rad/s

Only course-keeping was tested. The wave direction is defined in Section 4.2.3. The desired heading was chosen as ψ_d = constant *with* $\dot{r}_d = r_d = 0$ *in the two experiments. Only a PD-controller was needed because the size of the basin limited each test to approximately* 12 *(s) which is not enough for the integrator to converge. Hence for sea-keeping, the Ocean Basin autopilot reduces to:*

$$\tau_N = -K_p(\hat{\psi} - \psi_d) - K_p T_d \hat{r} \tag{8.150}$$

$$\delta_1 = \delta_2 = \frac{1}{2}\left(\frac{\tau_N}{K}\right) \tag{8.151}$$

where $\hat{\psi}$ *and* $\hat{r}$ *are computed using the passive wave filter described in Section 6.1.5; see Example 6.5. The inputs to the wave filter are position and attitude camera measurements and a Seatex MRU-6 motion unit, which are integrated in a state observer.*

The experimental results for bow and following seas are shown in Figure 8.18–8.19. It is seen that the ship suffers from heavy rolling due to lateral incoming waves. Even though this is an extreme situation, the autopilot performs satisfactorily, thanks to the wave filter. It is important to notice that the estimated LF component of the heading $\hat{\psi}$ *tracks the desired heading under relatively large WF motions both in roll and yaw. This also reduces the rudder*

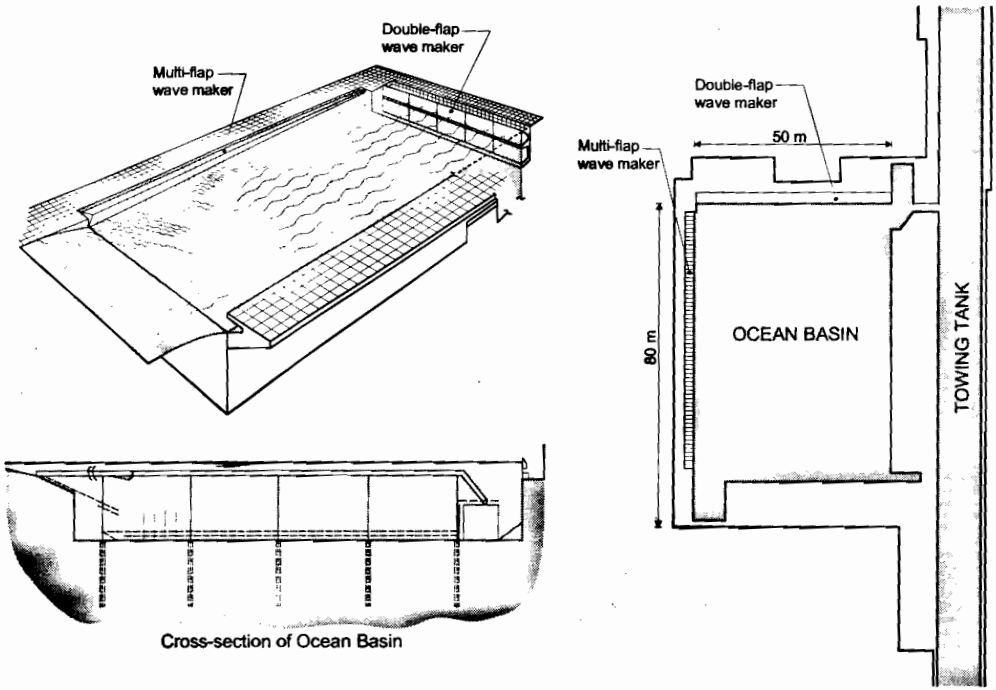


Figure 8.17: The Ocean Basin at MARINTEK, Trondheim. Courtesy to MARINTEK.

action significantly. It is also seen that the ship's rolling motion for following seas, which is a well known phenomenon.

The test was also performed without wave filtering using $y_1 = \psi + \psi_w$ *and* $y_2 = r + r_w$ *directly in the feedback controller. The result was bang-bang control at maximum allowed rudder amplitude resulting in performance degradation in yaw. Poor performance in yaw will induce larger roll amplitudes which was clearly seen in the experiments. From this it can be concluded that all commercial autopilot system should include a wave filter to reduce rolling and to obtain accurate course control.*

8.4.3 PID Control including Acceleration Feedback

The autopilot system of Section 8.4.2 can be extended to include acceleration feedback (see Section 7.1.3) by differentiating the output of a yaw rate gyro r_{gyro} according to:

$$\dot{r} \approx \frac{s}{\omega_f + s} r_{\text{gyro}} \tag{8.152}$$

The filter frequency ω_f must, however, be larger than the control bandwidth ω_b. In most cases this is easy to satisfy since ω_f can be chosen as high as 10–50 Hz (or 63–314 rad/s) if an accurate yaw rate gyro is applied. A discrete-time representation of the filter (8.152) is found in Appendix B.3. This is particular useful for smaller vessels, which are more vulnerable to environmental disturbances than large vessels. The main idea is to increase the moment of inertia by yaw rate feedback, such that external disturbances are suppressed; see Section 7.1.3. Consider the controller:

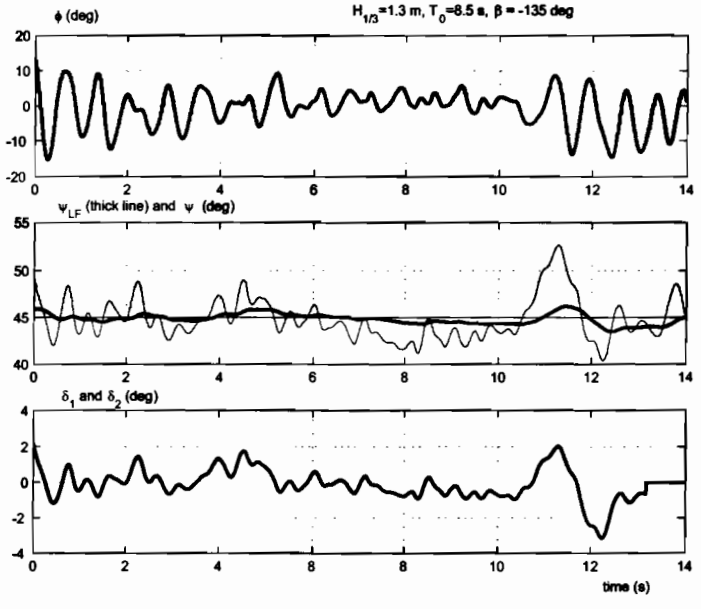


Figure 8.18: Experimental results for the car carrier in Sea State 4. Courtesy to UECC.

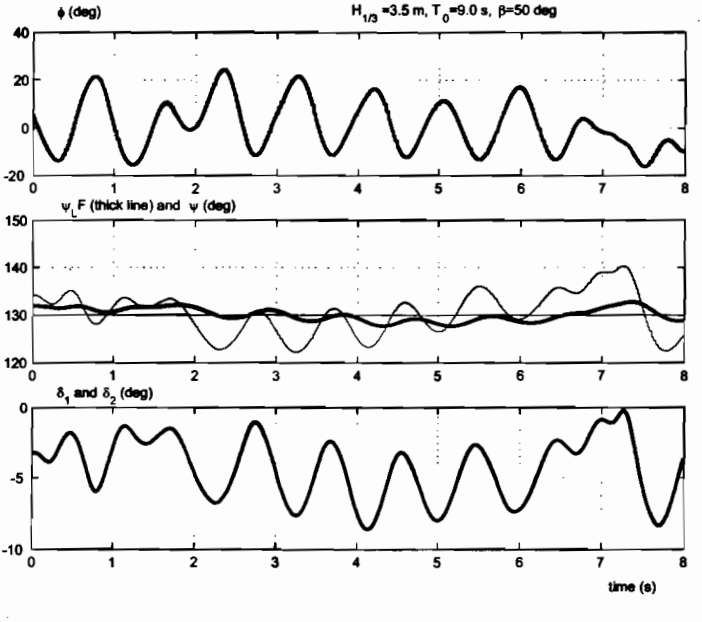


Figure 8.19: Experimental results for the car carrier in Sea State 5. Courtesy to UECC.

$$\tau_{\rm N} = \tau_{\rm FF} \underbrace{-K_p\tilde{\psi} - K_d\tilde{r} - K_i\int_0^t \tilde{\psi}(\tau)d\tau}_{\rm PID} \underbrace{-K_m\dot{r}}_{\substack{\text{acceleration}\\ \text{feedback}}} \tag{8.153}$$

$$\tau_{\rm FF} = (T + K_m)\dot{r}_d + r_d \tag{8.154}$$

Notice that the term K_m must be included in $\tau_{\rm FF}$ correspondingly. Substituting this expression into (8.135) yields the closed-loop error dynamics:

$$(T + K_m)\ddot{e} + (1 + K_d)\dot{e} + K_p e + K_i\int_0^t e(\tau)d\tau = 0 \tag{8.155}$$

This suggests the following pole placement algorithm (see Section 7.1.3):

Autopilot Pole Placement Algorithm (PID and Acceleration Feedback)

1. Specify the bandwidth $\omega_b > 0$ and the relative damping ratio $\zeta > 0$
2. Compute the natural frequency: $\omega_n = \frac{1}{\sqrt{1-2\zeta^2+\sqrt{4\zeta^4-4\zeta^2+2}}}\omega_b$
3. Specify the gain: $K_m \geq 0$ (optional acceleration feedback)
4. Compute the P-gain: $K_p = (T + K_m)\omega_n^2$
5. Compute the D-gain: $K_d = 2\zeta\omega_n(T + K_m) - 1$
6. Compute the I-gain: $K_i = \frac{\omega_n}{10}K_p$

A final implementation issue is the problem of 1st-order wave-induced disturbances. Using a wave filter for ψ, r, and, $\dot{r}$ is recommended if all these signals are used in feedback. Wave filtering for systems using velocity and acceleration feedback is discussed by Lindegaard and Fossen (2001*a*) , and Lindegaard (2003).

8.4.4 PID Control including Wind Feedforward

The main result of Section 8.4.2 was a PID-controller:

$$\tau_{\rm N}(s) = \tau_{\rm FF}(s) \underbrace{-K_p\left(1 + T_d s + \frac{1}{T_i s}\right)\tilde{\psi}(s)}_{\tau_{\rm PID}} \tag{8.156}$$

with reference feedforward:

$$\tau_{\rm FF} = T\dot{r}_d + r_d \tag{8.157}$$

For a ship with two simultaneously controlled rudders the rudder commands are computed as:

$$\delta_1 = \delta_2 = \frac{1}{2}\left(\frac{\tau_{\rm N} - \tau_{\rm wind}}{K}\right) \tag{8.158}$$

where

$$\tau_{\rm wind} = [X_{\rm wind}, Y_{\rm wind}, N_{\rm wind}]^\top \tag{8.159}$$

is a vector of three nonlinear functions; $X_{\text{wind}} = X_{\text{wind}}(V_r, \gamma_r), Y_{\text{wind}} = Y_{\text{wind}}(V_r, \gamma_r)$, and $N_{\text{wind}} = N_{\text{wind}}(V_r, \gamma_r)$ of relative wind speed, V_r, and direction, γ_r, respectively. It is straightforward to measure V_r and γ_r using a conventional wind sensor. These measurements are usually low-pass filtered and used as inputs to a wind load model for the generation of forces and moments; see Section 4.1.

In cases where wind feedforward is omitted ($\tau_{\text{wind}} = 0$), integral action can be used to compensate for wind loads. The drawback is that the integrator needs several minutes to remove a large wind component during the start-up of an autopilot system. Integral action works fairly well during fixed heading (station-keeping and transit) while in a maneuvering situation large course deviations might be expected. Consequently, it is advantageous to implement wind feedforward to reduce the loads on the integrator and to obtain maximum performance during start-up and in maneuvering situations.

8.4.5 Linear Quadratic Optimal Control

The ship autopilot problem can be defined as a linear quadratic optimization problem:

$$J = \min_{\delta} \left\{ \frac{\alpha}{T} \int_0^T [e^2 + \lambda_1 r^2 + \lambda_2 \delta^2] d\tau \right\} \tag{8.160}$$

where α is a constant to be interpreted later, $e = \psi_d - \psi$ is the heading error, δ is the actual rudder angle, and λ_1 and λ_2 are two factors weighting the cost of heading errors e and heading rate r against the control effort δ.

Sailing in restricted waters usually requires accurate control, while the minimization of fuel consumption is more important in open seas. This can be obtained by changing the weights λ_1 and λ_2. We will discuss three criteria for control weighting.

The Steering Criterion of Koyama (1967)

The first criterion was derived by Koyama (1967) who observed that the ship's swaying motion y could be approximated by a sinusoid during autopilot control, that is:

$$y = \sin(et) \quad \Longrightarrow \quad \dot{y} = e\cos(et) \tag{8.161}$$

The length of one arch L_a of the sinusoid can be calculated as:

$$L_a = \int_0^{\pi} \sqrt{(1 + \dot{y}^2)} d\tau = \int_0^{\pi} \sqrt{[1 + e^2 \cos^2(e\tau)]} d\tau \approx \pi \left(1 + \frac{e^2}{4}\right) \tag{8.162}$$

Hence, the relative elongation due to a sinusoidal course error is:

$$\frac{\Delta L}{L} = \frac{L_a - L}{L} = \frac{\pi(1 + e^2/4) - \pi}{\pi} = \frac{e^2}{4} \tag{8.163}$$

This suggests that the percentage loss of speed during course control can be calculated by using the elongation in distance due to a sinusoidal course error. Consequently, Koyama (1967) proposed minimizing the speed loss term $e^2/4$ against the increased resistance due to steering given by the term δ^2. Consequently:

$$J = \min_{\delta} \left\{ 100 \left(\frac{\pi}{180}\right)^2 \frac{1}{T} \int_0^T \left[\frac{e^2}{4} + \lambda_2 \delta^2\right] d\tau \approx \frac{0.0076}{T} \int_0^T [e^2 + \lambda_2 \delta^2] d\tau \right\} \quad (8.164)$$

In this context (8.160) can be interpreted as:

$$J = \text{loss of speed (\%)} \quad (8.165)$$

$$\alpha = 0.0076 \quad (8.166)$$

Notice that $\lambda_1 = 0$ in this analysis. In practice it might be desirable to penalize r^2 by choosing $\lambda_1 > 0$. Koyama suggested a λ_2-factor of approximately 8–10. Experiments show that such high values for λ_2 avoids large rudder angles, and thus high turning rates. Therefore, $\lambda_2 = 10$ will be a good choice in bad weather, where it is important to suppress high frequency rudder motions.

The Steering Criterion of Norrbin (1971)

Another approach for computation of λ_2 was proposed by Norrbin (1972) . Consider the surge equation (8.7)–(8.8) in the form:

$$(m - X_{\dot{u}})\dot{u} = X_{|u|u}|u|u + (1-t)T + T_{\text{loss}} \quad (8.167)$$

where:

$$T_{\text{loss}} = (m + X_{vr})vr + X_{cc\delta\delta} c^2 \delta^2 + (X_{rr} + mx_g)r^2 + X_{\text{ext}} \quad (8.168)$$

Norrbin (1972) suggested minimizing the loss term T_{loss} to obtain maximum forward speed u. Consequently, the controller should minimize the centripetal term vr, the square rudder angle δ^2 and the square heading rate r^2 while the unknown disturbance term X_{ext} is neglected in the analysis. The assumptions in doing this are as follows:

1. The sway velocity v is approximately proportional to r. Combining (8.26) and (8.27), yields:

$$v(s) = \frac{K_v(1 + T_v s)}{K(1 + Ts)} r(s) \approx \frac{K_v}{K} r(s) \quad (8.169)$$

   under the assumption that $T_v \approx T$. Hence, the centripetal term vr will be approximately proportional to the square of the heading rate, that is $vr \approx (K_v/K)r^2$

2. The ship's yawing motion is periodic (sinusoid) under autopilot control such that

$$r_{\max} = \omega_r \, e_{\max} \quad (8.170)$$

   where ω_r is the frequency of the sinusoidal yawing.

These two assumptions suggest that the loss term T_{loss} can be minimized by minimizing e^2 and δ^2 which is the same result obtained in Koyama's analysis. The only difference between the criteria of Norrbin and Koyama is that the λ_2-values arising from Norrbin's approach will be different when computed for the same ship. The performance of the controller also depends on the sea state. This suggests that a trade-off between the λ_2-values proposed by Koyama and Norrbin could be made according to:

$$\underbrace{\text{(calm sea)}}_{\text{Norrbin}} \qquad 0.1 \leq \lambda_2 \leq 10 \qquad \underbrace{\text{(rough sea)}}_{\text{Koyama}} \tag{8.171}$$

The Steering Criterion of Van Amerongen and Van Nauta Lemke (1978)

Experiments with the steering criteria of Koyama and Norrbin soon showed that the performance could be further improved by considering the squared yaw rate r^2, in addition to e^2 and δ^2 (Van Amerongen and Van Nauta Lemke 1978). Consequently, the following criterion was proposed:

$$J = \min_{\delta} \left\{ \frac{0.0076}{T} \int_0^T (e^2 + \lambda_1 r^2 + \lambda_2 \delta^2) d\tau \right\} \tag{8.172}$$

For a tanker and a cargo ship, Van Amerongen and Van Nauta Lemke (1978, 1980) gave the following values for the weighting factors λ_1 and λ_2 corresponding to the data set of Norrbin (1972):

tanker:	$L =$	300 m	$\lambda_1 =$	15.000	$\lambda_2 =$ 8.0
cargo ship:	$L =$	200 m	$\lambda_1 =$	1.600	$\lambda_2 =$ 6.0

The solution of the optimal steering criteria is found by considering Nomoto's 1st order model in the form:

$$\dot{\psi}' = r' \tag{8.173}$$

$$T'\dot{r} + (U/L)r = (U/L)^2 K' \delta \tag{8.174}$$

Straightforward application of optimal control theory to the criterion of Van Amerongen and Van Nauta Lempke (1978), yields (see Section 7.2):

$$\delta = K_p(\psi_d - \psi) - K_d r \tag{8.175}$$

where (see Exercise 8.6):

$$K_p = \sqrt{\frac{1}{\lambda_2}} \tag{8.176}$$

$$K_d = \frac{L}{U} \frac{\sqrt{1 + 2K_p K'T' + K'^2 (U/L)^2 \, (\lambda_1/\lambda_2)} - 1}{K'} \tag{8.177}$$

Consequently, the solution of the criteria of Koyama and Norrbin is obtained by setting $\lambda_1 = 0$ and $\lambda_2 = \lambda$ which yields:

$$K_p = \sqrt{\frac{1}{\lambda}} \tag{8.178}$$

$$K_d = \frac{L}{U} \frac{\sqrt{1 + 2K_p K'T'} - 1}{K'} \tag{8.179}$$

From these expressions it is seen that K_p depends on the weighting factor λ, while K_d depends on K_p as well as the model parameters K' and T'. Hence, accurate steering requires that K' and T' are known with sufficient accuracy.

An extension to Nomoto's 2nd-order model is found by considering (see Section 8.1.2):

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{8.180}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{8.181}$$

where $\mathbf{x} = [v, r, \psi]^\top$, $\mathbf{u} = \delta$ and:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix}, \quad \mathbf{C} = [0, 1, 1] \tag{8.182}$$

Let $\mathbf{x}_d = [0, 0, \psi_d]^\top$ and:

$$\begin{aligned} \mathbf{e} &= \mathbf{y} - \mathbf{y}_d \\ &= \mathbf{C}(\mathbf{x} - \mathbf{x}_d) \end{aligned} \tag{8.183}$$

The steady-state optimal solution minimizing the quadratic performance index (assuming $\mathbf{y}_d = constant$):

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T (\mathbf{e}^\top \mathbf{Q}\mathbf{e} + \mathbf{u}^\top \mathbf{R}\mathbf{u})\, d\tau \right\} \tag{8.184}$$

where $\mathbf{Q} = \text{diag}\{0, q_{22}, q_{33}\} \geq 0$ and $\mathbf{R} = r_{11} > 0$ are the weights is (see Section 7.2):

$$\mathbf{u} = \mathbf{G}_1 \mathbf{x} + \mathbf{G}_2 \mathbf{y}_d \tag{8.185}$$

where:

$$\mathbf{G}_1 = -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty \tag{8.186}$$

$$\mathbf{G}_2 = -\mathbf{R}^{-1}\mathbf{B}^\top (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-T} \mathbf{C}^\top \mathbf{Q} \tag{8.187}$$

and $\mathbf{P}_\infty$ is the solution of the matrix Riccati equation:

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_\infty + \mathbf{C}^\top \mathbf{Q}\mathbf{C} = \mathbf{0} \tag{8.188}$$

The robustness of optimal autopilots for course-keeping control with state estimator is analyzed in Holzhüter (1992).

Example 8.11 (Experimental Results with an Optimal Autopilot)
An optimal autopilot using the criterion of Van Amerongen and Van Nauta Lemke (1978) has been implemented and tested experimentally. The vessel considered is the "Nordmann Borg" supply vessel shown in Figure 8.20. The performance of the heading controller with reference feedforward is shown in Figure 8.21. The upper plot shows the course changing capabilities for different autopilot inputs, while the lower plot is a zoom showing the step responses during the interval $t \in [1950, 2350]$ *(s).*

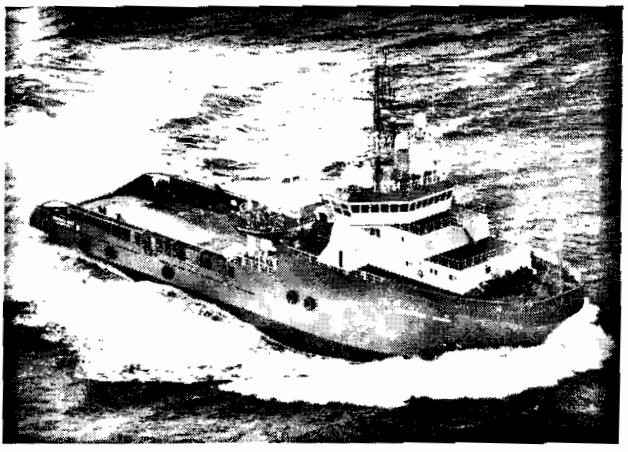


Figure 8.20: The Nordmand Borg Supply Vessel. Courtesy to Solstad Shipping.

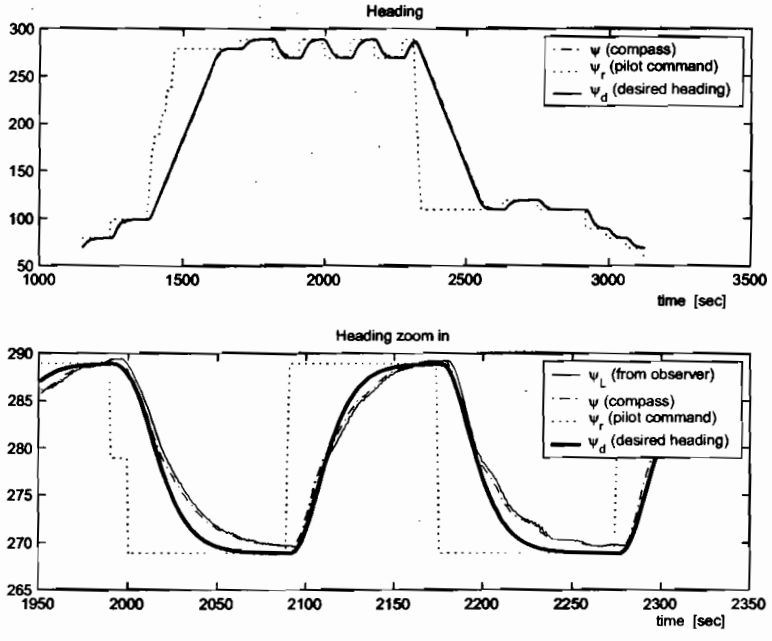


Figure 8.21: Heading as a function of time. A reference model is used to avoid steps during course-changing.

8.4.6 State Feedback Linearization

Full state feedback implies that both $y_1 = \psi + \psi_w$ and $y_2 = r + r_w$ are measured. For systems where r is estimated in a state observer by using the yaw angle measurement y_1, an output feedback controller must be designed, see Section 8.4.10. Even though both y_1 and y_2 are measured, a wave (notch) filter is needed in order to reduce wear and tear of the steering machine; see Section 6.1.

State Feedback Linearization

Consider the nonlinear autopilot model of Norrbin (1963) in Section 8.1.4:

$$T\dot{r} + H_{\mathrm{N}}(r) = \underbrace{K\delta + \tau_{\mathrm{wind}}}_{\tau_{\mathrm{N}}} \tag{8.189}$$

where τ_{wind} represents wind loads and:

$$H_{\mathrm{N}}(r) = n_3 r^3 + n_2 r^2 + n_1 r + n_0 \tag{8.190}$$

A state feedback linearizing control law with wind feedforward for this system can be designed according to (Fossen 1993*b*):

$$\delta = \frac{\tau_{\mathrm{N}} - \tau_{\mathrm{wind}}}{K} \tag{8.191}$$

where

$$\tau_{\mathrm{N}} = Ta + n_3 r^3 + n_2 r^2 + n_1 r \tag{8.192}$$

$$a = \ddot{\psi}_d - K_p\tilde{\psi} - K_d\dot{\tilde{\psi}} - K_i \int_0^t \tilde{\psi}(\tau)d\tau \tag{8.193}$$

Here a can be interpreted as the commanded angular acceleration in yaw. For a more detailed discussion on feedback linearization, see Section 7.3. Notice that the constant term n_o in (8.190) due to an rudder angle offset is not compensated for in τ_{N} since a includes integral action. Therefore, the integral part of the controller will remove the constant term n_o, and possibly τ_{wind} if wind feedforward is omitted. Wind feedforward is, however, advantageous since it might take a long time to integrate up an estimate of the wind load in bad weather, that is, for large values of τ_{wind}. This again yields poor performance when starting the autopilot system (cold start).

The resulting error dynamics become:

$$T\ddot{e} + (1 + K_d)\dot{e} + K_p e + K_i \int_0^t e(\tau)d\tau = 0 \tag{8.194}$$

indicating how the controller gains K_d, K_p, and K_i should be chosen. Extensions to adaptive feedback linearization is found in Fossen and Paulsen (1992), while feedback linearization with saturating and slew rate limiting actuators is discussed by Tzeng *et al.* (1999).

8.4.7 Adaptive Feedback Linearization and Optimality

An adaptive optimal course-keeping autopilot can be derived by means of Lyapunov stability theory (Fossen and Paulsen 1992). Consider the nonlinear ship steering dynamics (8.189) in the form:

$$T\dot{r} + \boldsymbol{\phi}_0^\top(r)\boldsymbol{\theta}_0 = \tau_N \tag{8.195}$$

where

$$\boldsymbol{\phi}_0(r) = [r^3, r^2, r, 1]^\top \tag{8.196}$$

$$\boldsymbol{\theta}_0 = [n_3, n_2, n_1, n_0]^\top \tag{8.197}$$

Here $\boldsymbol{\phi}_0(r)$ is the regressor vector (known) and $\boldsymbol{\theta}_0$ is a vector of unknown parameters to be estimated on-line.

Feedback Linearization and Optimality

The model (8.195) can be transformed to a linear 2nd-order system:

$$\ddot{\psi} = a \tag{8.198}$$

by choosing the feedback linearizing controller as:

$$\tau_N = Ta + \boldsymbol{\phi}_0^\top(r)\boldsymbol{\theta}_0 \tag{8.199}$$

where a can be interpreted as the *commanded acceleration*; see Section 7.3. Hence, this can be formulated as an optimal control problem with performance criterion:

$$J = \min_a \left\{ \int_0^T \left[(\psi_d - \psi)^2 + \lambda_1 \dot{\psi}^2 + \lambda_2 a^2 \right] d\tau \right\} \tag{8.200}$$

Here the tracking error $\psi_d - \psi$ is penalized together with the yawing rate $\dot{\psi}$ and the commanded acceleration a with factors λ_1 and λ_2, respectively. Solving this LQ problem for the system model $\ddot{\psi} = a$, yields the following steady-state solution for the optimal commanded acceleration (see Section 7.2.1):

$$a = K_p(\psi_d - \psi) - K_d\dot{\psi} \tag{8.201}$$

where:

$$K_p = \sqrt{\frac{1}{\lambda_2}}, \qquad K_d = \sqrt{2\sqrt{\frac{1}{\lambda_2}} + \frac{\lambda_1}{\lambda_2}} \tag{8.202}$$

Notice that (8.201) does not depend on the unknown system parameters T and $\boldsymbol{\theta}$.

Parameter Adaptation

The adaptive control law can be chosen as:

$$\begin{aligned} \tau_N &= \hat{T}a + \boldsymbol{\phi}_0^\top(r)\hat{\boldsymbol{\theta}}_0 \\ &= \boldsymbol{\phi}^\top(r)\hat{\boldsymbol{\theta}} \end{aligned} \tag{8.203}$$

where the hat denotes the parameter estimates and:

$$\boldsymbol{\phi}(r) = [a, \boldsymbol{\phi}_0^\top(r)]^\top, \quad \hat{\boldsymbol{\theta}} = [\hat{T}, \hat{\boldsymbol{\theta}}_0^\top]^\top \tag{8.204}$$

Let us define the parameter estimation errors as $\tilde{T} = \hat{T} - T$ and $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$. Consequently, the closed-loop dynamics can be written:

$$T\,[\ddot{\psi} - a] = \boldsymbol{\phi}^\top(r)\tilde{\boldsymbol{\theta}} \tag{8.205}$$

Substituting the optimal controller (8.201) into this expression, yields:

$$T\,[\ddot{\psi} + K_d\dot{\psi} + K_p(\psi - \psi_d)] = \boldsymbol{\phi}^\top(r)\tilde{\boldsymbol{\theta}} \tag{8.206}$$

It is then clear that optimality with respect to (8.200) requires that $\tilde{\boldsymbol{\theta}} = \mathbf{0}$ (no parametric uncertainties). With these goals in mind a parameter estimator can be derived by applying Lyapunov stability theory. Let the closed-loop dynamics be written in abbreviated form as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}\frac{1}{T}\boldsymbol{\phi}^\top(r)\tilde{\boldsymbol{\theta}} \tag{8.207}$$

where $\mathbf{x} = [\psi - \psi_d, \dot{\psi}]^\top$ is the state vector and:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_d \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \tilde{\boldsymbol{\theta}} = [\tilde{T}, \tilde{\boldsymbol{\theta}}_0^\top]^\top \tag{8.208}$$

The parameter updating mechanism for $\hat{\boldsymbol{\theta}}$ can now be derived by using the Lyapunov function candidate:

$$V(\mathbf{x}, \tilde{\boldsymbol{\theta}}) = \mathbf{x}^\top\mathbf{P}\mathbf{x} + \frac{1}{T}\tilde{\boldsymbol{\theta}}^\top\boldsymbol{\Gamma}^{-1}\tilde{\boldsymbol{\theta}} \tag{8.209}$$

where $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^\top > 0$ is the adaptation gain. Differentiating V with respect to time, gives:

$$\begin{aligned} \dot{V} &= \mathbf{x}^\top\mathbf{P}\dot{\mathbf{x}} + \dot{\mathbf{x}}^\top\mathbf{P}\mathbf{x} + \frac{2}{T}\tilde{\boldsymbol{\theta}}^\top\boldsymbol{\Gamma}^{-1}\dot{\tilde{\boldsymbol{\theta}}} \\ &= \mathbf{x}^\top(\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A})\mathbf{x} + \frac{2}{T}\tilde{\boldsymbol{\theta}}^\top(\boldsymbol{\Gamma}^{-1}\dot{\tilde{\boldsymbol{\theta}}} + \boldsymbol{\phi}(r)\mathbf{b}^\top\mathbf{P}\mathbf{x}) \end{aligned} \tag{8.210}$$

Choosing the parameter update law as:

$$\dot{\hat{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma}\boldsymbol{\phi}(r)e \tag{8.211}$$

with $e = \mathbf{c}^\top\mathbf{x}$ and:

$$\begin{aligned} \mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} &= -\mathbf{Q} \\ \mathbf{P}\mathbf{b} &= \mathbf{c} \end{aligned} \tag{8.212}$$

where $\mathbf{P} = \mathbf{P}^\top > 0$ and $\mathbf{Q} = \mathbf{Q}^\top > 0$, gives:

$$\dot{V} = -\mathbf{x}^\top \mathbf{Q}\mathbf{x} \leq 0 \tag{8.213}$$

Notice that $\dot{\tilde{\boldsymbol{\theta}}} = \dot{\hat{\boldsymbol{\theta}}}$ since $\dot{\boldsymbol{\theta}} = \mathbf{0}$ (constant parameter assumption). Since this expression is only negative semi-definite (a quadratic error term $-\tilde{\boldsymbol{\theta}}^\top \tilde{\boldsymbol{\theta}}$ is missing in $\dot{V}$) only global convergence of $\mathbf{x} \to 0$ can be guaranteed. Lyapunov stability theory for autonomous systems ensures that $\psi \to \psi_d$ and $\dot{\psi} \to 0$ as $t \to \infty$, and that $\tilde{\theta}$ is bounded. It should be noted that $\tilde{\theta}$ will converge to zero only if the system is persistently excited. This is, however, not necessary for perfect tracking since $\psi \to \psi_d$ even though $\tilde{\boldsymbol{\theta}} \neq 0$.

Notice that the controller possesses integral action through the bias term n_0 which is estimated on-line using the element 1 in the regressor vector. This is a well known trick in adaptive control. In fact, the bias term n_0 will also include an estimate of wind, currents, and wave drift forces since these are slowly-varying disturbances which can be treated as a system bias.

8.4.8 Nonlinear Backstepping

The feedback linearizing controller can be extended to include nonlinear damping terms and to exploit good damping terms in the model by using backstepping designs. In this section only full state feedback is considered. The nonlinear backstepping controller can be designed by writing the autopilot model (8.56) in SISO strict feedback form (Fossen and Strand 1999*a*):

$$\dot{\psi} = r \tag{8.214}$$

$$m\dot{r} + d(r)r = \delta \tag{8.215}$$

where $m = T/K$ and $d(r) = \frac{1}{K} H_{\mathrm{N}}(r)$. The only nonlinearity in this model is due to the maneuvering characteristic $H_{\mathrm{N}}(r)$.

In Section 7.4.3 it was shown that the backstepping controller for this system is:

$$\delta = m\dot{\alpha}_1 + d(r)r - z_1 - k_2 z_2 - n_2(z_2)z_2 \tag{8.216}$$

$$\alpha_1 = r_d - [k_1 + n_1(z_1)]z_1 \tag{8.217}$$

where $k_1 > 0$ and $k_2 > 0$ are two feedback gains, and $n_i(z_i) \geq 0$ $(i = 1, 2)$ are two optional nonlinear damping terms, e.g. chosen as nondecreasing functions $n_i(z_i) = \kappa_i |z_i|^{n_i}$ with $n_i \geq 1$ and $\kappa_i \geq 0$ $(i = 1, 2)$ as design parameters. The following change of coordinates is needed to implement the controller:

$$z_1 = \psi - \psi_d \tag{8.218}$$

$$z_2 = r - \alpha_1 \tag{8.219}$$

The backstepping controller includes a PD-term as well as reference feedforward. In addition the nonlinear damping terms $n_i(z_i)$ $(i = 1, 2)$ can be used to improve the performance and stability of the closed-loop system.

When using feedback linearization all the nonlinearities in $H_N(r)$ are compensated for. This requires that the dissipative terms are known with good accuracy which is not true in many cases. The backstepping controller gives more design flexibility with respect to the damping terms. In fact, it is possible to exploit good damping terms like $n_3 r^3$ and $n_1 r$ in $H_N(r)$ instead of cancelling them. This is straightforward in set-point regulation; see Krstic *et al.* (1995), for instance. In trajectory tracking control, however, it is not clear how good damping with respect to a time-varying reference trajectory should be defined. A discussion on backstepping versus feedback linearization is found in Section 7.4.2.

Extensions to integral action can be done by using the method of Loria *et al.* (1999) and Fossen *et al.* (2001) which is referred to as "*backstepping with integral action*". Alternatively, an *integrator augmentation technique* can be applied. Both these methods are described in detail in Sections 7.4.4 and 7.4.5.

The actuator dynamics can be included in the design by using the approach of Fossen and Berge (1997) where backstepping is performed in three steps to include a first-order actuator model.

8.4.9 SISO Sliding Mode Control

Another attractive nonlinear design technique is *sliding mode control* (Utkin 1977) which incorporates techniques to handle model uncertainty. Sliding mode techniques are discussed in detail by Utkin (1992) while applications to marine vessels are found in Yoerger and Slotine (1985), Slotine and Li (1991), Healey and Lienard (1993), McGookin *et al.* (2000a, 2000b), for instance.

Define a scalar measure of tracking:

$$s = \dot{\tilde{\psi}} + 2\lambda\tilde{\psi} + \lambda^2 \int_0^t \tilde{\psi}(\tau)d\tau \tag{8.220}$$

where $\tilde{\psi} = \psi - \psi_d$ is the yaw angle tracking error and $\lambda > 0$ is a design parameter reflecting the bandwidth of the controller. For $s = 0$ this expression describes a sliding surface (manifold) with exponentially stable dynamics. To see this let us define a second sliding surface:

$$s_0 = \tilde{\psi} + \lambda \int_0^t \tilde{\psi}(\tau)d\tau \tag{8.221}$$

such that the manifold $s = 0$ can be rewritten as:

$$s = \dot{s}_0 + \lambda s_0 = 0 \tag{8.222}$$

Hence, both s_0 and $\tilde{\psi}$ converge exponential to zero since the linear system:

$$\begin{bmatrix} \dot{\tilde{\psi}} \\ \dot{s}_0 \end{bmatrix} = \begin{bmatrix} -\lambda & 1 \\ 0 & -\lambda \end{bmatrix} \begin{bmatrix} \tilde{\psi} \\ s_0 \end{bmatrix} \tag{8.223}$$

has to real eigenvalues at $-\lambda$. This ensures that the tracking error $\tilde{\psi} \to 0$ on the manifold $s = 0$. Hence, the *control objective* is reduced to find a nonlinear control law which ensures that:

$$\lim_{t\to\infty} s = 0 \tag{8.224}$$

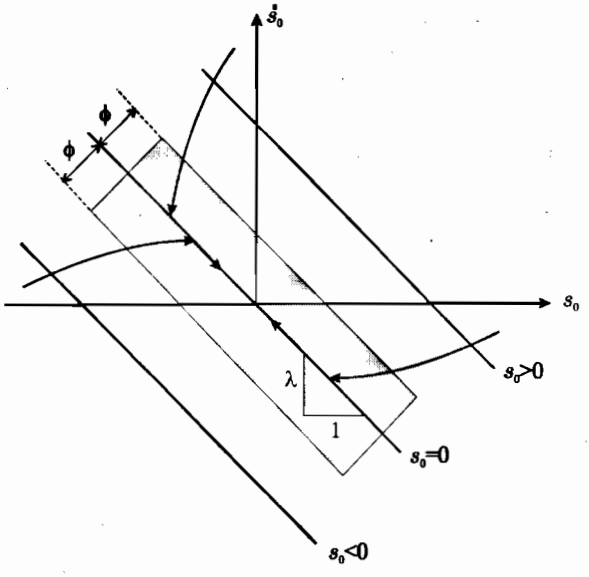


Figure 8.22: Graphical interpretation of the sliding surface $s = \dot{s}_0 + \lambda s_0$ and boundary layer $\phi > 0$.

A graphical interpretation of the *sliding surfaces* is given in Figure 8.22.

It is seen that a trajectory starting at $s > 0$ will move towards the sliding surface $s = 0$. Then $s = 0$ is reached the trajectory will continue moving on the straight line corresponding to $s = 0$ towards the equilibrium point $s_0 = 0$. A similar behavior is observed when starting with a negative value of s.

When deriving the control law, a stable ship model with nonlinear damping is considered:

$$T\dot{r} + n_3 r^3 + n_1 r = K\delta + \tau_{\text{wind}} \tag{8.225}$$

where τ_{wind} represents wind loads. Define a new signal v according to:

$$v = r - s \quad \Longrightarrow \quad s = r - v \tag{8.226}$$

such that

$$\begin{aligned} T\dot{s} &= T\dot{r} - T\dot{v} \\ &= K\delta + \tau_{\text{wind}} - (n_3 r^2 + n_1) r - T\dot{v} \\ &= K\delta + \tau_{\text{wind}} - (n_3 r^2 + n_1)(v + s) - T\dot{v} \end{aligned} \tag{8.227}$$

Consider the CLF:

$$V(s) = \frac{1}{2} T s^2, \quad T > 0 \tag{8.228}$$

Differentiating V along the trajectories of s yields:

$$\begin{aligned} \dot{V}(s) &= sT\dot{s} \\ &= s[K\delta + \tau_{\text{wind}} - (n_3 r^2 + n_1)(v + s) - T\dot{v}] \\ &= -[n_3 r^2 + n_1] s^2 + s[K\delta + \tau_{\text{wind}} - [n_3 r^2 + n_1] v - T\dot{v}] \end{aligned} \tag{8.229}$$

Let the control law be chosen as:

$$\delta = \frac{\hat{T}}{\hat{K}}\dot{v} + \frac{1}{\hat{K}}[\hat{n}_3 r^2 + n_1]v - \frac{1}{\hat{K}}\tau_{\text{wind}} - K_d s - K_s \text{sgn}(s) \tag{8.230}$$

where $K_d > 0$ and $K_s > 0$, while $\hat{T}, \hat{K}$, and $\hat{n}_3$ are estimates of T, K, and n_3, respectively. Notice that $n_1 = 1$ for a stable ship; see Section 8.1.4. The signum function is defined as:

$$\text{sgn}(s) = \begin{cases} 1 & \text{if } s > 0 \\ 0 & \text{if } s = 0 \\ -1 & \text{otherwise} \end{cases} \tag{8.231}$$

This implies that:

$$\begin{aligned} \dot{V}(s) &= -[n_3 r^2 + n_1 + K_d]s^2 - K_s |s| \\ &\quad + \left[\left(\frac{\hat{T}}{\hat{K}} - \frac{T}{K} \right) \dot{v} + \left(\frac{1}{\hat{K}} - \frac{1}{K} \right) [n_1 v - \tau_{\text{wind}}] \right. \\ &\quad \left. + \left(\frac{\hat{n}_3}{\hat{K}} - \frac{n_3}{K} \right) r^2 v \right] s \end{aligned} \tag{8.232}$$

In order for this expression to become negative, K_s must be chosen to be so large that the parameter errors are dominated. Consequently:

$$\begin{aligned} K_s &\geq \left| \left(\frac{\hat{T}}{\hat{K}} - \frac{T}{K} \right) \dot{v} \right| + \left| \left(\frac{1}{\hat{K}} - \frac{1}{K} \right) [n_1 v - \tau_{\text{wind}}] \right| \\ &\quad + \left| \left(\frac{\hat{n}_3}{\hat{K}} - \frac{n_3}{K} \right) r^2 v \right| \end{aligned} \tag{8.233}$$

implies that:

$$\dot{V}(s) \leq -(n_3 r^2 + n_1 + K_d)s^2 \leq 0 \tag{8.234}$$

Consequently, $s \to 0$ and thus $\tilde{\psi} \to 0$. One way to find an estimate of K_s is to assume e.g. 20 % uncertainty in all elements such that:

$$K_s \geq 1.2\frac{\hat{T}}{\hat{K}}|\dot{v}| + 1.2\frac{1}{\hat{K}}|n_1 v - \tau_{\text{wind}}| + 1.2\frac{\hat{n}_3}{\hat{K}}|r^2 v| \tag{8.235}$$

It is well known that the switching term $K_s \text{sgn}(s)$ can lead to chattering for large values of K_s. Hence, $K_s > 0$ should be treated as a design parameter with (8.233) as a guideline. Recall that Lyapunov stability analysis results in conservative requirements for all gains.

Chattering in the controller can be eliminated by replacing the signum function with a saturating function. Slotine and Li (1991) suggest smoothing out the control discontinuity inside a boundary layer according to:

$$\text{sat}(s) = \begin{cases} \text{sgn}(s) & \text{if } |s/\phi| > 1 \\ s/\phi & \text{otherwise} \end{cases} \tag{8.236}$$

where $\phi > 0$ can be interpreted as the boundary layer thickness. This substitution will assign a low-pass filter structure to the dynamics inside the boundary layer; see Figure 8.22. Another possibility is to replace $K_s \text{sgn}(s)$ with $K_s \tanh(s/\phi)$ where $\phi > 0$ is a design parameter used to shape the slope of the $\tanh(\cdot)$ function close to the origin.

Example 8.12 (Experimental Testing of a Medium Sized Ship)
A simplified version of the sliding mode controller was implemented and tested onboard the M/S Nornews Express. This is a 4600 *(dwt) ship with length* $L_{pp} = 110$ *(m) and beam* $B = 17.5$ *(m). The Nomoto gain and time constants were* $\hat{K} = 0.35$ *(*s^{-1}*) and* $\hat{T} = 29.0$ *(s) in the experiment suggesting that* $\hat{n}_3 = 0$ *and* $n_1 = 1$ *(linear stable ship); see Section 8.1.4. No wind feedforward was used. The control law used in the experiment was chosen as:*

$$\delta = \frac{\hat{T}}{\hat{K}}\dot{v} + \frac{1}{\hat{K}}[\hat{n}_3 r^2 + n_1]v - K_d s - K_s \tanh(s/\phi) \tag{8.237}$$

with $K_s = 0$ *since the estimates of* K *and* T *were quite good. The feedforward signals were computed as:*

$$\dot{v} = \dot{r} - \dot{s} = \dot{r}_d - 2\lambda\tilde{r} - \lambda^2\tilde{\psi} \tag{8.238}$$

$$v = r - s = r_d - 2\lambda\tilde{\psi} - \lambda^2 \int_0^t \tilde{\psi}(\tau)d\tau \tag{8.239}$$

Notice that $K_d s$ *contains three terms representing a conventional PID-controller. The performance of the autopilot with these model parameters are shown in Figure 8.23. This design can easily be robustified by using* $K_s > 0$ *and* $\phi > 0$*. Similarly, nonlinear damping can be included by choosing* $\hat{n}_3 > 0$*.*

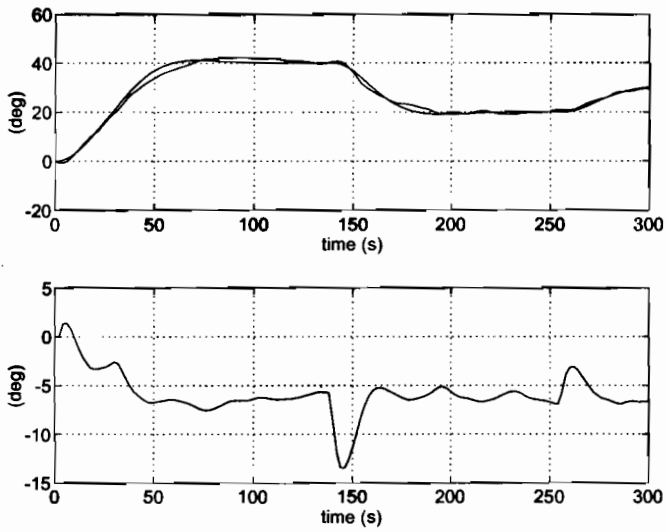


Figure 8.23: Tracking control with the M/S Nornews Express. Upper plots show ψ and ψ_d while lower plot shows the rudder angle δ.

8.4.10 Output Feedback

Many ships are only equipped with a compass for heading control. In these cases a state observer reconstructing r from $y_1 = \psi + \psi_w$ must be constructed; see Sections 6.1.3–6.1.4. The state observer can be combined with a state feedback controller, and stability can be proven through a *separation principle*. This is an attractive approach from an industrial point of view, since it gives a modular software representation; one subroutine for the state observer and one for the feedback controller. For linear systems this design is recognized as the LQG-controller where you combine a linear quadratic (LQ) controller with the Kalman filter. Nonlinear extensions are also possible thanks to the results of (Panteley and Loria 1998) which in fact is a *nonlinear separation principle*. A nonlinear separation principle allows the designer to combine a nonlinear state feedback controller with a nonlinear observer in cascade and thereafter prove uniform global asymptotic stability (UGAS) for the total system. This has been done with great success in Loria *et al.* (2000) where the case study was ship positioning in 3 DOF; see Section 11.2.4 for a detailed description. This result can also be applied to the 1 DOF autopilot case by removing the sway-yaw subsystem in the analysis.

An alternative approach to the separation principle is to design an *output feedback controller* directly by specifying $y_1 = \psi + \psi_w$ as the system output. In this case an "internal estimate" of r will be generated in the controller. This section discusses observer backstepping and semiglobal Lyapunov-based design techniques for this purpose. More results on nonlinear observer design can be found in Nijmeijer and Fossen (1999), while an adaptive output feedback controller is proposed by Lauvdal and Fossen (1998*a*).

Observer Backstepping

An output feedback autopilot can easily be designed by using the vectorial observer backstepping approach of Fossen and Grøvlen (1998). An extension to course unstable ships is found in Robertson and Johansson (1998) while integral action and wave filtering can also be included by using the technique of Aarset *et al.* (1998). Consider the autopilot model in the form:

$$\dot{\psi} = r \tag{8.240}$$

$$m\dot{r} + dr = \delta \tag{8.241}$$

where Nomoto's 1st-order model is recognized by choosing $m = T/K$ and $d = 1/K$. The main result is the following observer:

$$\dot{\hat{\psi}} = \hat{r} + K_1\tilde{\psi} \tag{8.242}$$

$$m\dot{\hat{r}} + d\hat{r} = \delta + K_2\tilde{\psi} \tag{8.243}$$

while the controller is given by:

$$\delta = \phi(z_1, z_2, \hat{\psi}) + C_2 z_2 + D_2 z_2 + z_1 \tag{8.244}$$

$$\phi(z_1, z_2, \hat{\psi}) = -(C_1 + D_1)^2 z_1 + (C_1 + D_1)(z_2 + K_1\tilde{\psi}) - \dot{r}_d - d\hat{\psi} \tag{8.245}$$

with:

$$z_1 = \hat{\psi} - \psi_d \tag{8.246}$$
$$z_2 = \hat{r} - \alpha_1 \tag{8.247}$$
$$\alpha_1 = r_d - C_1 z_1 - D_1 z_1 \tag{8.248}$$

where $C_1 > 0$ and $C_2 > 0$ represent linear damping and D_1 and D_2 are nonlinear damping gains. The gain requirements are:

$$K_1 = Q_1/P_1, \quad P_1 > 0 \tag{8.249}$$
$$K_2 = P_1/P_2 + d, \quad P_2 > 0 \tag{8.250}$$
$$P_2 d = Q_2 \tag{8.251}$$
$$D_1 = d_1 K_1^2, \quad d_1 > 0 \tag{8.252}$$
$$D_2 = d_4 \omega_1^2, \quad d_4 > 0 \tag{8.253}$$
$$\omega_1 = (C_1 + D_1) K_1 + K_2 \tag{8.254}$$

where $Q_1 > \frac{1}{4}(\frac{1}{d_1} + \frac{1}{d_4}) > 0$ and $Q_2 > \frac{1}{4d_4} > 0$; see Section 11.2.5. The stability analysis of the observer-controller in 3 DOF is also presented in Section 11.2.5 where GES is proven. The autopilot case is therefore a 1 DOF special case of the more general solution.

Extension to systems with nonlinear damping can be done for systems in the form:

$$m\dot{r} + dr + d_n(r) = \tau \tag{8.255}$$

where $d_n(r)$ is a *monotone nonlinear damping* term (see Arcak and Kokotovic 1999*a*, Arcak and Kokotovic 1999*b*), that is:

$$\frac{\partial d_n(r)}{\partial r} > 0 \Rightarrow d_n(r)\, r > 0, \quad \forall r \in \mathbf{R} \tag{8.256}$$

For more details on designs with nonlinear damping see Aamo *et al.* (2000) and references therein.

Semiglobal Output Feedback Control

An attractive design methodology for heading control of ships is the semiglobal exponential output feedback controller of Vik and Fossen (1997). This controller is based on results in robotics (Berghuis 1993), but augmented with a wave filter. The main result when applied to Norrbin's model (8.56) is:

$$\delta = \frac{1}{K}\left(T\dot{r}_d + H_N(r_d) - K_d\left[s_1 - s_2\right]\right) \tag{8.257}$$
$$s_1 = \dot{\tilde{\psi}} + 2\lambda_1\tilde{\psi} + \lambda_1^2 \int_0^t \tilde{\psi}(\tau)d\tau \tag{8.258}$$
$$s_2 = \dot{\tilde{\psi}} + \lambda_2\tilde{\psi} \tag{8.259}$$

where

$$\bar{\psi} = (y_1 - \hat{\psi}_w) - \psi_d \quad \text{(tracking error)} \tag{8.260}$$

$$\tilde{\psi} = y_1 - (\hat{\psi} + \hat{\psi}_w) \quad \text{(estimation error)} \tag{8.261}$$

and $y_1 = \psi + \psi_w$ is the compass measurement. The controller has the following design parameters $K_d > 0$, $\lambda_1 > 0$, and $\lambda_2 > 0$ while the observer states are updated as:

$$\dot{\hat{\psi}} = \hat{r} + K_1\bar{\psi} + L_1\tilde{\psi} \tag{8.262}$$

$$\dot{\hat{r}} = \ddot{\psi}_d + K_2\bar{\psi} + L_2\tilde{\psi} \tag{8.263}$$

$$\dot{\hat{\xi}}_w = \hat{\psi}_w + L_3\tilde{\psi} \tag{8.264}$$

$$\dot{\hat{\psi}}_w = -2\lambda\omega_0\hat{\psi}_w - \omega_0^2\hat{\xi}_w + L_4\tilde{\psi} \tag{8.265}$$

where K_i $(i = 1, 2)$ and L_j $(j = 1, \ldots, 4)$ are the observer gains. In Vik and Fossen (1997) it was shown, after some laborious calculations, that the equilibrium point of the closed-loop observer-controller error dynamics is *semiglobal exponentially stable*. A simulation study of the *Mariner class vessel* (Chislett and Strøm-Tejsen 1965*b*) is also found in Vik and Fossen (1997). This reference also gives insight on how the controller-observer gains should be chosen.

8.5 Exercises

Exercise 8.1 *Find a non-dimensional state-space model for the Nomoto model (8.55) in Example8.5.*

Exercise 8.2 *Find the scaling factors α, β and γ in the expressions:*

$$N'_v = \frac{1}{\alpha}N_v$$

$$Z''_p = \frac{1}{\beta}Z_p$$

$$X''_{\dot{v}} = \frac{1}{\gamma}X_{\dot{v}}$$

Exercise 8.3 *Compute the turning radius R for the ship model in mariner.m theoretically for $\delta_R = 15$ (deg) and $U = 8$ (m/s). Use the Matlab™ GNC toolbox and simulate the ship in a turning circle. Does the theoretically computed value correspond to the simulated results? Explain this result. Simulate the ship for different speeds and plot R as a function U.*

Exercise 8.4 *Derive a nonlinear backstepping controller for the model of Bech and Wagner Smith (1969):*

$$\dot{\psi} = r \tag{8.266}$$

$$T_1T_1\ddot{r} + (T_1 + T_2)\dot{r} + KH_B(r) = K(\delta + T_3\dot{\delta}) \tag{8.267}$$

and prove that the equilibrium point of the closed-loop system is GES.

Exercise 8.5 *Consider the nonlinear ship model:*

$$\dot{\psi} = r \tag{8.268}$$

$$m\dot{r} + dr + d_n(r) = \delta \tag{8.269}$$

and the control law:

$$\delta = -K_p\psi - K_d r \tag{8.270}$$

where $K_p > 0$ and $K_d > 0$. Show that the Lyapunov function:

$$V = \frac{1}{2}\left(mr^2 + K_p\psi^2\right) \tag{8.271}$$

satisfies $\dot{V} \leq 0$ if $d_n(r)$ is a monotonic function, that is:

$$\frac{\partial d_n(r(t))}{\partial r(t)} > 0 \Rightarrow d_n(r(t))\, r(t) > 0 \tag{8.272}$$

What are the conditions on $H_B(r)$ and $H_n(r)$, see the models in Section 8.1.4, for this to be true.

Exercise 8.6 *Consider Nomoto's model in the form:*

$$\dot{\psi} = r \tag{8.273}$$

$$T'\dot{r} + (U/L)r = (U/L)^2 K'\,\delta \tag{8.274}$$

and let the LQ weighting matrices be chosen as:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & \lambda_1 \end{bmatrix}, \quad R = \lambda_2 \tag{8.275}$$

where $\lambda_1 > 0$ and $\lambda_2 > 0$. Show that (8.176) and (8.177) are the LQR solution to (8.172), see Section 7.2.1.

Chapter 9

Autopilots with Roll Damping

This chapter discusses methods for autopilot roll stabilization using fins alone or in combination with rudders. The main motivation for using roll stabilizing systems on merchant ships are to prevent cargo damage and to increase the effectiveness of the crew by avoiding or reducing seasickness. This is also important from a safety point of view. For naval ships critical marine operations like landing a helicopter, formation control, underway replenishment, or effectiveness of the crew during combat are critical operations.

Several *passive* and *active* (feedback control) systems have been proposed to accomplish roll reduction; see Burger and Corbet (1960), Lewis (1967), and Bhattacharyya (1978) for a more detailed discussion. Some passive solutions are:

Bilge keels: Bilge keels are fins in planes approximately perpendicular to the hull or near the turn of the bilge. The longitudinal extent varies from about 25 to 50 percent of the length of the ship. Bilge keels are widely used, are inexpensive but increase the hull resistance. In addition to this, they are effective mainly around the natural roll frequency of the ship. This effect significantly decreases with the speed of the ship. Bilge keels were first demonstrated in about 1870.

Hull Modifications: The shape and size of the ship hull can be optimized for minimum rolling using hydrostatic and hydrodynamic criteria. This must, however, be done before the ship is built.

Anti-Rolling Tanks: The most common anti-rolling tanks in use are free-surface tanks, U-tube tanks and diversified tanks. These systems provide damping of the roll motion even at small speeds. The disadvantages of course are the reduction in metacenter

height due to free water surface effects and that a large amount of space is required. The earliest versions were installed about the year 1874.

The most widely used systems for *active* roll damping are:

Fin Stabilizers: Fin stabilizers are a highly attractive devices for roll damping. They provide considerable damping if the speed of the ship is not too low. The disadvantage with additional fins are increased hull resistance and high costs associated with the installation, since at least two new hydraulic systems must be installed. Retractable fins are popular, since they are inside the hull when not in use (no additional drag). It should be noted that fins are not effective at low speed and that they cause underwater noise in addition to drag. Fin stabilizers were patented by *John I. Thornycroft* in 1889.

Rudder-Roll Damping (RRD): Roll damping by means of the rudder is relatively inexpensive compared to fin stabilizers, has approximately the same effectiveness, and causes no drag or underwater noise if the system is turned off. However, RRD requires a relatively fast rudder to be effective, typically rudder rates $\dot{\delta}_{\max} = 5$–20 (deg/s) are needed. RRD will not be effective at low ship speeds.

For a history of ship stabilization, the interested reader is advised to consult Bennett (1991), while a detailed evaluation of different ship roll stabilization systems can be found in Sellars and Martin (1992).

9.1 Autopilot Models for Steering and Roll Damping

The 3 DOF horizontal models in Section 8.1 can easily be extended to include the *roll* mode. The resulting model is a 4 DOF model in *surge, sway, roll,* and *yaw.*

9.1.1 The Linear Model of Van Amerongen and Van Cappelle (1981)

The speed equation (8.9) can be decoupled from the *sway, roll,* and *yaw* modes. The resulting linear model takes the form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(u_o)\boldsymbol{\nu} + \mathbf{G}\boldsymbol{\eta} = \boldsymbol{\tau} \tag{9.1}$$

where u_o =constant, $\boldsymbol{\nu} = [v, p, r]^\top$ and $\boldsymbol{\eta} = [e, \phi, \psi]^\top$ are the states while $\boldsymbol{\tau}$ is the control vector.

For a ship with homogeneous mass distribution and xz-plane symmetry, $I_{xy} = I_{yz} = 0$ and $y_g = 0$. In addition, it is convenient to choose the origin $\mathbf{r}_g = [x_g, 0, z_g]^\top$ of the body-fixed coordinate system such that $I_{xz} = 0$ with corresponding added inertia $K_{\dot{r}} = N_{\dot{p}} = 0$; see Exercise 9.1.

From the general expressions (3.55) and (3.92) in Sections 3.1.3 and 3.2.1, respectively, we get (with non-zero I_{xz}):

$$\mathbf{M} = \begin{bmatrix} m - Y_{\dot{v}} & -mz_g - Y_{\dot{p}} & mx_g - Y_{\dot{r}} \\ -mz_g - K_{\dot{v}} & I_x - K_{\dot{p}} & \underbrace{(-I_{xz} - N_{\dot{p}}) = 0}_{\text{Exercise 9.1}} \nearrow \\ mx_g - N_{\dot{v}} & (-I_{xz} - N_{\dot{p}}) = 0 \nearrow & I_z - N_{\dot{r}} \end{bmatrix} \tag{9.2}$$

The expression for $\mathbf{N}(u_o)$ is obtained by linearization of $\mathbf{C}(\boldsymbol{\nu})$ and $\mathbf{D}(\boldsymbol{\nu})$ about $\boldsymbol{\nu} = [u_0, 0, 0]^\top$ – i.e.:

$$\mathbf{N}(u_o) = \begin{bmatrix} -Y_v & -Y_p & mu_0 - Y_r \\ -K_v & -K_p & -mz_g u_0 - K_r \\ -N_v & -N_p & mx_g u_0 - N_r \end{bmatrix} \tag{9.3}$$

Recall from Section 3.2.3 that the *linear* restoring forces and moments for a surface vessels can be written:

$$\mathbf{G} = \text{diag}\{0,\ W\overline{GM}_T,\ 0\} \tag{9.4}$$

where $W = mg$ is the weight of water and $\overline{GM}_T$ is the transverse metacenter height.

In addition to these equations, the kinematic equations (assuming $q = \theta = 0$):

$$\dot{\phi} = p \tag{9.5}$$

$$\dot{\psi} = \cos\phi\, r \approx r \tag{9.6}$$

must be augmented to the system model. The general kinematic expressions are found in Section 2.2.1.

Actuator Model for Combined Fin and Rudder Control Systems

The actuator model can be written, see Section 7.5:

$$\boldsymbol{\tau} = \mathbf{TKu} \tag{9.7}$$

where $\mathbf{K} = \text{diag}\{K_1, \ldots, K_r\}$ is a diagonal matrix of force coefficients K_i $(i = 1_1, \ldots, r)$, and $\mathbf{T}$ is a matrix describing the location and direction of the different actuators. For fully actuated ships $\boldsymbol{\tau} = [Y, K, N]^\top$ and $\mathbf{u} \in \mathbb{R}^r$ $(r \geq 3)$. The model (9.7) can be used to describe an arbitrar number of fin and rudder servos.

State-Space Model

The linear model (9.1) together with (9.5)–(9.6) can be written in state-space model by defining the state vector as $\mathbf{x} = [v, p, r, \phi, \psi]^\top$. The elements associated with the matrices $\mathbf{A}$ and $\mathbf{B}$ are given by:

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{\mathbf{A}} \mathbf{x} + \underbrace{\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ b_{31} & b_{32} & \cdots & b_{3r} \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix}}_{\mathbf{B}} \mathbf{u} \tag{9.8}$$

where the elements a_{ij} are found from:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = -\mathbf{M}^{-1}\mathbf{N}(u_o); \quad \begin{bmatrix} * & a_{14} & * \\ * & a_{24} & * \\ * & a_{34} & * \end{bmatrix} = -\mathbf{M}^{-1}\mathbf{G} \tag{9.9}$$

while the elements b_{ij} are given by $\mathbf{B} = \mathbf{M}^{-1}\mathbf{T}\mathbf{K}$. Finally, the roll and yaw outputs are defined as:

$$\phi = \underbrace{[0,0,0,1,0]}_{\mathbf{c}_{\text{roll}}^\top}\mathbf{x}, \qquad \psi = \underbrace{[0,0,0,0,1]}_{\mathbf{c}_{\text{yaw}}^\top}\mathbf{x} \tag{9.10}$$

Decompositions in Roll and Sway-Yaw Subsystems

To simplify the system for further analysis, the state vector is reorganized such that state variables associated with the steering and roll dynamics are separated. Consequently, (9.8) is rewritten as:

$$\left[\begin{array}{c} \dot{v} \\ \dot{r} \\ \dot{\psi} \\ \hline \dot{p} \\ \dot{\phi} \end{array}\right] = \left[\begin{array}{ccc|cc} a_{11} & a_{13} & 0 & a_{12} & a_{14} \\ a_{31} & a_{33} & 0 & a_{32} & a_{34} \\ 0 & 1 & 0 & 0 & 0 \\ \hline a_{21} & a_{23} & 0 & a_{22} & a_{24} \\ 0 & 0 & 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} v \\ r \\ \psi \\ \hline p \\ \phi \end{array}\right] + \left[\begin{array}{cccc} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{31} & b_{32} & \cdots & b_{3r} \\ 0 & 0 & \cdots & 0 \\ \hline b_{21} & b_{22} & \cdots & b_{2r} \\ 0 & 0 & \cdots & 0 \end{array}\right] \mathbf{u} \tag{9.11}$$

Define:

$$\left[\begin{array}{c} \dot{\mathbf{x}}_\psi \\ \dot{\mathbf{x}}_\phi \end{array}\right] = \left[\begin{array}{cc} \mathbf{A}_{\psi\psi} & \mathbf{A}_{\psi\phi} \\ \mathbf{A}_{\phi\psi} & \mathbf{A}_{\phi\phi} \end{array}\right] \left[\begin{array}{c} \mathbf{x}_\psi \\ \mathbf{x}_\phi \end{array}\right] + \left[\begin{array}{c} \mathbf{B}_\psi \\ \mathbf{B}_\phi \end{array}\right] \mathbf{u} \tag{9.12}$$

where $\mathbf{x}_\psi = [v, r, \psi]^\top$ and $\mathbf{x}_\phi = [p, \phi]^\top$. If the coupling matrices are small, that is $\mathbf{A}_{\psi\phi} = \mathbf{A}_{\phi\psi} = \mathbf{0}$, the following subsystems:

$$\left[\begin{array}{c} \dot{p} \\ \dot{\phi} \end{array}\right] = \left[\begin{array}{cc} a_{22} & a_{24} \\ 1 & 0 \end{array}\right] \left[\begin{array}{c} p \\ \phi \end{array}\right] + \left[\begin{array}{cccc} b_{21} & b_{22} & \cdots & b_{2r} \\ 0 & 0 & \cdots & 0 \end{array}\right] \mathbf{u} \tag{9.13}$$

and

$$\left[\begin{array}{c} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{array}\right] = \left[\begin{array}{ccc} a_{11} & a_{13} & 0 \\ a_{31} & a_{33} & 0 \\ 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} v \\ r \\ \psi \end{array}\right] + \left[\begin{array}{cccc} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{31} & b_{32} & \cdots & b_{3r} \\ 0 & 0 & \cdots & 0 \end{array}\right] \mathbf{u} \tag{9.14}$$

will describe the ship dynamics. The last expression is recognized as the 2nd-order Nomoto model with r control inputs.

Transfer Functions for Steering and Rudder-Roll Damping

The linearized model is quite useful for frequency analysis of RRD systems. For simplicity consider a ship with one rudder $\mathbf{u} = \delta$ such that $\mathbf{b} = [b_{11}, b_{21}, b_{31}, 0, 0]^\top$. For the state-space model (9.11) the transfer functions $\phi(s)/\delta(s) = \mathbf{c}_{\text{roll}}^\top(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}$ and $\psi(s)/\delta(s) = \mathbf{c}_{\text{yaw}}^\top(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}$ become:

$$\frac{\phi}{\delta}(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \approx \underbrace{\frac{K_{\text{roll}}\,\omega_{\text{roll}}^2\,(1 + T_5 s)}{(1 + T_4 s)(s^2 + 2\zeta\omega_{\text{roll}} s + \omega_{\text{roll}}^2)}}_{\text{no coupling between roll and sway-yaw}} \tag{9.15}$$

$$\frac{\psi}{\delta}(s) = \frac{c_3 s^3 + c_2 s^2 + c_1 s + c_0}{s(s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0)} \approx \underbrace{\frac{K_{\text{yaw}}\,(1 + T_3 s)}{s(1 + T_1 s)(1 + T_2 s)}}_{\text{no coupling between roll and sway-yaw}} \tag{9.16}$$

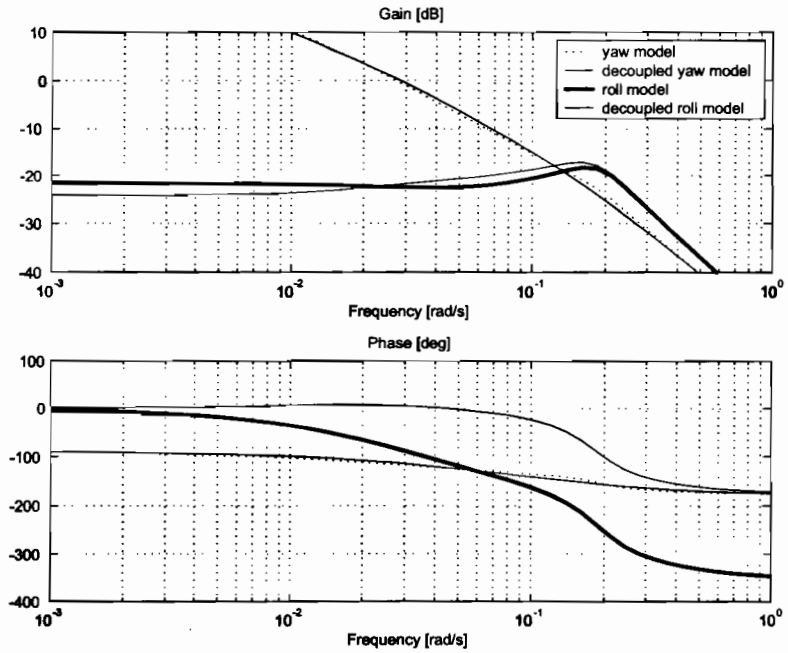


Figure 9.1: Transfer functions for the roll and sway-yaw subsystems corresponding to the Son and Nomoto container ship.

where the decoupled models (9.13) and (9.14) have been applied. In most cases, this approximation is rough so care should be taken. In Figure 9.1 it is seen that the phase of the roll transfer function is quite inaccurate for the decoupled model. This can be improved by using a model reduction via a balanced state-space realization (see *modred.m* and *ssbal.m* in Matlab^{TM}).

Also parametric investigations show that cross-couplings between steering and roll might give robust performance problems of RRD control systems (Blanke and Christensen 1993). This is also documented in Blanke (1996) who has identified the ship parameters for several loading conditions during sea trials with a series of ships. The results clearly show changes in the dynamics between the different ships in the series indicating that there is a robustness problem due to changes in load conditions and rudder shape. Nonlinear effects also give rise to the same problem. Identification of ship steering-roll models are discussed by Blanke and Tiano (1997). The interested reader is also advised to consult Van der Klugt (1987) for a discussion on decoupled linear models for RRD, while nonlinear models are presented in Section 9.1.2.

Example 9.1 (Roll and Sway-Yaw Transfer Functions)
The roll and yaw transfer functions corresponding to the model of Son and Nomoto (1981) are plotted in Figure 9.1 using the Matlab^{TM} GNC Toolbox file `ExRRD1.m`. *The plots show both the full state-space model (9.8) and the decoupled models (9.13)–(9.14). The model considered is a container ship of length* $L = 175$ *(m) and with a displacement volume of* $21,222$ *(m^3). The ship is moving at service speed* $u_0 = 7.0$ *(m/s). The model of Son and Nomoto*

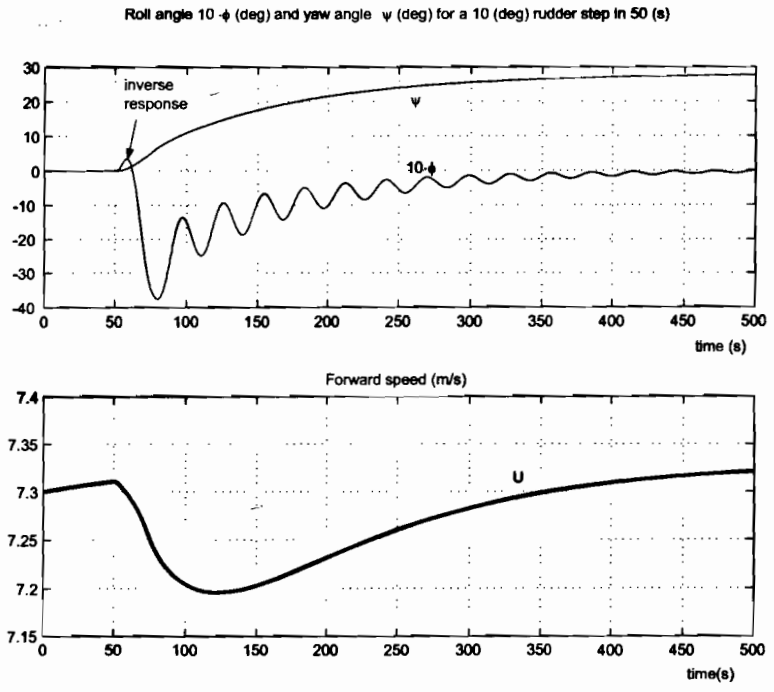


Figure 9.2: Roll angle $10\dot{\phi}$ and yaw angle ψ versus time for a 10 (deg) rudder step in 50 (s). Notice the inverse response in roll and speed reduction during turning.

(1981) is based on a 3rd-order Taylor series expansion (see Section 8.1.4) of the hydrodynamic forces including higher order restoring terms replacing (9.4). The nonlinear model is included in the GNC Toolbox under the file name `container.m` *while a linearized version of this model is found in* `Lcontainer.m`. *The nonlinear model is described more closely in the next section. The numerical values for the transfer functions are:*

$$\begin{aligned} \frac{\phi}{\delta}(s) &= \frac{0.0032(s-0.036)(s+0.077)}{(s+0.026)(s+0.116)(s^2+0.136s+0.036)} \\ &\approx \frac{0.083(1+49.1s)}{(1+31.5s)(s^2+0.134s+0.033)} \end{aligned} \tag{9.17}$$

and:

$$\begin{aligned} \frac{\psi}{\delta}(s) &= \frac{0.0024(s+0.0436)(s^2+0.162s+0.035)}{s(s+0.0261)(s+0.116)(s^2+0.136s+0.036)} \\ &\approx \frac{0.032(1+16.9s)}{s(1+24.0s)(1+9.2s)} \end{aligned} \tag{9.18}$$

corresponding to $\omega_{\text{roll}} = 0.189$ *(rad/s) and* $\zeta = 0.36$. *It is seen that the amplitudes of the roll and yaw models are quite close. However, the decoupled model in roll does not describe*

the phase with sufficient accuracy, so stability problems could be an issue when designing a model-based RRD. The main reason for this is that one pole-zero pair is omitted in the decoupled roll model. Since, this is a right-half-plane zero, $z = 0.036$ (rad/s), the pole-zero pair gives an additional phase lag of -180 (deg) as observed in the plot of the full model. This will of course result in serious stability problems when trying to damp the roll motion.

In practise it will be difficult to design a RRD for this system since the controller should reduce the energy at the peak frequency $\omega_{\text{roll}} = 0.189$ (rad/s) which is much higher than the right-half-plane zero $z = 0.036$ (rad/s). This is a non-minimum phase property which cannot be changed with feedback (recall that only poles and not zeros can be moved using feedback control). The non-minimum phase characteristic is observed as an inverse response in roll when a step input is applied; see Figure 9.2.

The plots are generated by simulating the nonlinear model of Son and Nomoto (see `ExRRD3.m`*). The non-minimum phase behavior due to the right-half-plane zero is discussed more closely by Fossen and Lauvdal (1994) where both linear and nonlinear analyses of the models of Son and Nomoto are considered. The nonlinear equivalent to a right-half-plane zero is* unstable zero dynamics.

9.1.2 The Nonlinear Model of Son and Nomoto (1981)

A nonlinear model including roll for a high speed container ship has been proposed by Son and Nomoto (1981, 1982):

$$(m + m_x)\dot{u} - (m + m_y)vr = X + \tau_x \tag{9.19}$$

$$(m + m_y)\dot{v} + (m + m_x)ur + m_y\alpha_y\dot{r} - m_y l_y\dot{p} = Y + \tau_y \tag{9.20}$$

$$(I_x + J_x)\dot{p} - m_y l_y\dot{v} - m_x l_x ur = K - W\overline{GM}_T\phi + \tau_k \tag{9.21}$$

$$(I_z + J_z)\dot{r} + m_y\alpha_y\dot{v} = N - x_g Y + \tau_n \tag{9.22}$$

where m_x, m_y, J_x, and J_y denote the added mass and added moment of inertia in the x- and y-directions about the z- and x-axes, respectively. The control inputs are recognized as $\boldsymbol{\tau} = [\tau_x, \tau_y, \tau_k, \tau_n]^\top$. The added mass x-coordinates of m_x and m_y are denoted by α_x and α_y,while l_x and l_y are the added mass z-coordinates of m_x and m_y, respectively.

The terms on the right-hand side of (9.19)–(9.22) are defined in terms of a 3rd-order Taylor series expansion where small coefficients are neglected. The remaining terms are:

$$\begin{aligned} X = {} & X(u) + (1-t)T + X_{vr}vr + X_{vv}v^2 + X_{rr}r^2 + X_{\phi\phi}\phi^2 \\ & + X_\delta \sin\delta + X_{\text{ext}} \end{aligned} \tag{9.23}$$

$$\begin{aligned} Y = {} & Y_v v + Y_r r + Y_\phi\phi + Y_p p + Y_{vvv}v^3 + Y_{rrr}r^3 + Y_{vvr}v^2 r + Y_{vrr}vr^2 \\ & + Y_{vv\phi}v^2\phi + Y_{v\phi\phi}v\phi^2 + Y_{rr\phi}r^2\phi + Y_{r\phi\phi}r\phi^2 + Y_\delta\cos\delta + Y_{\text{ext}} \end{aligned} \tag{9.24}$$

$$\begin{aligned} K = {} & K_v v + K_r r + K_\phi\phi + K_p p + K_{vvv}v^3 + K_{rrr}r^3 + K_{vvr}v^2 r + K_{vrr}vr^2 \\ & + K_{vv\phi}v^2\phi + K_{v\phi\phi}v\phi^2 + K_{rr\phi}r^2\phi + K_{r\phi\phi}r\phi^2 + K_\delta\cos\delta + K_{\text{ext}} \end{aligned} \tag{9.25}$$

$$\begin{aligned} N = {} & N_v v + N_r r + N_\phi\phi + N_p p + N_{vvv}v^3 + N_{rrr}r^3 + N_{vvr}v^2 r + N_{vrr}vr^2 \\ & + N_{vv\phi}v^2\phi + N_{v\phi\phi}v\phi^2 + N_{rr\phi}r^2\phi + N_{r\phi\phi}r\phi^2 + N_\delta\cos\delta + N_{\text{ext}} \end{aligned} \tag{9.26}$$

where $X(u)$ is usually modelled as quadratic drag $X(u) = X_{|u|u}|u|u$ and the subscript ext denotes external forces and moments due to wind, waves, and currents.

Matlab:
These models of Son and Nomoto (1981) are implemented in the GNC Toolbox as:

```
[xdot,U] = container(x,ui)
```

The linearized model for $U = U_0$ is accessed as:

```
[xdot,U] = Lcontainer(x,ui,U0)
```

where `xdot=[u v r x y psi p phi delta]'` and `ui=[ delta_c n_c]'`. In the linear case only one input, `delta_c`, is used since the forward speed `U0` is constant. For the nonlinear model propeller rpm, `n_c`, should be positive.

9.1.3 The Nonlinear Model of Christensen and Blanke (1986)

An alternative model formulation describing the steering and roll motions of ships has been proposed by Christensen and Blanke (1986). This model is written as:

$$\begin{bmatrix} m-Y_{\dot v} & -mz_g-Y_{\dot p} & mx_g-Y_{\dot r} & 0 & 0 \\ -mz_g-K_{\dot v} & I_x-K_{\dot p} & 0 & 0 & 0 \\ mx_g-N_{\dot v} & 0 & I_z-N_{\dot r} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot v \\ \dot p \\ \dot r \\ \dot\phi \\ \dot\psi \end{bmatrix} = \begin{bmatrix} Y_{\text{ext}} \\ K_{\text{ext}} \\ N_{\text{ext}} \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{\tau}$$

$$+ \begin{bmatrix} Y_{uv}|u| & Y_{|u|p}|u| & -mu + Y_{ur}u & Y_{uu\phi}u^2 & 0 \\ K_{|u|v}|u| & K_{up}u + K_p & K_{ur}u & \overline{WGM}_T + K_{uu\phi}u^2 & 0 \\ N_{uv}u & 0 & N_{|u|r}|u| - mx_g u & N_{|u|u\phi}|u|u & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \\ \psi \end{bmatrix}$$

where the forces and moments associated with the roll motion are assumed to involve the square terms of the surge speed u^2 and $|u|u$. The terms Y_{ext}, K_{ext}, and N_{ext} consist of possible contributions from external disturbances while control inputs like rudders, propellers, bow thrusters, etc., are included in $\boldsymbol{\tau}$.

9.2 Rudder-Roll Damping (RRD) Control Systems

Rudder roll damping (RRD) was first suggested in the late seventies; see Cowler and Lambert (1972, 1975), Carley (1975), Lloyd (1975), and Baitis (1980). Research in the early eighties showed that it was indeed feasible to control the heading of a ship with at least one rudder while simultaneously using the rudder for roll damping. If only one rudder is used, this is an *underactuated control* problem. In the linear case this can be solved by *frequency separation* of the steering and roll modes since course control can be assumed to be a low-frequency tracking control problem while roll damping can be achieved at higher frequencies.

Before designing a RRD system the applicability of the control system in terms of effectiveness should be determined (Roberts 1993). For a large number of ships it is in fact impossible to obtain a significant roll damping effect due to limitations of the rudder servo and the relatively large inertia of the ship.

Motivated by the results in the 1970s RRD was tested by the U.S. Navy by Baitis *et al.* (1983, 1989), in Sweden by Källström (1987), Källström *et al.* (1988), Källström and Schultz (1990), Källström and Theoren (1994), and in the Netherlands by Amerongen and coauthors. Van Amerongen *et al.* (1987), Van Amerongen and Van Nauta Lempke (1987), and Van der Klugt (1987) introduced LQG-theory in RRD systems. A similar approach has been proposed by Katebi *et al.* (1987), while adaptive RRD is discussed in Zhou (1990).

Blanke and co-workers have developed a RRD autopilot (Blanke *et al.* 1989) that has been implemented by the Danish Navy on 14 ships (Munk and Blanke 1987). Sea trials show that some of the ships had less efficient RRD systems than others. In Blanke and Christensen (1993) it was shown that the cross-couplings between steering and roll were highly sensitive to parametric variations which again resulted in robustness problems. Different loading conditions and varying rudder shapes have been identified as reasons for this (Blanke 1996). In Stoustrup *et al.* (1995) it has been shown that a robust RRD controller can be designed by separating the roll and steering specifications and then optimizing the two controllers independently. The coupling effects between the roll and yaw modes have also been measured in model scale and compared with full-scale trial results (Blanke and Jensen 1997) while a new approach to identification of steering-roll models have been presented by Blanke and Tiano (1997).

More recently H_∞ control has been used to deal with model uncertainties in RRD control systems. This allows the designer to specify frequency dependent weights for frequency separation between the steering and roll modes; see Yang and Blanke (1997, 1998). Qualitative feedback theory (QFT) has also been applied to solve the combined RRD-heading control problem under model uncertainty; see Hearns and Blanke (1998). Results from sea trials are reported in Blanke *et al.* (2000*a*).

Simulation and full-scale experimental results of RRD systems using a multivariate autoregressive model and the minimum AIC estimate procedure have been reported by Oda *et al.* (1996, 1997). Experimental results with various control strategies are also reported by Sharif *et al.* (1996). A nonlinear RRD control system using sliding mode control for compensation of modeling errors is reported in Lauvdal and Fossen (1997).

A gain scheduling algorithm for input rate and magnitude saturations in RRD damping systems have been developed by Lauvdal and Fossen (1998*b*). This method is motivated by the automatic gain controller (AGC) by Van der Klugt (1987) and a technique developed for stabilization of integrator chains with input rate saturation.

In this section the focus will be on linear quadratic optimal RRD. The interested reader is recommended to consult the references above for other design techniques.

9.2.1 Linear Quadratic Optimal RRD Control System

Consider the state-space model (9.8) with $r \geq 1$ rudder servos. This is written:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{9.27}$$

with $\mathbf{x} = [v, p, r, \phi, \psi]^\top$ and:

$$\phi = \mathbf{c}_{\text{roll}}^T \mathbf{x}, \qquad \psi = \mathbf{c}_{\text{yaw}}^T \mathbf{x} \tag{9.28}$$

The control objective is simultaneous course control $\psi = \psi_d = \text{constant}$ and RRD ($p_d = \phi_d = 0$). There will be a trade-off between accurate heading control (minimizing $\tilde{\psi} = \psi - \psi_d$)

and control action needed to increase the natural frequency ω_{roll} and damping ratio ζ_{roll} in roll. Also notice that it is impossible to regulate ϕ to a non-zero value while simultaneously controlling the heading angle to a non-zero value by means of one single rudder. This can easily be seen by performing a steady-state analysis of the closed-loop system. This suggests that we want to specify the output to be controlled as:

$$\mathbf{y} = [p, r, \phi, \psi]^\top, \quad \mathbf{y}_d = [0, 0, 0, \psi_d]^\top \tag{9.29}$$

Defining $\mathbf{y} = \mathbf{C}\mathbf{x}$ implies that:

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{9.30}$$

Application of optimal control theory implies that the control objective should be specified as an optimization problem for course-keeping, roll damping, and minimum fuel consumption. The trade-off between these quantities can be expressed as:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T \left(\tilde{\mathbf{y}}^\top \mathbf{Q} \tilde{\mathbf{y}} + \mathbf{u}^\top \mathbf{R} \mathbf{u} \right) \, d\tau \right\} \tag{9.31}$$

where $\tilde{\mathbf{y}} = \mathbf{y} - \mathbf{y}_d$, $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d$, and $\mathbf{x}_d = [0, 0, 0, 0, \psi_d]^\top$. Accurate steering is weighted against roll damping by specifying the cost matrix $\mathbf{Q} = \text{diag}\{Q_p, Q_r, Q_\phi, Q_\psi\} \geq 0$, while $\mathbf{R} = \text{diag}\{R_1, R_{21}, \ldots, R_r\} > 0$ weights the use of the different rudder servos.

The solution to the LQ trajectory tracking problem is; see Section 7.2.3:

$$\mathbf{u} = \mathbf{G}_1 \mathbf{x} + \mathbf{G}_2 \mathbf{y}_d \tag{9.32}$$

where

$$\mathbf{G}_1 = -\mathbf{R}^{-1} \mathbf{B}^\top \mathbf{P}_\infty \tag{9.33}$$

$$\mathbf{G}_2 = -\mathbf{R}^{-1} \mathbf{B}^\top (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \mathbf{C}^\top \mathbf{Q} \tag{9.34}$$

with $\mathbf{P}_\infty = \mathbf{P}_\infty^\top > 0$ is given by:

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^\top \mathbf{P}_\infty + \mathbf{C}^\top \mathbf{Q} \mathbf{C} = \mathbf{0} \tag{9.35}$$

Frequency Separation and Bandwidth Limitations

Since $(\mathbf{A}, \mathbf{B})$ is controllable and full state feedback is applied, it is possible to move all the five poles of the system. The closed-loop system becomes:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} = \underbrace{(\mathbf{A} + \mathbf{B}\mathbf{G}_1^\top)}_{\mathbf{A}_c} \mathbf{x} + \mathbf{B}\mathbf{G}_2 \underbrace{\mathbf{h}\psi_d}_{\mathbf{y}_d} \tag{9.36}$$

where

$$\mathbf{h} = [0, 0, 0, 1]^\top \tag{9.37}$$

The closed-loop transfer function in yaw is:

$$\psi(s) = \mathbf{c}_{\text{yaw}}^{\top}(s\mathbf{I} - \mathbf{A}_c)^{-1}\mathbf{B}\mathbf{G}_2\mathbf{h}\psi_d(s) \tag{9.38}$$

which clearly satisfy:

$$\lim_{t\to\infty} \psi(t) = \psi_d \tag{9.39}$$

Notice that integral action is needed in a practical implementation of the controller. Similarly, the closed loop in roll becomes:

$$\phi(s) = \mathbf{c}_{\text{roll}}^{\top}(s\mathbf{I} - \mathbf{A}_c)^{-1}\mathbf{B}\mathbf{G}_2\mathbf{h}\psi_d(s) \tag{9.40}$$

If one rudder is used to control both ϕ and ψ, frequency separation is necessary to achieve this. Assume that the steering dynamics is slower than the frequency $1/T_l$, and that the natural frequency in roll is higher than $1/T_h$. Hence, the VRU and compass measurements can be low- and high-pass filtered according to:

$$\frac{\phi}{\phi_{\text{vru}}}(s) = h_h(s) = \frac{T_h s}{1 + T_h s} \tag{9.41}$$

$$\frac{\psi}{\psi_{\text{compass}}}(s) = h_l(s) = \frac{1}{1 + T_l s} \tag{9.42}$$

It is also necessary to filter the roll and yaw rate measurements $p(s)$ and $r(s)$. These signals can also be computed by numerical differentiation of $\phi_{\text{vru}}(s)$ and $\psi_{\text{vru}}(s)$ using a state estimator. Alternatively they can be measured directly by using a gyro; see Chapter 6.

This suggests that the bandwidth of the course controller must satisfy (frequency separation):

$$\omega_b \ll \omega_{\text{roll}} \tag{9.43}$$

This again implies that the low- and high-pass filters must satisfy:

$$\underbrace{\omega_{\text{yaw}}}_{\substack{\text{cross-over}\\ \text{frequency}}} < \underbrace{\omega_b}_{\substack{\text{bandwidth}\\ \text{in yaw}}} < \underbrace{1/T_l}_{\substack{\text{low-pass filter}\\ \text{frequency}}} < \underbrace{1/T_h}_{\substack{\text{high-pass filter}\\ \text{frequency}}} < \underbrace{\omega_{\text{roll}}}_{\substack{\text{natural}\\ \text{frequency}}}$$

which clearly puts a restriction on the ships that can be stabilized. For many ships this requirement is impossible to satisfy due to limitations of the rudder servos (control forces).

Example 9.2 (RRD Control System)
Let $G_1 = [g_{11}, g_{12}, g_{13}, g_{14}, g_{15}]$ and $G_2 = [0, 0, 0, g_{24}]$ such that the solution (9.32) of the SISO LQ trajectory tracker problem can be written (assuming one rudder):

$$\delta = [g_{11}, g_{12}, g_{13}, g_{14}, g_{15}]\, x + g_{24}\psi_d \tag{9.44}$$

or:

$$\delta = \underbrace{-K_v v}_{\text{sway feedback}} \underbrace{-K_p(\psi - \psi_d) - K_d r}_{\text{PD course controller}} \underbrace{-K_{r1} p - K_{r2}\phi}_{\text{roll damper}} \tag{9.45}$$

where $K_v = -g_{11}, K_p = -g_{15} = g_{24}, K_d = -g_{13}, K_{r1} = -g_{12}$, *and* $K_{r2} = -g_{14}$. *Frequency separation suggests that:*

$$\delta = h_l(s)\delta_{course} + h_h(s)\delta_{roll} \tag{9.46}$$

where

$$\delta_{course} = -K_v v - K_p(\psi - \psi_d) - K_d r \tag{9.47}$$

$$\delta_{roll} = -K_{r1} p - K_{r2}\phi \tag{9.48}$$

The controller gains can be found by using the GNC toolbox m-function; see Section 7.2.3:

```
[G1,G2]=lqtracker(A,B,C,Q,R)
```

Alternatively, the gains can be computed by using pole placement. The two subsystems (9.13) and (9.14) with course autopilot and RRD become (neglecting the interactions between the systems and assuming frequency separation):

$$\begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \underbrace{a_{22} - b_{21}K_{r1}}_{-2\zeta_\phi\omega_\phi} & \underbrace{(a_{24} - b_{21}K_{r2})}_{-\omega_\phi^2} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p \\ \phi \end{bmatrix} = 0 \tag{9.49}$$

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} a_{11} - b_{11}K_v & a_{13} - b_{11}K_d & -b_{11}K_p \\ a_{31} - b_{31}K_v & a_{33} - b_{31}K_d & -b_{31}K_p \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v \\ r \\ \psi - \psi_d \end{bmatrix} = 0 \tag{9.50}$$

The poles can placed directly in Matlab^{TM} *using:*

```
[Kr1,Kr2]=place(A_phiphi,B_phiphi,[p_phi1,p_phi2])
[Kv,Kp,Kd]=place(A_psipsi,B_psipsi,[p_psi1,p_psi2,p_psi3])
```

For roll it is seen that:

$$-\omega_\phi^2 = a_{24} - b_{21}K_{r2}, \qquad -2\zeta_\phi\omega_\phi = a_{22} - b_{21}K_{r1} \tag{9.51}$$

or

$$K_{r1} = \frac{a_{22} + 2\zeta_\phi\omega_\phi}{b_{21}}, \qquad K_{r2} = \frac{a_{24} + \omega_\phi^2}{b_{21}} \tag{9.52}$$

where ζ_ϕ *and* ω_ϕ *are pole placement design parameters that can be used instead of eigenvalues. The model of Son and Nomoto (see* `ExRRD2.m` *in the GNC Toolbox) has been used to demonstrate how an LQ optimal RRD control system can be designed. The linear state-space model for the container ship is:*

$$\mathbf{A} = \begin{bmatrix} -0.0406 & -1.9614 & 0.2137 & 0.1336 & 0 \\ 0.0011 & -0.1326 & -0.1246 & -0.0331 & 0 \\ -0.0010 & 0.0147 & -0.1163 & -0.0006 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -0.0600 \\ 0.0035 \\ 0.0026 \\ 0 \\ 0 \end{bmatrix} \tag{9.53}$$

The controller gains were computed using `[G1,G2]=lqtracker(A,B,C,Q,R)` *with the weights:*

```
Q=diag([10000 1000 10 1]), R=0.5
```

resulting in:

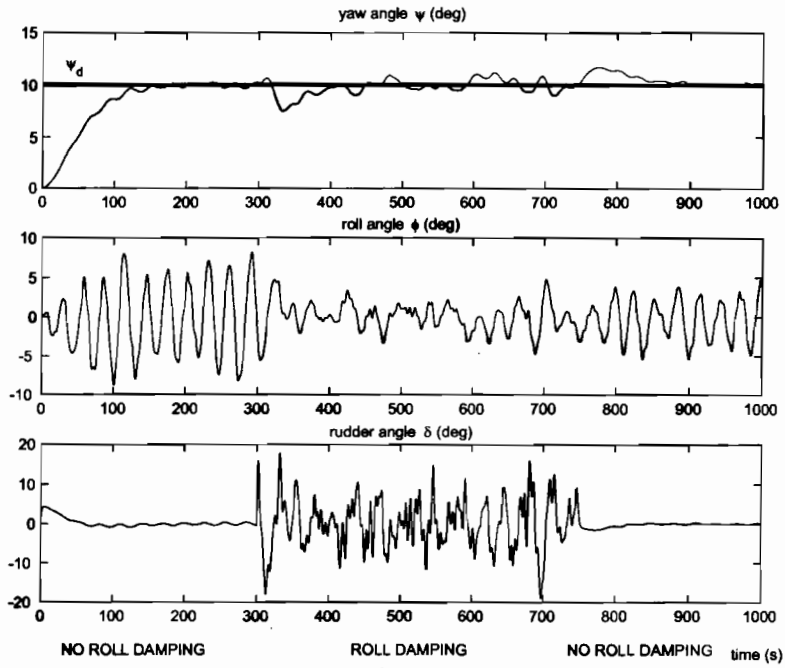


Figure 9.3: Performance of RRD control system during course-keeping and course-changing (10 deg). The RRD system is active between $t = 300 - 700$ (s).

```
G1=[0.1631-16.1193-6.7655-1.1644-0.4472],  G2=[0 0 0 0.4472]
```

Notice that $g_{15} = -g_{24}$. *The open-loop and closed loop poles are computed in Matlab*$^{\text{TM}}$ *by using the commands* `damp(A)` *and* `damp(A+B*G1)`; *see Table 9.1.*

Table 9.1: Eigenvalues, damping ratios and frequencies for RRD control systems.

Eigenvalues		Damping		Frequencies (rad/s)	
open loop	closed loop	open loop	closed loop	open loop	closed loop
0	-0.061	–	1.00	–	0.016
-0.027	-0.026	1.00	1.00	0.027	0.026
$-0.071 + 0.183i$	$-0.100 + 0.165i$	0.36	0.52	0.197	0.193
$-0.071 - 0.183i$	$-0.100 - 0.165i$	0.36	0.52	0.197	0.193
-0.121	-0.131	1.00	1.00	0.121	0.131

It is seen that the natural frequency and relative damping ratio in roll are $\omega_{roll} = 0.193$*(rad/s) and* $\zeta_{roll} = 0.36$, *respectively. This is improved to* $\omega_{roll} = 0.197$*(rad/s) and* $\zeta_{roll} = 0.52$ *by roll feedback. It is difficult to increase the relative damping ratio further due to limitations of the steering machine (*$\dot{\delta}_{\max} = 20$ *(deg/s) and* $\delta_{\max} = 20$ *(deg)). These values can, however, be changed in* `RRDcontainer.m`.

Since the roll frequency ω_{roll} ***is*** 0.193*(rad/s) and the cross-over frequency in yaw* ω_{yaw}

is 0.03 *(rad/s), see Figure 9.1 in Example 9.1, it is approximately one decade between the frequencies* ω_{yaw} *and* ω_{roll}. *Therefore, frequency separation can be obtained by choosing the low-pass and high-pass filter frequencies as* $1/T_l = 0.1$ *(rad/s) in yaw and* $1/T_h = 0.05$ *(rad/s) in roll, respectively. It is seen that the heading controller moves the poles to* $-0.061, -0.026$ *and* -0.131 *resulting in satisfactory course-changing capabilities; see Figure 9.3. It is also seen that the course-keeping performance is degraded during RRD. The additional yawing motion, typically* $1-2$ *degrees in amplitude, is the price paid for adding roll feedback to an autopilot system. Also notice that the right-half-plane zero in the transfer function* $\phi/\delta_1(s)$ *given by (9.15) is unchanged since feedback only moves the poles.*

9.2.2 Performance Criterion for RRD

The percentage roll reduction of RRD system can be computed by using the following criterion of Oda et al. (1992):

$$\text{Roll reduction} = \frac{\sigma_{\text{AP}} - \sigma_{\text{RRD}}}{\sigma_{\text{AP}}} \times 100(\%) \tag{9.54}$$

where

AP = standard deviation of roll rate during course-keeping (RRD off)
RRD = standard deviation of roll rate during course-keeping (RRD on)

For the case study in Example 9.2, $\sigma_{\text{AP}} = 0.0105$ and $\sigma_{\text{RRD}} = 0.0068$. This resulted in a roll reduction of approximately 35 (%) during course-keeping. For small high-speed vessels a roll reduction as high as 50–75 % can be obtained. This of course depends on the shape of the hull (hydrodynamic effects) and the capacity of the steering machine. In particular the maximum rudder rate $\dot{\delta}_{\max}$ should be in the magnitude of 15–20 % to obtain good results.

9.3 Fin Stabilization Control Systems and RRD

This section discusses methods for RRD and fin stabilization. The most effective roll damping systems are those which combine stabilizing fins and rudders; see Källström (1981) and Roberts and Braham (1990). Warship stabilization using integrated rudder and fins are discussed by Roberts (1992). More recently robust fin stabilizer controller design using the QFT and H_∞ design techniques have been presented by Hearns *et al.* (2000) while the performance of classical PID, optimized PID (Hickey *et al.* 2000), and H_∞-controllers are compared in Katebi *et al.* (2000). Sea trials with the *MV Barfleur* using PID and H_∞-controllers are presented in Hickey *et al.* (1997), and experimental results with a fin/RRD damping system onboard a frigate-size Royal Naval warship are reported in Sharif *et al.* (1995, 1996).

Reduction of vertical accelerations of fast ferries using fins and a T-foil is discussed by Esteban *et al.* (2000) and Giron-Sierra *et al.* (2001) while the modeling and identification results are reported in de-la-Cruz *et al.* (1998) and Aranda *et al.* (2000). A Matlab/Simulink simulation tool for fast ferries is presented in Esteban *et al.* (2001).

Fin stabilizers are attractive for roll reduction since they are highly effective, works on a large number of ships, and are much more easier to control, even for varying load conditions and actuator configurations. Fins stabilizers are effective at high speed, but at the price of

additional drag and added noise. The most economical systems are retractable fins, where additional drag is avoided during normal operation, since fin stabilizers are not needed in moderate weather. Another nice feature of fin stabilizing systems is that they can be used to control ϕ to a non-zero value (heel control). This is impossible with a RRD control system where accurate control of ψ has the priority.

9.3.1 Linear Quadratic Energy Optimal Autopilot with Roll Damping

In this section an integrated fin and RRD system will be designed. Notice that a stand-alone fin stabilization system can be constructed by simply removing the rudder inputs from the input matrix.

When designing an LQ optimal fin/RRD system the following two different model representations can be used:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} = \boldsymbol{\tau} \quad \text{where} \quad \boldsymbol{\tau} = \mathbf{T}\underbrace{\mathbf{K}\mathbf{u}}_{\mathbf{f}} \tag{9.55}$$

or

$$\dot{\boldsymbol{\nu}} = \underbrace{-\mathbf{M}^{-1}\mathbf{D}}_{\substack{\text{upper left part} \\ \text{of } \mathbf{A} \text{ in (9.8)}}} \boldsymbol{\nu} + \underbrace{\mathbf{M}^{-1}\mathbf{T}\mathbf{K}}_{\substack{\text{upper part} \\ \text{of } \mathbf{B} \text{ in (9.8)}}} \mathbf{u} \tag{9.56}$$

In the first representation generalized force $\boldsymbol{\tau}$ is used as control input while the last representation use $\mathbf{u}$, that is, propeller rpm, rudder angles, fin angles, etc. In practise it is advantageous to use (9.55) instead of (9.56), since actuator failures can be handled independently by the control allocation algorithm without redesigning the control law. Notice that the $\mathbf{B}$ matrix in (9.56) depends on $\mathbf{T}$ and $\mathbf{K}$ while these matrices are not used in (9.55); see Section 7.5. This is demonstrated in the following example:

Example 9.3 (Actuator Model for Fins and Rudders)
Consider a ship with port and starboard fins where δ_{f1} and δ_{f2} are used to denote the starboard and port fin angle deflections (forces in the z-direction). Let δ_{r1} and δ_{r2} denote the starboard and port rudders (forces in the y-direction). The control forces are (linear theory); see Section 7.5:

$$Y = K_1\delta_{r1} + K_2\delta_{r2} \tag{9.57}$$
$$Z = K_3\delta_{f1} + K_4\delta_{f2} \tag{9.58}$$

If the rudders are located aft of the ship at the positions $(-l_{x_r}, \pm l_{y_r}, 0)$ and the fins forward at $(l_{x_f}, \pm l_{y_f}, l_{z_f})$ the roll and yaw moments become:

$$K = l_{y_f}K_3\delta_{f1} - l_{y_f}K_4\delta_{f2} \tag{9.59}$$
$$N = -l_{x_r}K_1\delta_{r1} - l_{x_r}K_2\delta_{r2} \tag{9.60}$$

The input model (9.7) for sway, roll and yaw is:

$$\boldsymbol{\tau} = \underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & l_{y_f} & -l_{y_f} \\ -l_{x_r} & -l_{x_r} & 0 & 0 \end{bmatrix}}_{\mathbf{T}} \cdot \underbrace{\mathrm{diag}\{K_1, K_2, K_3, K_4\}}_{\mathbf{K}} \mathbf{u} \tag{9.61}$$

where $\mathbf{u} = [\delta_{f1}, \delta_{f2}, \delta_{r1}, \delta_{r2}]^\top$. *Combined roll damping and yaw control suggests that* $\boldsymbol{\tau} = [*, K, N]^\top$, *where* K *and* N *are the controls while* $*$ *denotes that the force* Y *in sway is not controlled, but treated as an interaction force caused by the input* $\mathbf{u}$. *The reduced order model for combined roll and yaw control is:*

$$\bar{\boldsymbol{\tau}} = \begin{bmatrix} K \\ N \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & l_{y_f} & -l_{y_f} \\ -l_{x_r} & -l_{x_r} & 0 & 0 \end{bmatrix}}_{\bar{\mathbf{T}}} \cdot \underbrace{\mathrm{diag}\{K_1, K_2, K_3, K_4\}}_{\mathbf{K}} \mathbf{u} \tag{9.62}$$

indicating that 4 controls u_i $(i = 1, \ldots, 4)$ *can be used to generate* K *and* N. *The control inputs can be computed as* $\mathbf{u} = \mathbf{K}^{-1}\bar{\mathbf{T}}^{\dagger}\bar{\boldsymbol{\tau}}$, *that is:*

$$\mathbf{u} = \underbrace{\mathrm{diag}\{1/K_1, 1/K_2, 1/K_3, 1/K_4\}}_{\mathbf{K}^{-1}} \cdot \underbrace{\begin{bmatrix} 0 & -\frac{1}{2l_{x_r}} \\ 0 & -\frac{1}{2l_{x_r}} \\ \frac{1}{2l_{y_f}} & 0 \\ -\frac{1}{2l_{y_f}} & 0 \end{bmatrix}}_{\bar{\mathbf{T}}^{\dagger}} \underbrace{\begin{bmatrix} K \\ N \end{bmatrix}}_{\bar{\boldsymbol{\tau}}} \tag{9.63}$$

The generalized force vector is found as:

$$\boldsymbol{\tau} = \mathbf{T}\mathbf{K}\mathbf{u} = \mathbf{T}\mathbf{K}(\mathbf{K}^{-1}\bar{\mathbf{T}}^{\dagger}\bar{\boldsymbol{\tau}}) = \mathbf{T}\bar{\mathbf{T}}^{\dagger}\bar{\boldsymbol{\tau}} \tag{9.64}$$

while the resulting input vector in sway, roll, and yaw becomes:

$$\boldsymbol{\tau} = \left[-\frac{N}{l_{x_r}}, K, N\right]^\top \tag{9.65}$$

where K *and* N *are generated by the control system. Notice that the control input in yaw,* N, *contributes with an interaction force* $-N/l_{x_r}$ *in sway.*

Energy Optimal Criterion for Combined Fin and RRD Stabilization

It is possible to relate the LQ controllers for the models (9.55) and (9.56). This is demonstrated by considering a ship equipped with r_1 rudders and r_2 fins. The total number of actuators is $r = r_1 + r_2$, implying that $\mathbf{u} \in \mathbb{R}^r$. The DOFs considered are *sway, roll,* and *yaw,* that is, $n = 3$. Consequently, $\boldsymbol{\nu} = [v, p, r]^\top \in \mathbb{R}^n$. It is also assumed that the ship is at least fully actuated such that $r \geq n$.

An energy optimal criterion weighting the different control forces $\mathbf{f} = [f_1, ..., f_r]^\top$ against accurate tracking and roll damping is:

$$J = \min_{\mathbf{f}} \left\{ \frac{1}{2} \int_0^T (\mathbf{e}^\top \mathbf{Q}\mathbf{e} + \mathbf{f}^\top \mathbf{R}_f \mathbf{f})\, d\tau \right\} \tag{9.66}$$

where $\mathbf{e} = \mathbf{y} - \mathbf{y}_d$. The elements in $\mathbf{Q}$ =diag$\{Q_p, Q_r, Q_\phi, Q_\psi\} \geq 0$ are used to weight accurate steering against roll damping. The rudder and fin servos are weighted against each other by specifying the elements in $\mathbf{R}_f$=diag$\{R_{\delta 1}, R_{\delta 2}, \ldots, R_{\delta r_1}, R_{f1}, R_{f2}, \ldots, R_{fr_2}\} > 0$. If $r_1 = 0$ and $R_{\delta 1} = R_{\delta 2} = \ldots = R_{\delta r_1} = 0$ only fin stabilization is obtained (no rudders).

The performance index (9.66) can be rewritten as:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T (\mathbf{e}^\top \mathbf{Q} \mathbf{e} + \mathbf{u}^\top \underbrace{\mathbf{K}^\top \mathbf{R}_f \mathbf{K}}_{\mathbf{R}_u} \mathbf{u}) \, d\tau \right\} \tag{9.67}$$

which again can be transformed to (recall from Section 7.5 that $\mathbf{u} = \mathbf{K}^{-1}\mathbf{T}_w^\dagger \boldsymbol{\tau}$ and $\mathbf{K}$ is diagonal):

$$J = \min_{\boldsymbol{\tau}} \left\{ \frac{1}{2} \int_0^T (\mathbf{e}^\top \mathbf{Q} \mathbf{e} + \boldsymbol{\tau}^\top \underbrace{(\mathbf{T}_w^\dagger)^\top \mathbf{R}_u \mathbf{T}_w}_{\mathbf{R}_\tau} \boldsymbol{\tau}) \, d\tau \right\} \tag{9.68}$$

The difference of the criteria (9.66), (9.67), and (9.68) are that they weight control force $\mathbf{f}$, actuator inputs $\mathbf{u}$, and generalized force $\boldsymbol{\tau}$ against $\mathbf{e}$, respectively. The control weights are related by:

$$\mathbf{R}_u = \mathbf{K}^\top \mathbf{R}_f \mathbf{K}, \qquad \mathbf{R}_\tau = (\mathbf{T}_w^\dagger)^\top \mathbf{R}_f \mathbf{T}_w^\dagger \tag{9.69}$$

As mentioned earlier it is advantageous to solve for the optimal control force $\boldsymbol{\tau}$, that is Criterion (9.68), and then use control allocation to compute $\mathbf{u}$. The result is (see `alloc.m` in the Matlab^{TM} GNC Toolbox):

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{T}_w^\dagger \boldsymbol{\tau} \tag{9.70}$$

which gives us design freedom with respect to actuator failure since the state-space model (9.55) is independent of $\mathbf{K}$ and $\mathbf{T}$. Instead of specifying $\mathbf{R}_\tau$ in (9.68) it is more convenient to specify $\mathbf{R}_f$, that is the price of control force, and then apply the transformation (9.69).

The solution to the LQ problem (9.68) is (see Section 7.2.3):

$$\boldsymbol{\tau} = \mathbf{G}_1 \mathbf{x} + \mathbf{G}_2 \mathbf{y}_d \tag{9.71}$$

where

$$\mathbf{G}_1 = -[(\mathbf{T}_w^\dagger)^\top \mathbf{R}_f \mathbf{T}_w^\dagger]^{-1} \mathbf{B}^\top \mathbf{P}_\infty \tag{9.72}$$

$$\mathbf{G}_2 = -[(\mathbf{T}_w^\dagger)^\top \mathbf{R}_f \mathbf{T}_w^\dagger]^{-1} \mathbf{B}^\top (\mathbf{A} + \mathbf{B}\mathbf{G}_1)^{-\top} \mathbf{C}^\top \mathbf{Q} \tag{9.73}$$

where $\mathbf{Q}$ and $\mathbf{R}_f$ are design matrices while $\mathbf{P}_\infty = \mathbf{P}_\infty^\top > 0$ is given by:

$$\mathbf{P}_\infty \mathbf{A} + \mathbf{A}^\top \mathbf{P}_\infty - \mathbf{P}_\infty \mathbf{B}[(\mathbf{T}_w^\dagger)^\top \mathbf{R}_f \mathbf{T}_w^\dagger]^{-1} \mathbf{B}^\top \mathbf{P}_\infty + \mathbf{C}^\top \mathbf{Q} \mathbf{C} = 0 \tag{9.74}$$

9.4 Operability and Motion Sickness Incidence Criteria

Operability criteria for manual and intellectual work as well as motion sickness are important design criteria for evaluation of autopilot and roll damping systems. Sea sickness is especially important in high-speed craft and ships with high vertical accelerations.

9.4.1 Human Operability Limiting Criteria in Roll

Operability limiting criteria with regard to vertical and lateral accelerations and roll angle for the effectiveness of the crew and the passengers are given in Table 9.2.

Table 9.2: Criteria for effectivness of the crew (Faltinsen 1990).

Standard Deviation (Root Mean Square) Criteria			
Vertical acceleration ($\dot{w}$)	Lateral acceleration ($\dot{v}$)	Roll angle (ϕ)	Description of work
0.20 g	0.10 g	6.0 deg	Light manual work
0.15 g	0.07 g	4.0 deg	Heavy manual work
0.10 g	0.05 g	3.0 deg	Intellectual work
0.05 g	0.04 g	2.5 deg	Transit passengers
0.02 g	0.03 g	2.0 deg	Cruise liner

This gives an indication on what type of work that can be expected to be carried out for different roll angles/sea states.

9.4.2 Criterion for Motion Sickness Incidence (MSI)

In addition to operability limiting criteria passenger comfort can be evaluated with respect to motion sickness.

The ISO 2631-3:1985 Criterion for MSI

The International Standardization Organization (ISO) motion seasickness incidence criterion is reported in ISO 2631-1 (1997). This report replaces ISO 2631-3 (1985); see *http://www.iso.ch.* The most important factors for seasickness are vertical (heave) accelerations a_z (m/s^2), exposure time t (hours), and encounter frequency ω_e (rad/s).

The ISO standard proposes an MSI of 10% which means that 10% of the passengers become seasick during t hours. The MSI curves as function of exposure time are shown in Figure 9.4 where:

$$a_z(t,\omega_e) = \begin{cases} 0.5\sqrt{2/t} & \text{for} \quad 0.1 \text{ Hz} < \frac{\omega_e}{2\pi} \leq 0.315 \text{ Hz} \\ 0.5\sqrt{2/t} \cdot 6.8837 \left(\frac{\omega_e}{2\pi}\right)^{1.67} & \text{for} \quad 0.315 \text{ Hz} \leq \frac{\omega_e}{2\pi} \leq 0.63 \text{ Hz} \end{cases} \tag{9.75}$$

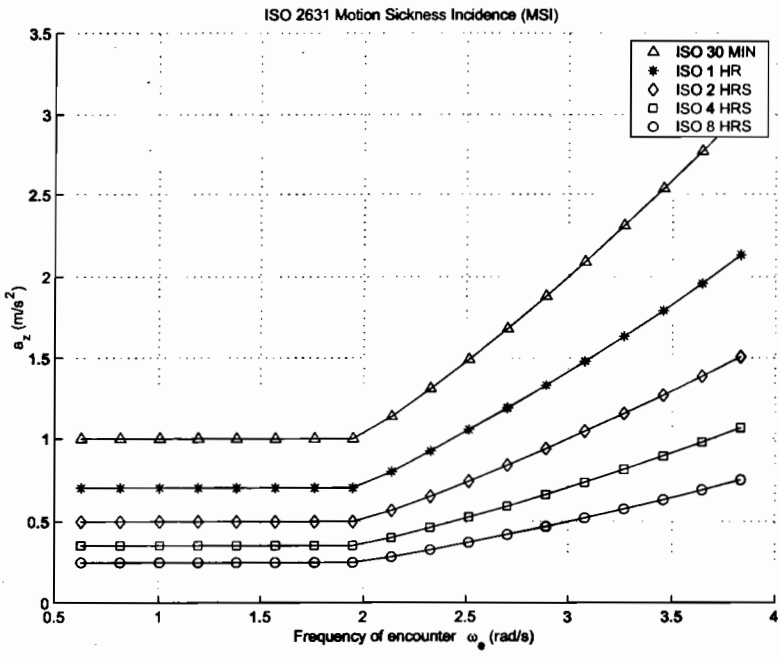


Figure 9.4: Heave acceleration a_z (m/s^2) as a function of frequency of encounter ω_e (rad/s) for different exposure times. The ISO curves represent a MSI of 10%.

Matlab:
The MSI curves (9.75) as functions of the exposure time are implemented in the GNC Toolbox as:

```
[a_z,w_e] = ISOmsi(t)
```

Figure 9.4 is generated by using the example file:

```
ExMSI
```

The main limitation of the ISO criterion is that it only predicts the exceedence of the 10% MSI point. It is also assumed that the accelerations in the CG are representative for the entire ship and that a representative wave period can be used instead of the actual wave. In many cases it is advantageous to use the extended sickness method for more accurate predictions. This method is presented below.

The O'Hanlon and McCauley (1974) Criterion for MSI

The O'Hanlon and McCauley (1974) probability integral method is attractive to use since it produces a MSI criterion in percent for combinations of heave acceleration a_z (m/s^2) and frequency of encounter ω_e (rad/s). The MSI index is defined as the number of sea sick people in percentage for an exposure time of two hours; see Lloyd (1989) and Lewis (1989).

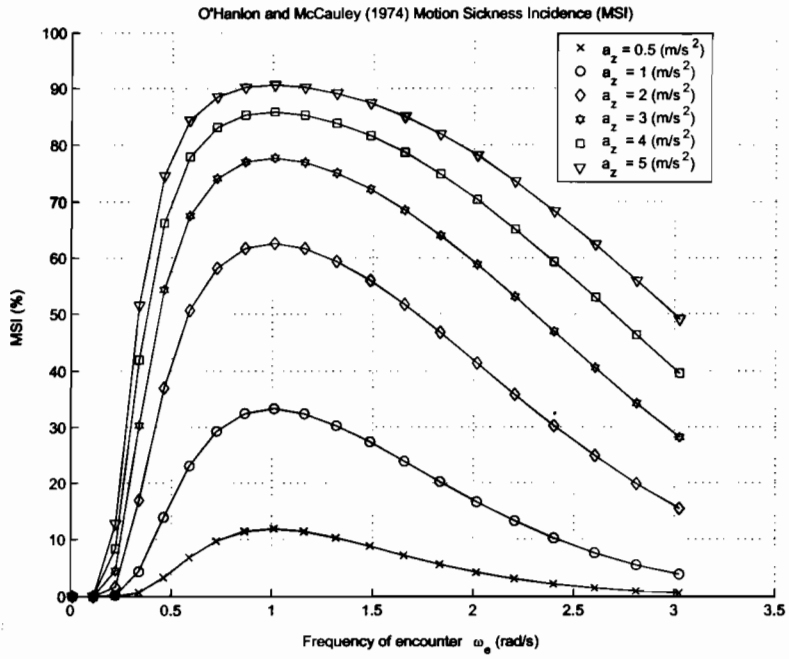


Figure 9.5: MSI is the number of motion sick persons in percent during a two hours exposure time as a function of encounter frequency ω_e (rad/s) and heave acceleration a_z (m/s^2).

The criterion is defined as:

$$\text{MSI} = 100\left(0.5 \pm \text{erf}\left(\frac{\pm \log_{10}(a_z/g) \mp \mu_{\text{MSI}}}{0.4}\right)\right) \quad (\%) \tag{9.76}$$

where

$$\mu_{\text{MSI}} = -0.819 + 2.32\,(\log_{10}\omega_e)^2 \tag{9.77}$$

and

$$\text{erf}(x) = \text{erf}(-x) = \frac{1}{\sqrt{2\pi}}\int_0^x \exp\left(-\frac{z^2}{2}\right)dz \tag{9.78}$$

Matlab:
The GNC Toolbox function:

```
msi = HMmsi(a_z,w_e)
```

can be used for computation of the MSI. Notice that the erf-function in `HMmsi.m` is scaled different from the Matlab^{TM} function `erf.m`. The MSI curves in Figure 9.5 are plotted for different a_z and ω_e using the example file:

```
ExMSI
```

The major drawback of the O'Hanlon and McCauley method is that it only applies to a two hour exposure time. Another effect to take into account is that the O'Hanlon and McCauley MSI criterion is derived from tests with young men seated separately in insulated cabins. According to the ISO 2631-1, the MSI number is about 1.5 higher amongst women and children suggesting that the actual MSI number for passengers of average age and sex distribution should be at least 1.25 times higher.

9.5 Exercises

Exercise 9.1 *Show that the pair* (x_g, z_g) *in* $\mathbf{r}_g = [x_g, y_g, z_g]^\top$ *describing the horizontal location of the body-fixed coordinate system in a ship can be chosen such that the inertia matrix* $\mathbf{I}_o = \mathbf{I}_o^\top$ *becomes diagonal. Assume homogeneous constant mass distribution and* xz*-plane symmetry such that* $I_{xy} = I_{yz} = 0$ *and* $y_g = 0$ *(Hint: use the parallel axes theorem* $\mathbf{I}_o = \mathbf{I}_c - m\mathbf{S}^2(\mathbf{r}_g^b)$*)*
Answer: $mx_g z_g = -I_{zx}^{cg}$ *where* x_g *or* z_g *must be zero. Usually* $x_g \neq 0$ *and* $z_g = 0$.

Exercise 9.2 *A high-speed craft is moving at forward speed* $U = 20.0$ *(m/s) in head sea. Assume that the mean vertical acceleration is* $a_z = 2.0$ *(m/s*^{2}*). The peak frequency of the waves is* $\omega_0 = 1.0$ *(rad/s). How many out of 86 passengers become seasick if the O'Hanlon and McCauley MSI criterion is used? Compute the MSI for different values* β *of incident waves and plot the MSI as a function of* β*. What are the most critical directions of the waves with respect to sea sickness and how many become seasick for these angles?*
Answers: $\beta = 180$ *deg (10 persons).* $\beta = 0$ *and* ± 90 *deg (54 persons).*

Chapter 10

Trajectory Tracking and Maneuvering Control

This chapter discusses *trajectory tracking* and *maneuvering control*. The objective in trajectory tracking control is to force the system output $\mathbf{y}(t)$ to track a desired output $\mathbf{y}_d(t)$, while in maneuvering control the task is to converge to and follow a parametrized path $\mathbf{y}_d(\theta)$ as a function of a path variable denoted by $\theta = \theta(t)$. In maneuvering control the objective is twofold: (1) converge to and follow a desired parametrized path $\mathbf{y}_d(\theta)$, and (2) satisfy a desired dynamic behavior along the path (Skjetne *et al.* 2002*c*), like a speed assignment for $\dot{\theta}(t)$. This is in accordance with Definitions 5.1 and 5.4 in Section 5.2.

10.1 Trajectory Tracking Control

Conventional way-point guidance systems are usually designed as trajectory tracking controllers. In its simplest form this involves the use of a classical autopilot system where the yaw angle command ψ_d is generated such that the *cross-track error* is minimized. This can be done in a multivariable controller, for instance of $\mathcal{H}_\infty$ or LQG type, or by including an additional tracking error control-loop in the autopilot. A way-point trajectory tracking system is usually designed such that the ship can move forward with reference speed U_d at the same time as the path cross-track error is minimized. The desired path can be generated using a route management system or by specifying the desired route by way-points, see Section 5.2. If weather data are available, the optimal route can be generated such that the effects of wind and water resistance are minimized.

When designing a 3 DOF trajectory tracking control system, the solution will depend on the number of available actuators. In this section the following cases are discussed:

- **2 controls:** Trajectory tracking control using forward thrust T for *speed control* and a single rudder δ to minimize the *cross-track error*. The proposed solutions include methods based on PID, line-of-sight, and LQG control for minimization of the cross-track error.
- **3 or more controls**: Low-speed trajectory tracking control of ships in *surge*, *sway*, and *yaw*. This is usually referred to as a dynamic positioning (DP) system and the topic is treated separately in Section 11.2.

In DP systems the surge and sway positions, and the yaw angle can be controlled independently, while trajectory tracking control with two controls is done by mapping the surge and sway tracking errors to cross-track errors. Such methods are presented below.

10.1.1 Conventional PID Cross-Tracking System

Recall from Section 5.2.1 that the cross-track error e_2 between a desired trajectory (x_d, y_d) and a moving vessel is:

$$e_2 = -(x - x_d)\sin\psi + (y - y_d)\cos\psi \tag{10.1}$$

For a vessel with coordinates (x, y) and heading ψ, the error term e_2 represents the deviation to the path in the y-direction–i.e., in vessel-paralell coordinates; see Definition 3.4 in Section 3.3.2.

Assumption 10.1 (Straight-Line Path)
Assume that the desired path is constructed as straight lines with the inertial frame x-axis along the path such that $x_d = x$ and $y_d = 0$.

Assumption 10.1 implies that v and ψ will be small for a vessel moving along the path. This implies that the cross-track error can be approximated as:

$$e_2 \overset{\psi \approx 0}{\approx} y \tag{10.2}$$

and that the kinematic equations reduce to:

$$\dot{x} = u\cos\psi - v\sin\psi \overset{v \approx 0 \text{ and } \psi \approx 0}{\approx} U \tag{10.3}$$

$$\dot{y} = u\sin\psi + v\cos\psi \overset{v \approx 0 \text{ and } \psi \approx 0}{\approx} U\psi \tag{10.4}$$

Consequently, the vessel is moving with approximately constant speed $U = \sqrt{u^2 + v^2} \approx u$ along the path.

A conventional cross-track controller is usually designed by using Nomoto's model (8.29) such that:

$$\dot{y} = U\psi \tag{10.5}$$

$$\dot{\psi} = r \tag{10.6}$$

$$T\dot{r} + r = K\delta + r_b \tag{10.7}$$

$$\dot{r}_b = 0 \tag{10.8}$$

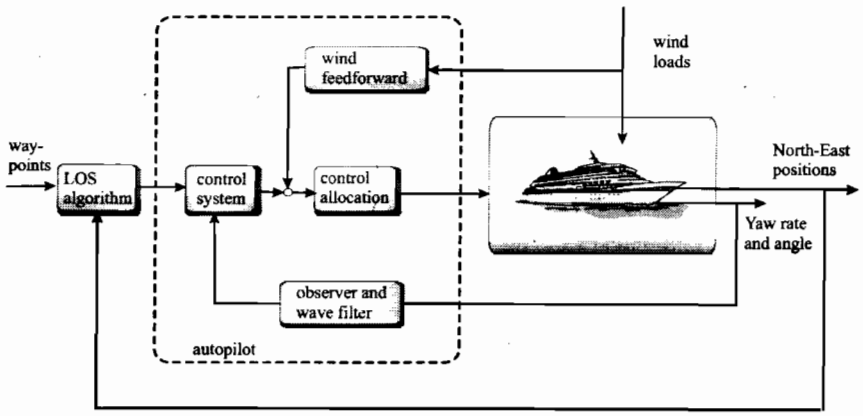


Figure 10.1: Autopilot including a LOS guidance system.

where r_b is a bias term and δ is the control input. Consequently:

$$e_2(s) = h_\delta(s)\delta(s) + h_b(s)r_b(s) \tag{10.9}$$

where:

$$h_\delta(s) = \frac{e_2}{\delta}(s) = \frac{KU}{s^2(1+Ts)} \tag{10.10}$$

$$h_b(s) = \frac{e_2}{r_b}(s) = \frac{U}{s^2(1+Ts)} \tag{10.11}$$

For this system it is straightforward to design a PID-controller for regulation of e_2 to zero. Integral action is needed in order to compensate for the bias term r_b representing environmental disturbances and the rudder offset.

10.1.2 Line of Sight Cross-Tracking System

An attractive method for path following removing Assumption 10.1 is to use a line-of-sight (LOS) guidance system, see Section 5.3. For a ship, the cross-track error e_2 is regulated to zero by designing the outer feedback loop and guidance system as:

$$\psi_d(t) = \text{atan2}\left(y_{\text{los}} - y(t), x_{\text{los}} - x(t)\right) \tag{10.12}$$

where the LOS coordinates $(x_{\text{los}}, y_{\text{los}})$ are implicitly given by:

$$(y_{\text{los}} - y(t))^2 + (x_{\text{los}} - x(t))^2 = (nL_{pp})^2 \tag{10.13}$$

$$\left(\frac{y_{\text{los}} - y_{k-1}}{x_{\text{los}} - x_{k-1}}\right) = \left(\frac{y_k - y_{k-1}}{x_k - x_{k-1}}\right) = \text{constant} \tag{10.14}$$

where n is the number of ship lengths L_{pp}.

A procedure for switching between the different way-points (x_k, y_k) for $(k = 1, \ldots, N)$ is presented in Section 5.3. The autopilot and LOS guidance systems are shown in Figure 10.1.

10.1.3 Linear Quadratic Optimal Cross-Tracking System

A linear quadratic Gaussian (LQG) optimal control system can be designed to minimize the cross-track error.

Vessel Parallel Coordinates

The LQG controller can be designed for the linear parameter varying (LPV) model:

$$\dot{\mathbf{x}} = \mathbf{A}(u_o)\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} + \mathbf{F}\boldsymbol{\nu}_o \tag{10.15}$$

where u_o is surge speed, $\mathbf{x} = [\boldsymbol{\eta}_p^\top, \Delta\boldsymbol{\nu}^\top]^\top$ is the state vector using VP coordinates, $\mathbf{u} = \boldsymbol{\tau}$ and $\boldsymbol{\nu}_o = [u_o, 0, 0, 0, 0, 0]^\top$, see Section 3.3.2 for details. For this model, u_o can be used as a LQG gain scheduling parameter. The VP coordinates are based on the assumption that the turning rate $\dot{\psi}$ is small which is less restrictive than Assumption 10.1.

NED Coordinates

An alternative to the VP model is to use NED coordinates–i.e., the coordinates (n, e) in (2.18) under the assumption that $\phi = \theta = 0$. Consequently:

$$\dot{n} = u\cos\psi + v\cos\psi \tag{10.16}$$
$$\dot{e} = u\sin\psi + v\cos\psi \tag{10.17}$$

The nonlinear kinematic equations are linearized under the assumptions that $\sin\psi \approx \psi$, $\cos\psi \approx 1$ and $v \approx 0$, implying that the NED coordinate system must be rotated such that x-axis points towards the next way-point to make ψ is small–i.e., Assumption 10.1 is valid (Holzhüter 1990). It then follows that:

$$\dot{n} = U + d_x \tag{10.18}$$
$$\dot{e} = U\psi + v + d_y \tag{10.19}$$

where $U = \sqrt{u^2 + v^2} \approx u$, and the two bias terms d_x and d_y are included to counteract for modeling errors due to the linearization. This model has been frequently used by Holzhüter and Schultze (1996), and Holzhüter (1997) In order to utilize optimal control and filtering theory, the resulting state-space model is written as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \tag{10.20}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{10.21}$$
$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{10.22}$$

where $\mathbf{x} = [\xi_w, \psi_w, n, e, \psi, u, v, r, v_b, r_b, d_x, d_y]^\top$ is the state vector and $\mathbf{u} = \delta$. The control objective is regulation of $\mathbf{y} = [e, \psi]^\top$ to zero while GPS position and gyro measurements imply that $\mathbf{z} = [n, e, \psi + \psi_w]^\top$. Notice that the state vector includes two states (ξ_w, ψ_w) for wave filtering, see Sections 6.1.3– 6.1.4. For the 2nd-order Nomoto model this gives:

$$\begin{bmatrix} \dot{\xi}_w \\ \dot{\psi}_w \\ \dot{n} \\ \dot{e} \\ \dot{\psi} \\ \dot{u} \\ \dot{v} \\ \dot{r} \\ \dot{v}_b \\ \dot{r}_b \\ \dot{d}_x \\ \dot{d}_y \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\omega_o^2 & -2\lambda\omega_o & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & U & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{11} & a_{12} & 0 & -a_{11} & -a_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{21} & a_{22} & 0 & -a_{21} & -a_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \xi_w \\ \psi_w \\ n \\ e \\ \psi \\ u \\ v \\ r \\ v_b \\ r_b \\ d_x \\ d_y \end{bmatrix}$$

$$+ \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ b_1 \\ b_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{B}} \delta + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ K_w & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{E}} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{bmatrix}$$

For this system:

$$\mathbf{y} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{C}} \mathbf{x} \tag{10.23}$$

$$\mathbf{z} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{H}} \mathbf{x} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{10.24}$$

where a_{ij} and b_i are defined in Section 8.1.2, and v_b and r_b are two slowly-varying parameters describing modeling errors and environmental disturbances.

In many applications only forward speed U, heading angle ψ and position (x, y) are measured. Often estimation of the sway velocity v is ill-conditioned. In such cases a simpler model structure, neglecting the sway dynamics, is recommended.

LQG Controller

The control objective is:

$$J = \min_{\mathbf{u}} \left\{ \frac{1}{2} \int_0^T \left(\mathbf{y}^\top \mathbf{Q}_c \mathbf{y} + \mathbf{u}^\top \mathbf{R}_c \mathbf{u} \right) d\tau \right\} \tag{10.25}$$

where $\mathbf{Q}_c = \mathbf{Q}_c^\top > 0$ and $\mathbf{R}_c > 0$ are the state and controller weight matrices.

The LQG controller is, see Section 7.2.1:

$$\mathbf{u} = \underbrace{-\mathbf{R}_c^{-1}\mathbf{B}^\top\mathbf{P}_c}_{\mathbf{G}}\hat{\mathbf{x}} \tag{10.26}$$

$$\dot{\mathbf{P}}_c = \mathbf{P}_c\mathbf{A} + \mathbf{A}^\top\mathbf{P}_c - \mathbf{P}_c\mathbf{B}\mathbf{R}_c^{-1}\mathbf{B}^\top\mathbf{P}_c + \mathbf{C}^\top\mathbf{Q}_c\mathbf{C} \tag{10.27}$$

where $\hat{\mathbf{x}}$ is the Kalman filter estimate of $\mathbf{x}$ given by, see Section 6.1.4:

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \underbrace{\mathbf{P}_f\mathbf{H}^\top\mathbf{R}_f^{-1}}_{\mathbf{K}}(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}) \tag{10.28}$$

$$\dot{\mathbf{P}}_f = \mathbf{A}\mathbf{P}_f + \mathbf{P}_f\mathbf{A}^\top + \mathbf{E}\mathbf{Q}_f\mathbf{E}^\top - \mathbf{P}_f\mathbf{H}^\top\mathbf{R}_f^{-1}\mathbf{H}\mathbf{P}_f \tag{10.29}$$

$\mathbf{Q}_f = \mathbf{Q}_f^\top > 0$ and $\mathbf{R}_f > 0$ are the filter weight matrices.

Optimal control using Bryson and Ho's time-varying controller for trajectory tracking and maneuvering control is discussed by Kvam *et al.* (2000), Duc-Hung *et al.* (2000), and Duc-Hung and Ohtsu (2000).

10.1.4 Underactuated Trajectory Tracking Control

In Pettersen and Nijmeijer (1999*a*) an underactuated tracking controller controlling n, e, and ψ is derived. Global exponential stability towards an arbitrarily small neighborhood of the reference trajectory (global exponential practical stability) is proven. Motivated by the similarities in structure between the ship model and chained form systems, it was shown in Pettersen and Nijmeijer (1999*b*) how a recursive technique for developing tracking controllers for chained form systems presented in Jiang and Nijmeijer (1999), can be used to develop a control law for the ship model with drift. Performing a change of coordinates to achieve model equations in a triangular-like form, and using integrator backstepping (Krstic *et al.* 1995), a continuous state feedback tracking control law was developed, giving semiglobal exponential stability of the desired trajectory. Sira-Ramirez (1999) used a combination of exact linearization and "high-gain" control to develop an output tracking controller for the ship, giving GES of the position trajectory. More recent results on underactuated tracking control are found in Pettersen and Nijmeijer (2001), Jiang (2002), and Do *et al.* (2002a, 2002b, 2002c).

10.2 Maneuvering Control

The primary goal in *maneuvering control* is to steer the vessel along a desired path. The speed assignment along the path is the secondary goal. These goals are solved as two separate tasks (Skjetne *et al.* 2002*c*). The first task is to reach and follow a desired path as a function of a scalar path variable θ, left as an extra degree of freedom for the second task. In the second task, θ is used as a state variable to satisfy an additional dynamic specification along the path. This setting is more general than the trajectory tracking problem, in which the auxiliary variable θ is assigned identically to a time signal $\theta(t)$.

Hauser and Hindmann (1995) introduced a procedure to design a *maneuver regulation* controller. To determine the *path variable* θ, they used a numerical projection from the current state onto the path. An already available tracking controller was then converted into a maneuver regulation controller, and a quadratic Lyapunov function was employed to guarantee that the states converge to the path and move along the path. This procedure applies to feedback linearizable systems, where the path is specified for the full state. Encarnacao and Pascoal (2001*b*) proposed an extension to solve the *output maneuvering problem* by backstepping. However, for systems of relative degree higher than two, their approach requires higher order time derivatives of θ.

In this section, the general maneuvering problem is divided into a *geometric task* and a *dynamic task* (Skjetne *et al.* 2002*c*). The geometric task is to reach the path and stay on it, while the dynamic task specifies a time, speed, or acceleration assignment along it. The proposed method is a design procedure solving a *robust maneuvering problem* for fully actuated systems in vectorial strict feedback form of any relative degree in presence of bounded disturbances or unknown parameters (Skjetne *et al.* 2002*d*). Extensions to adaptive systems and underactuated control–i.e., maneuvering control of ships in 3 DOF using only two controls are also discussed, while extensions to formation control are found in Skjetne *et al.* (2002*b*).

The robust maneuvering controller solves the geometric part of the problem in n recursive steps. It then proceeds to construct an update law that ties together the geometric design with the speed assignment v_s for $\dot{\theta}$. The speed assignment may depend on the path $v_s(\theta)$, or it may be given as an exogenous signal $v_s(t)$. For ships and underwater vehicles this gives design freedom since the speed profile can be specified along the path in real time.

Since the output path is not time-dependent, the primary control objectives are:

- to converge to and follow a desired parametrized output path $\mathbf{y}_d(\theta)$
- satisfy a desired dynamic behavior along the path for $\theta(t), \dot{\theta}(t)$, or $\ddot{\theta}(t)$

However, note that time may enter the closed loop through the dynamic assignment for $\theta(t), \dot{\theta}(t)$, or $\ddot{\theta}(t)$. This motivates the following definitions:

Definition 10.1 (Maneuvering Problem)
Given a path $\boldsymbol{\xi}(\theta)$ in the state space, the maneuvering problem is to design a controller that solves the following two tasks:

1. Geometric Task: *Given $\varepsilon_{GT} > 0$, force the state x to enter an ε_{GT} – neighborhood of the desired path $\xi(\theta)$, that is, $\exists T \geq 0$ such that:*

$$|\mathbf{x}(t) - \boldsymbol{\xi}(\theta(t))| \leq \varepsilon_{GT}, \quad \forall t \geq T \tag{10.30}$$

for any C^1 function $\theta(t)$.

2. Dynamic Assignment Task: *Given $\varepsilon_{DT} > 0$, satisfy one or more of the following assignments:*

2.1 Time Assignment: *Force the variable θ to enter an ε_{DT} – neighborhood of a desired time signal $v_t(t)$, that is, $\exists T \geq 0$ such that*

$$|\theta(t) - v_t(t)| \leq \varepsilon_{DT}, \quad \forall t \geq T \tag{10.31}$$

2.2 Speed Assignment: *Force the speed $\dot{\theta}$ to enter an ε_{DT} – neighborhood of a desired speed $v_s(\theta,t)$, that is, $\exists T \geq 0$ such that*

$$\left|\dot{\theta}(t) - v_s(\theta(t),t)\right| \leq \varepsilon_{DT}, \quad \forall t \geq T \tag{10.32}$$

2.3 Acceleration Assignment: *Force the acceleration $\ddot{\theta}$ to enter an ε_{DT} – neighborhood of a desired acceleration $v_a(\dot{\theta},\theta,t)$, that is, $\exists T \geq 0$ such that:*

$$\left|\ddot{\theta}(t) - v_a\left(\dot{\theta}(t),\theta(t),t\right)\right| \leq \varepsilon_{DT}, \quad \forall t \geq T \tag{10.33}$$

Definition 10.2 (Output Maneuvering Problem)
Given an output path $\mathbf{y}_d(\theta)$, the output maneuvering problem is to solve the maneuvering problem with respect to the output $\mathbf{y}$ rather than the state $\mathbf{x}$.

10.2.1 Robust Output Maneuvering

For simplicity it will be assumed that the dynamic assignment is a speed assignment (10.32). A more restrictive version of the speed assignment would be to satisfy (10.32) identically, $\dot{\theta} \equiv v_s$. In this case if $v_s = k$ such that $\theta(t) = \theta(0) + kt$, the maneuvering problem becomes a trajectory tracking problem where the desired output is a prespecified time signal $\mathbf{y}_d(t)$.

Consider the nonlinear plant in strict feedback form of vector relative degree n:

$$\begin{aligned}
\dot{\mathbf{x}}_1 &= \mathbf{G}_1(\bar{\mathbf{x}}_1)\mathbf{x}_2 + \mathbf{f}_1(\bar{\mathbf{x}}_1) + \mathbf{W}_1(\bar{\mathbf{x}}_1)\boldsymbol{\delta}_1(t) \\
\dot{\mathbf{x}}_2 &= \mathbf{G}_2(\bar{\mathbf{x}}_2)\mathbf{x}_3 + \mathbf{f}_2(\bar{\mathbf{x}}_2) + \mathbf{W}_2(\bar{\mathbf{x}}_2)\boldsymbol{\delta}_2(t) \\
&\vdots \\
\dot{\mathbf{x}}_n &= \mathbf{G}_n(\bar{\mathbf{x}}_n)\mathbf{u} + \mathbf{f}_n(\bar{\mathbf{x}}_n) + \mathbf{W}_n(\bar{\mathbf{x}}_n)\boldsymbol{\delta}_n(t) \\
\mathbf{y} &= \mathbf{h}(\mathbf{x}_1)
\end{aligned} \tag{10.34}$$

where $\mathbf{x}_i \in \mathbb{R}^m$ $(i = 1,\ldots,n)$ are the states, $\mathbf{y} \in \mathbb{R}^m$ is the output, $\mathbf{u} \in \mathbb{R}^m$ is the control, and $\boldsymbol{\delta}_i$ are unknown bounded disturbances. $\bar{\mathbf{x}}_i$ denotes the vector $\bar{\mathbf{x}}_i = \left[\mathbf{x}_1^\top, \mathbf{x}_2^\top, \ldots, \mathbf{x}_i^\top\right]^\top$. The matrices $\mathbf{G}_i(\bar{\mathbf{x}}_i)$ and $\nabla\mathbf{h}(\mathbf{x}_1) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_1}(\mathbf{x}_1)$ are invertible for all $\bar{\mathbf{x}}_i$, $\mathbf{h}$ is a diffeomorphism, and $\mathbf{G}_i$, $\mathbf{f}_i$, and $\mathbf{W}_i$ are locally Lipschitz.

The control objective is to design a maneuvering controller that solves the output maneuvering problem for a desired parametrized output path:

$$\mathbf{Y}_d = \{\mathbf{y} \in \mathbb{R}^m : \mathbf{y} = \mathbf{y}_d(\theta),\ \theta \in J_\theta \subseteq \mathbb{R}\} \tag{10.35}$$

where J_θ is the interval of definition, $\mathbf{y}_d(\theta)$ is n times differentiable with respect to θ, and the path characterization vector:

$$\mathbf{y}_{cv}(\theta) = [\mathbf{y}_d'(\theta)^\top, \mathbf{y}_d''(\theta)^\top, \mathbf{y}_d^{(3)}(\theta)^\top, \ldots, \mathbf{y}_d^{(n)}(\theta)^\top]^\top \tag{10.36}$$

is uniformly bounded in θ.

Recursive Design Procedure

A backstepping design is developed to solve the maneuvering problem for (10.34). The first two steps are given to show how to deal with $\dot{\theta}$ while the *i-th* Step is given in Table 10.1. The design procedure is based on adaptive tracking and the concept of tuning functions (Krstic *et al.* 1995).

Step 1:

The new variables:

$$\omega_s = v_s(\theta, t) - \dot{\theta} \tag{10.37}$$

$$\mathbf{z}_1 = \mathbf{y} - \mathbf{y}_d(\theta) \tag{10.38}$$

$$\mathbf{z}_i = \mathbf{x}_i - \boldsymbol{\alpha}_{i-1}, \qquad (i = 2, \ldots, n) \tag{10.39}$$

are introduced, where v_s is bounded and $\mathcal{C}^{n-1}$, and $\boldsymbol{\alpha}_{i-1}$ are virtual controls to be specified later. Differentiating (10.38) with respect to time results in:

$$\begin{aligned}\dot{\mathbf{z}}_1 &= \dot{\mathbf{y}} - \mathbf{y}_d'\dot{\theta} \\ &= \nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + \nabla\mathbf{h}\mathbf{G}_1\boldsymbol{\alpha}_1 + \nabla\mathbf{h}\mathbf{f}_1 + \nabla\mathbf{h}\mathbf{W}_1\boldsymbol{\delta}_1(t) - \boldsymbol{\nu}_1 v_s + \boldsymbol{\nu}_1\omega_s\end{aligned}$$

where $\boldsymbol{\nu}_1 = \mathbf{y}_d'(\theta) = \frac{\partial \mathbf{y}_d}{\partial \theta}$. Choose a Hurwitz matrix $\mathbf{A}_1$ so that $\mathbf{P}_1 = \mathbf{P}_1^\top > 0$ is the solution to $\mathbf{P}_1\mathbf{A}_1 + \mathbf{A}_1^\top\mathbf{P}_1 = -\mathbf{Q}_1 < 0$, and define the first CLF:

$$V_1 = \mathbf{z}_1^\top\mathbf{P}_1\mathbf{z}_1 \tag{10.40}$$

whose time derivative is:

$$\begin{aligned}\dot{V}_1 &= 2\mathbf{z}_1^\top\mathbf{P}_1\left[\nabla\mathbf{h}\mathbf{G}_1\boldsymbol{\alpha}_1 + \nabla\mathbf{h}\mathbf{f}_1 - \boldsymbol{\nu}_1 v_s\right] \\ &\quad + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{W}_1\boldsymbol{\delta}_1 + 2\mathbf{z}_1^\top\mathbf{P}_1\boldsymbol{\nu}_1\omega_s\end{aligned}$$

Then the first virtual control law is picked as:

$$\begin{aligned}\boldsymbol{\alpha}_1 &= \mathbf{G}_1^{-1}(\nabla\mathbf{h})^{-1}\left[\mathbf{A}_1\mathbf{z}_1 - \nabla\mathbf{h}\mathbf{f}_1 + \boldsymbol{\nu}_1 v_s + \boldsymbol{\alpha}_{10}\right] \\ &= \mathbf{f}_{\alpha_1}(\mathbf{x}_1, \theta, t)\end{aligned} \tag{10.41}$$

where $\boldsymbol{\alpha}_{10}$ is a nonlinear damping term to be picked. Define the first *tuning function*, $\tau_1 \in \mathbb{R}$, as:

$$\tau_1 = 2\mathbf{z}_1^\top\mathbf{P}_1\boldsymbol{\nu}_1 \tag{10.42}$$

After an application of Young's inequality, the derivative $\dot{V}_1$ becomes:

$$\begin{aligned}\dot{V}_1 &= -\mathbf{z}_1^\top\mathbf{Q}_1\mathbf{z}_1 + \tau_1\omega_s + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{W}_1\boldsymbol{\delta}_1 + 2\mathbf{z}_1^\top\mathbf{P}_1\boldsymbol{\alpha}_{10} \\ &\leq -\mathbf{z}_1^\top\mathbf{Q}_1\mathbf{z}_1 + \tau_1\omega_s + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 \\ &\quad + 2\mathbf{z}_1^\top\mathbf{P}_1\left\{\boldsymbol{\alpha}_{10} + \frac{1}{2}\kappa_1(\nabla\mathbf{h})\mathbf{W}_1\mathbf{W}_1^\top(\nabla\mathbf{h})^\top\mathbf{P}_1\mathbf{z}_1\right\} + \frac{1}{\kappa_1}\boldsymbol{\delta}_1^\top\boldsymbol{\delta}_1\end{aligned}$$

The nonlinear damping term is now chosen as:

$$\boldsymbol{\alpha}_{10} = -\frac{1}{2}\kappa_1(\nabla\mathbf{h})\mathbf{W}_1\mathbf{W}_1^\top(\nabla\mathbf{h})^\top\mathbf{P}_1\mathbf{z}_1, \qquad \kappa_1 > 0 \tag{10.43}$$

and the result of Step 1 is:

$$\dot{V}_1 \leq -\mathbf{z}_1^\top \mathbf{Q}_1 \mathbf{z}_1 + \boldsymbol{\Delta}_1^\top \mathbf{K}_1 \boldsymbol{\Delta}_1 + 2\mathbf{z}_1^\top \mathbf{P}_1 \nabla\mathbf{h}\mathbf{G}_1 \mathbf{z}_2 + \tau_1 \omega_s \tag{10.44}$$

$$\dot{\mathbf{z}}_1 = \mathbf{A}_1 \mathbf{z}_1 - \frac{1}{2}\kappa_1 (\nabla\mathbf{h}) \mathbf{W}_1 \mathbf{W}_1^\top (\nabla\mathbf{h})^\top \mathbf{P}_1 \mathbf{z}_1 + \nabla\mathbf{h}\mathbf{G}_1 \mathbf{z}_2 + \nabla\mathbf{h}\mathbf{W}_1 \boldsymbol{\delta}_1 + \boldsymbol{\nu}_1 \omega_s \tag{10.45}$$

where $\boldsymbol{\Delta}_1 = \boldsymbol{\delta}_1$ and $\mathbf{K}_1 = 1/\kappa_1$. If this were the last step, then $\mathbf{z}_2 = \mathbf{0}$ and $\omega_s = 0$ would reduce (10.44) to:

$$\dot{V}_1 \leq -\underline{q_1} |\mathbf{z}_1|^2 + \bar{k}_1 |\boldsymbol{\Delta}_1|^2 < 0, \quad \forall |\mathbf{z}_1| > \sqrt{\frac{\bar{k}_1}{\underline{q_1}}} |\boldsymbol{\Delta}_1|$$

where $\underline{q_1} = \lambda_{\min}(\mathbf{Q}_1)$, and $\bar{k}_1 = 1/\kappa_1$, which implies ISS from the disturbance $\boldsymbol{\delta}_1$ to the state $\mathbf{z}_1$. To aid next step, let:

$$\dot{\boldsymbol{\alpha}}_1 = \boldsymbol{\sigma}_1 + \boldsymbol{\nu}_2 \dot{\theta} + \boldsymbol{\varpi}_{1,1} \boldsymbol{\delta}_1 \tag{10.46}$$

where $\boldsymbol{\sigma}_1$ collects the terms in $\dot{\boldsymbol{\alpha}}_1$ not containing $\dot{\theta}$ and $\boldsymbol{\delta}_1$, $\boldsymbol{\nu}_2$ the terms multiplying $\dot{\theta}$, and $\boldsymbol{\varpi}_{1,1}$ the terms multiplying the disturbance $\boldsymbol{\delta}_1$:

$$\boldsymbol{\sigma}_1 = \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{x}_1} [\mathbf{G}_1 \mathbf{x}_2 + \mathbf{f}_1] + \frac{\partial \boldsymbol{\alpha}_1}{\partial t} \tag{10.47}$$

$$\boldsymbol{\nu}_2 = \boldsymbol{\alpha}_1' = \frac{\partial \boldsymbol{\alpha}_1}{\partial \theta} \tag{10.48}$$

$$\boldsymbol{\varpi}_{1,1} = \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{x}_1} \mathbf{W}_1 \tag{10.49}$$

Step 2:

Consider $\mathbf{x}_3$ as the next control variable. Differentiating (10.39), $i = 2$, with respect to time gives:

$$\begin{aligned} \dot{\mathbf{z}}_2 &= \dot{\mathbf{x}}_2 - \dot{\boldsymbol{\alpha}}_1 \\ &= \mathbf{G}_2 \mathbf{z}_3 + \mathbf{G}_2 \boldsymbol{\alpha}_2 + \mathbf{f}_2 + \mathbf{W}_2 \boldsymbol{\delta}_2(t) - \boldsymbol{\sigma}_1 - \boldsymbol{\nu}_2 \dot{\theta} - \boldsymbol{\varpi}_{1,1} \boldsymbol{\delta}_1(t) \end{aligned}$$

where $\mathbf{z}_3 = \mathbf{x}_3 - \boldsymbol{\alpha}_2$. Choose $\mathbf{A}_2$ Hurwitz and let $\mathbf{P}_2 = \mathbf{P}_2^\top > 0$ be the solution to $\mathbf{P}_2 \mathbf{A}_2 + \mathbf{A}_2^\top \mathbf{P}_2 = -\mathbf{Q}_2 < 0$. Define the Step 2 CLF:

$$V_2 = V_1 + \mathbf{z}_2^\top \mathbf{P}_2 \mathbf{z}_2 \tag{10.50}$$

whose time derivative is:

$$\begin{aligned} \dot{V}_2 \leq & -\mathbf{z}_1^\top \mathbf{Q}_1 \mathbf{z}_1 + \tau_1 \omega_s + 2\mathbf{z}_2^\top \mathbf{P}_2 \boldsymbol{\nu}_2 \omega_s + \frac{1}{\kappa_1} \boldsymbol{\delta}_1^\top \boldsymbol{\delta}_1 + 2\mathbf{z}_2^\top \mathbf{P}_2 [\mathbf{W}_2 \boldsymbol{\delta}_2 - \boldsymbol{\varpi}_{1,1} \boldsymbol{\delta}_1] \\ & + 2\mathbf{z}_2^\top \left\{ \mathbf{G}_1^\top (\nabla\mathbf{h})^\top \mathbf{P}_1 \mathbf{z}_1 + \mathbf{P}_2 [\mathbf{G}_2 \boldsymbol{\alpha}_2 + \mathbf{f}_2 - \boldsymbol{\sigma}_1 - \boldsymbol{\nu}_2 v_s] \right\} + 2\mathbf{z}_2^\top \mathbf{P}_2 \mathbf{G}_2 \mathbf{z}_3 \end{aligned}$$

The second virtual control law is picked as:

$$\begin{aligned} \boldsymbol{\alpha}_2 &= \mathbf{G}_2^{-1} \left[\mathbf{A}_2 \mathbf{z}_2 - \mathbf{P}_2^{-1} \mathbf{G}_1^\top (\nabla\mathbf{h})^\top \mathbf{P}_1 \mathbf{z}_1 - \mathbf{f}_2 + \boldsymbol{\sigma}_1 + \boldsymbol{\nu}_2 v_s + \boldsymbol{\alpha}_{20} \right] \\ &= \mathbf{f}_{\alpha_2}(\bar{\mathbf{x}}_2, \theta, t) \end{aligned} \tag{10.51}$$

where α_{20} is a nonlinear damping term to be designed. Define the second tuning function, $\tau_2 \in \mathbb{R}$, as:

$$\tau_2 = \tau_1 + 2\mathbf{z}_2^\top \mathbf{P}_2 \boldsymbol{\nu}_2 \tag{10.52}$$

Using Young's inequality again, the derivative $\dot{V}_2$ becomes:

$$\begin{aligned}\dot{V}_2 \leq & -\mathbf{z}_1^\top \mathbf{Q}_1 \mathbf{z}_1 - \mathbf{z}_2^\top \mathbf{Q}_2 \mathbf{z}_2 + \tau_2 \omega_s + \frac{1}{\kappa_1}\boldsymbol{\delta}_1^\top \boldsymbol{\delta}_1 + 2\mathbf{z}_2^\top \mathbf{P}_2 \mathbf{G}_2 \mathbf{z}_3 \\ & + 2\mathbf{z}_2^\top \mathbf{P}_2 \left\{ \boldsymbol{\alpha}_{20} + \frac{1}{2}\kappa_2 \left[\mathbf{W}_2 \mathbf{W}_2^\top + \boldsymbol{\varpi}_{1,1} \boldsymbol{\varpi}_{1,1}^\top \right] \mathbf{P}_2 \mathbf{z}_2 \right\} + \frac{1}{\kappa_2} \left[\boldsymbol{\delta}_2^\top \boldsymbol{\delta}_2 + \boldsymbol{\delta}_1^\top \boldsymbol{\delta}_1 \right]\end{aligned}$$

and the second nonlinear damping term $\boldsymbol{\alpha}_{20}$ is then chosen as:

$$\boldsymbol{\alpha}_{20} = -\frac{1}{2}\kappa_2 \left[\mathbf{W}_2 \mathbf{W}_2^\top + \boldsymbol{\varpi}_{1,1} \boldsymbol{\varpi}_{1,1}^\top \right] \mathbf{P}_2 \mathbf{z}_2, \quad \kappa_2 > 0 \tag{10.53}$$

Then Step 2 results in:

$$\dot{V}_2 \leq -\mathbf{z}_1^\top \mathbf{Q}_1 \mathbf{z}_1 - \mathbf{z}_2^\top \mathbf{Q}_2 \mathbf{z}_2 + \tau_2 \omega_s + \boldsymbol{\Delta}_2^\top \mathbf{K}_2 \boldsymbol{\Delta}_2 + 2\mathbf{z}_2^\top \mathbf{P}_2 \mathbf{G}_2 \mathbf{z}_3 \tag{10.54}$$

$$\begin{aligned}\dot{\mathbf{z}}_2 = & -\mathbf{P}_2^{-1} \mathbf{G}_1^\top \left(\nabla \mathbf{h} \right)^\top \mathbf{P}_1 \mathbf{z}_1 + \mathbf{A}_2 \mathbf{z}_2 - \frac{1}{2}\kappa_2 \left[\mathbf{W}_2 \mathbf{W}_2^\top + \boldsymbol{\varpi}_{1,1} \boldsymbol{\varpi}_{1,1}^\top \right] \mathbf{P}_2 \mathbf{z}_2 \\ & + \mathbf{G}_2 \mathbf{z}_3 + \boldsymbol{\nu}_2 \omega_s - \boldsymbol{\varpi}_{1,1} \boldsymbol{\delta}_1 + \mathbf{W}_2 \boldsymbol{\delta}_2\end{aligned} \tag{10.55}$$

where $\boldsymbol{\Delta}_2 = [\boldsymbol{\delta}_1^\top, \ \boldsymbol{\delta}_2^\top]^\top$ and $\mathbf{K}_2 = \text{diag}\{\frac{1}{\kappa_1} + \frac{1}{\kappa_2}, \frac{1}{\kappa_2}\}$. If this was the last step, then $\mathbf{z}_3 = \mathbf{0}$ and $\omega_s = 0$ would give:

$$\dot{V}_2 \leq -\underline{q_2} \left| \bar{\mathbf{z}}_2 \right|^2 + \bar{k}_2 \left| \boldsymbol{\Delta}_2 \right|^2 < 0, \quad \forall \left| \bar{\mathbf{z}}_2 \right| > \sqrt{\frac{\bar{k}_2}{\underline{q_2}}} \left| \boldsymbol{\Delta}_2 \right|$$

where $\underline{q_2} = \lambda_{\min} \left(\mathbf{Q}_1, \mathbf{Q}_2 \right), \bar{k}_2 = 1/\kappa_1 + 1/\kappa_2$, which indicates ISS from the disturbances $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2$ to the states $\mathbf{z}_1, \mathbf{z}_2$.

In aid of next step, differentiating $\boldsymbol{\alpha}_2$ gives:

$$\dot{\boldsymbol{\alpha}}_2 = \boldsymbol{\sigma}_2 + \boldsymbol{\nu}_3 \dot{\theta} + \boldsymbol{\varpi}_{2,1} \boldsymbol{\delta}_1 + \boldsymbol{\varpi}_{2,2} \boldsymbol{\delta}_2 \tag{10.56}$$

where

$$\boldsymbol{\sigma}_2 = \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_1} \left[\mathbf{G}_1 \mathbf{x}_2 + \mathbf{f}_1 \right] + \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_2} \left[\mathbf{G}_2 \mathbf{x}_3 + \mathbf{f}_2 \right] + \frac{\partial \boldsymbol{\alpha}_2}{\partial t} \tag{10.57}$$

$$\boldsymbol{\nu}_3 = \boldsymbol{\alpha}_2' = \frac{\partial \boldsymbol{\alpha}_2}{\partial \theta} \tag{10.58}$$

$$\boldsymbol{\varpi}_{2,1} = \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_1} \mathbf{W}_1, \qquad \boldsymbol{\varpi}_{2,2} = \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_2} \mathbf{W}_2 \tag{10.59}$$

Step i:

Table 10.1 shows the general *i-th* Step of the backstepping procedure.

Table 10.1: Robust Maneuvering: Steps i = 3,...,n.

$$\dot{\mathbf{z}}_i = \dot{\mathbf{x}}_i - \dot{\boldsymbol{\alpha}}_{i-1} = \mathbf{G}_i(\bar{\mathbf{x}}_i)\,\mathbf{z}_{i+1} + \mathbf{G}_i(\bar{\mathbf{x}}_i)\,\boldsymbol{\alpha}_i + \mathbf{f}_i(\bar{\mathbf{x}}_i) + \mathbf{W}_i(\bar{\mathbf{x}}_i)\,\boldsymbol{\delta}_i(t) - \boldsymbol{\sigma}_{i-1} - \boldsymbol{\nu}_i\dot{\theta} - \sum_{j=1}^{i-1}\boldsymbol{\varpi}_{i-1,j}\boldsymbol{\delta}_j$$

$$V_i = V_{i-1} + \mathbf{z}_i^\top \mathbf{P}_i \mathbf{z}_i$$

$$\dot{V}_i \le -\sum_{j=1}^{i-1}\mathbf{z}_j^\top \mathbf{Q}_j \mathbf{z}_j + \sum_{j=1}^{i-1}\sum_{k=j}^{i-1}\frac{1}{\kappa_k}\boldsymbol{\delta}_j^\top\boldsymbol{\delta}_j + 2\mathbf{z}_i^\top\left\{\mathbf{G}_{i-1}^\top \mathbf{P}_{i-1}\mathbf{z}_{i-1} + \mathbf{P}_i\left[\mathbf{G}_i\boldsymbol{\alpha}_i + \mathbf{f}_i - \boldsymbol{\sigma}_{i-1} - \boldsymbol{\nu}_i v_s\right]\right\} + 2\mathbf{z}_i^\top\mathbf{P}_i\left[\mathbf{W}_i\boldsymbol{\delta}_i - \sum_{j=1}^{i-1}\boldsymbol{\varpi}_{i-1,j}\boldsymbol{\delta}_j\right] + 2\mathbf{z}_i^\top\mathbf{P}_i\boldsymbol{\nu}_i\omega_s + \tau_{i-1}\omega_s + 2\mathbf{z}_i^\top\mathbf{P}_i\mathbf{G}_i\mathbf{z}_{i+1}$$

$$\boldsymbol{\alpha}_i = \mathbf{G}_i^{-1}\left[\mathbf{A}_i\mathbf{z}_i - \mathbf{P}_i^{-1}\mathbf{G}_{i-1}^\top\mathbf{P}_{i-1}\mathbf{z}_{i-1} - \mathbf{f}_i + \boldsymbol{\sigma}_{i-1} + \boldsymbol{\nu}_i v_s + \boldsymbol{\alpha}_{i0}\right] = \mathbf{f}_{\alpha_i}(\bar{\mathbf{x}}_i, \theta, t)$$

$$\boldsymbol{\alpha}_{i0} = -\tfrac{1}{2}\kappa_i\left[\mathbf{W}_i\mathbf{W}_i^\top + \sum_{j=1}^{i-1}\boldsymbol{\varpi}_{i-1,j}\boldsymbol{\varpi}_{i-1,j}^\top\right]\mathbf{P}_i\mathbf{z}_i, \quad \kappa_i > 0$$

$$\mathbf{P}_i\mathbf{A}_i + \mathbf{A}_i^\top\mathbf{P}_i = -\mathbf{Q}_i < 0$$

$$\tau_i = \tau_{i-1} + 2\mathbf{z}_i^\top\mathbf{P}_i\boldsymbol{\nu}_i \quad \text{i-th Tuning Function}$$

$$\Downarrow$$

$$\dot{V}_i \le -\sum_{j=1}^{i}\mathbf{z}_j^\top\mathbf{Q}_j\mathbf{z}_j + \tau_i\omega_s + 2\mathbf{z}_i^\top\mathbf{P}_i\mathbf{G}_i\mathbf{z}_{i+1} + \boldsymbol{\Delta}_i^\top\mathbf{K}_i\boldsymbol{\Delta}_i$$

$$\dot{\mathbf{z}}_i = -\mathbf{P}_i^{-1}\mathbf{G}_{i-1}^\top\mathbf{P}_{i-1}\mathbf{z}_{i-1} + \mathbf{A}_i\mathbf{z}_i - \tfrac{1}{2}\kappa_i\left[\mathbf{W}_i\mathbf{W}_i^\top + \sum_{j=1}^{i-1}\boldsymbol{\varpi}_{i-1,j}\boldsymbol{\varpi}_{i-1,j}^\top\right]\mathbf{P}_i\mathbf{z}_i + \mathbf{G}_i\mathbf{z}_{i+1} + \boldsymbol{\nu}_i\omega_s + \mathbf{W}_i\boldsymbol{\delta}_i - \sum_{j=1}^{i-1}\boldsymbol{\varpi}_{i-1,j}\boldsymbol{\delta}_j$$

$$\boldsymbol{\Delta}_i = \left[\boldsymbol{\delta}_1^\top\ \boldsymbol{\delta}_2^\top\ \cdots\ \boldsymbol{\delta}_i^\top\right], \quad \mathbf{K}_i := \mathrm{diag}\left\{\tfrac{1}{\kappa_1} + \tfrac{1}{\kappa_2} + \ldots + \tfrac{1}{\kappa_i}, \tfrac{1}{\kappa_2} + \ldots + \tfrac{1}{\kappa_i}, \cdots, \tfrac{1}{\kappa_i}\right\}$$

$$\dot{\boldsymbol{\alpha}}_i = \boldsymbol{\sigma}_i + \boldsymbol{\nu}_{i+1}\dot{\theta} + \sum_{j=1}^{i}\boldsymbol{\varpi}_{i,j}\boldsymbol{\delta}_j$$

$$\boldsymbol{\sigma}_i = \tfrac{\partial\boldsymbol{\alpha}_i}{\partial\mathbf{x}_1}\left[\mathbf{G}_1\mathbf{x}_2 + \mathbf{f}_1\right] + \ldots + \tfrac{\partial\boldsymbol{\alpha}_i}{\partial\mathbf{x}_i}\left[\mathbf{G}_i\mathbf{x}_{i+1} + \mathbf{f}_i\right] + \tfrac{\partial\boldsymbol{\alpha}_1}{\partial t}$$

$$\boldsymbol{\nu}_{i+1} = \boldsymbol{\alpha}_i'$$

$$\boldsymbol{\varpi}_{i,j} = \tfrac{\partial\boldsymbol{\alpha}_i}{\partial\mathbf{x}_j}\mathbf{W}_j$$

Step n:

Upon the completion of Step n, the choice for the control law:

$$\begin{aligned}\mathbf{u} &= \boldsymbol{\alpha}_n(\bar{\mathbf{x}}_n, \theta, t)\\ &= \mathbf{G}_n^{-1}[\mathbf{A}_n\mathbf{z}_n - \mathbf{P}_n^{-1}\mathbf{G}_{n-1}^\top\mathbf{P}_{n-1}\mathbf{z}_{n-1} - \mathbf{f}_n + \boldsymbol{\sigma}_{n-1} + \boldsymbol{\nu}_n v_s + \mathbf{u}_0] \qquad (10.60)\end{aligned}$$

$$\mathbf{u}_0 = -\frac{1}{2}\kappa_n\left[\mathbf{W}_n\mathbf{W}_n^\top + \sum_{j=1}^{n-1}\boldsymbol{\varpi}_{n-1,j}\boldsymbol{\varpi}_{n-1,j}^\top\right]\mathbf{P}_n\mathbf{z}_n, \qquad \kappa_n > 0 \qquad (10.61)$$

results in:

$$\dot{V}_n \le -\sum_{j=1}^{n}\mathbf{z}_j^\top\mathbf{Q}_j\mathbf{z}_j + \tau_n\omega_s + \boldsymbol{\Delta}_n^\top\mathbf{K}_n\boldsymbol{\Delta}_n \qquad (10.62)$$

$$\dot{\mathbf{z}}_n = -\mathbf{P}_n^{-1}\mathbf{G}_{n-1}^{\top}\mathbf{P}_{n-1}\mathbf{z}_{n-1} + \mathbf{A}_n\mathbf{z}_n - \frac{1}{2}\kappa_n\left[\mathbf{W}_n\mathbf{W}_n^{\top} + \sum_{j=1}^{n-1}\boldsymbol{\varpi}_{n-1,j}\boldsymbol{\varpi}_{n-1,j}^{\top}\right]\mathbf{P}_n\mathbf{z}_n$$
$$+ \boldsymbol{\nu}_n\omega_s + \mathbf{W}_n\boldsymbol{\delta}_n - \sum_{j=1}^{n-1}\boldsymbol{\varpi}_{n-1,j}\boldsymbol{\delta}_j \tag{10.63}$$

This implies that, if we let $\omega_s \equiv 0$, which renders $\dot{\theta} \equiv v_s$, then the system in the z-coordinates is an ISS system from the disturbances $\boldsymbol{\delta}_i$ to the error state $\bar{\mathbf{z}}_n$ where the gains κ_i are used to attenuate the disturbances. According to Jiang *et al.* (1994), one can assign the gain from the disturbances $\boldsymbol{\delta}_i$ to the output error $\mathbf{z}_1 = \mathbf{y} - \mathbf{y}_d$ to ensure any desired maneuvering accuracy.

In the case of no disturbances, $\boldsymbol{\delta}_i = \mathbf{0}$, one can set $\kappa_i = 0$ $(i = 1, \ldots, n)$. Then $\omega_s \equiv 0$ would render the equilibrium $\bar{\mathbf{z}}_n = 0$ UGES, but would not take advantage of the asymptotic formulation of the speed assignment task (10.32). For this task an extension of the above procedure is proposed.

Speed Assignment Task

To solve the speed assignment task (10.32), augment the Step n CLF to:

$$V = V_n + \frac{1}{2\mu_1}\omega_s^2, \qquad \mu_1 > 0 \tag{10.64}$$

Its derivative along the solution of the closed-loop dynamics in the z-coordinates is:

$$\dot{V} = \dot{V}_n + \frac{1}{\mu_1}\omega_s\dot{\omega}_s$$
$$\leq -\sum_{j=1}^{n}\mathbf{z}_j^{\top}\mathbf{Q}_j\mathbf{z}_j + \left[\tau_n + \frac{1}{\mu_1}\dot{\omega}_s\right]\omega_s + \boldsymbol{\Delta}_n^{\top}\mathbf{K}_n\boldsymbol{\Delta}_n$$

where the update law for $\dot{\omega}_s$ is yet to be constructed. To deal with the sign indefinite tuning function τ_n, choose:

$$\dot{\omega}_s = -\lambda\omega_s - \mu_1\tau_n, \qquad \lambda > 0 \tag{10.65}$$

which gives:

$$\dot{V} \leq -\mathbf{z}^{\top}\mathbf{Q}\mathbf{z} - \frac{\lambda}{\mu_1}\omega_s^2 + \boldsymbol{\Delta}^{\top}\mathbf{K}\boldsymbol{\Delta} \tag{10.66}$$

where $\mathbf{z} = [\mathbf{z}_1^{\top}, \ldots, \mathbf{z}_\eta^{\top}]^{\top}$, $\mathbf{Q} = \text{diag}\{\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_n\}$, $\boldsymbol{\Delta} = \boldsymbol{\Delta}_n$ and $\mathbf{K} = \mathbf{K}_n$. In the disturbance-free case, V is negative definite, and as $t \to \infty$, $\omega_s(t) = v_s(\theta(t), t) - \dot{\theta}(t) \to 0$, so that the speed assignment (10.32) is satisfied with exponential convergence. From (10.65) it is seen that $\omega_s(t)$ and $\mathbf{y}_d(\theta(t))$ are dependent on the system state $\mathbf{x}(t)$ through the final tuning function τ_n. Hence, feedback from the system state $\mathbf{x}$ is introduced to shape the desired output $\mathbf{y}_d(\theta)$ as a function of time. If, instead, we set $\omega_s \equiv 0$, we return to the tracking problem and loose this degree of freedom.

The interconnected system can be divided into three parts; the plant, the controller, and the guidance system as shown in Figure 10.2. The controller incorporates the dynamic equations (10.37), rewritten here as:

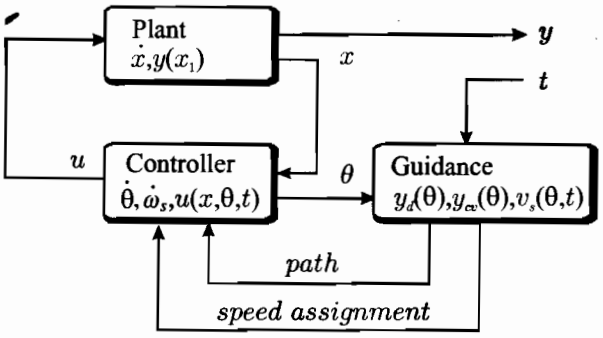


Figure 10.2: The main parts of the closed-loop system.

$$\dot{\theta} = v_s(\theta, t) - \omega_s \tag{10.67}$$

and (10.65). The control signal u is applied to the plant while the state θ is used by the guidance system. The guidance system incorporates the path definition (10.35), the path characterization vector (10.36), and the speed assignment $v_s(\theta, t)$. For a given θ and exogenous t it returns to the controller the necessary path and speed assignment signals.

Error Dynamics:

To rewrite the closed-loop system in vector form, let $\mathbf{A}_z \in \mathbb{R}^{nm \times nm}$, $\mathbf{F} \in \mathbb{R}^{nm \times nm}$, and $\mathbf{W} \in \mathbb{R}^{nm \times nm}$ be defined by:

$$\mathbf{A}_z(\bar{\mathbf{x}}_n, \theta, t) = \begin{bmatrix} \mathbf{A}_1 + \boldsymbol{\rho}_1 & \nabla \mathbf{h} \mathbf{G}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ -\mathbf{P}_2^{-1} \mathbf{G}_1^\top (\nabla \mathbf{h})^\top \mathbf{P}_1 & \mathbf{A}_2 + \boldsymbol{\rho}_2 & \mathbf{G}_2 & \cdots & \mathbf{0} \\ \mathbf{0} & -\mathbf{P}_3^{-1} \mathbf{G}_2^\top \mathbf{P}_2 & \mathbf{A}_3 + \boldsymbol{\rho}_3 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & -\mathbf{P}_n^{-1} \mathbf{G}_{n-1}^\top \mathbf{P}_{n-1} & \mathbf{A}_n + \boldsymbol{\rho}_n \end{bmatrix}$$

where:

$$\boldsymbol{\rho}_1 = -\frac{1}{2} \kappa_1 (\nabla \mathbf{h}) \mathbf{W}_1 \mathbf{W}_1^\top (\nabla \mathbf{h})^\top \mathbf{P}_1$$

$$\boldsymbol{\rho}_i = -\frac{1}{2} \kappa_i \left[\mathbf{W}_i \mathbf{W}_i^\top + \sum_{j=1}^{i-1} \boldsymbol{\varpi}_{i-1,j} \boldsymbol{\varpi}_{i-1,j}^\top \right] \mathbf{P}_i, \qquad (i = 2, \ldots, n)$$

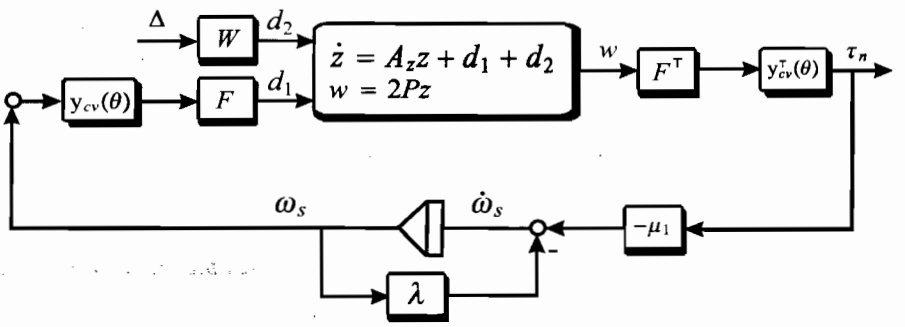


Figure 10.3: Closed-loop maneuvering system.

correspond to the nonlinear damping terms. Furthermore, let:

$$\mathbf{F}(\bar{\mathbf{x}}_n, \theta, t) = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{y}_d} & \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{y}_d'} & \mathbf{0} & \cdots & \mathbf{0} \\ \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{y}_d} & \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{y}_d'} & \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{y}_d''} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{\alpha}_{n-1}}{\partial \mathbf{y}_d} & \frac{\partial \boldsymbol{\alpha}_{n-1}}{\partial \mathbf{y}_d'} & \frac{\partial \boldsymbol{\alpha}_{n-1}}{\partial \mathbf{y}_d''} & \cdots & \frac{\partial \boldsymbol{\alpha}_{n-1}}{\partial \mathbf{y}_d^{(n)}} \end{bmatrix} \tag{10.68}$$

$$\mathbf{W}(\bar{\mathbf{x}}_n, \theta, t) = \begin{bmatrix} \nabla \mathbf{h} \mathbf{W}_1 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -\boldsymbol{\varpi}_{11} & \mathbf{W}_2 & \mathbf{0} & \cdots & \mathbf{0} \\ -\boldsymbol{\varpi}_{21} & -\boldsymbol{\varpi}_{22} & \mathbf{W}_3 & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ -\boldsymbol{\varpi}_{n-1,1} & -\boldsymbol{\varpi}_{n-1,2} & -\boldsymbol{\varpi}_{n-1,3} & \cdots & \mathbf{W}_n \end{bmatrix} \tag{10.69}$$

This gives:

$$\tau_n = 2\mathbf{y}_{cv}^\top \mathbf{F}^\top \mathbf{P} \mathbf{z} \tag{10.70}$$

where $\mathbf{P} = \text{diag}\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_n\}$. Thus, the closed-loop is:

$$\dot{\mathbf{z}} = \mathbf{A}_z \mathbf{z} + \mathbf{F} \mathbf{y}_{cv} \omega_s + \mathbf{W} \boldsymbol{\Delta} \tag{10.71}$$

$$\dot{\omega}_s = -2\mu_1 \mathbf{y}_{cv}^\top \mathbf{F}^\top \mathbf{P} \mathbf{z} - \lambda \omega_s \tag{10.72}$$

which has an interconnection structure as shown in Figure 10.3. This clearly implies the interconnection of two strictly passive systems.

Theorem 10.1 (Robust Maneuvering)
The closed-loop output maneuvering system consisting of the plant (10.34), the dynamic controller with dynamics (10.65) and (10.67), and control law (10.60) is ISS from the disturbance

Δ to the error states $(\mathbf{z}, \omega_s)$ where the gain $\Delta \to (\mathbf{z}, \omega_s)$ can be assigned arbitrarily small. Given $\varepsilon_{GT} > 0$ and $\varepsilon_{DT} > 0$, there exist κ^* such that for $\kappa_i > \kappa^*$, $i = 1, \ldots, n$ the maneuvering problem with respect to the output path (10.35) is solved:

$$\begin{aligned} [\mathbf{y}(t) - \mathbf{y}_d(\theta(t))] &\leq \varepsilon_{GT}, \quad \forall t \geq T \\ \left[\dot{\theta}(t) - v_s(\theta(t), t)\right] &\leq \varepsilon_{DT}, \quad \forall t \geq T \end{aligned}$$

for some $T \geq 0$. In addition, if $\varepsilon_{DT} < |v_s(\theta(t), t)|$, $\forall t \geq 0$, then the output $\mathbf{y}(t)$ is ensured to move along the path with desired speed.

In the disturbance-free case, $\Delta \equiv \mathbf{0}$, the nonlinear damping gains $\kappa_1, \kappa_2, \ldots, \kappa_n$ can be set to zero, and the closed-loop system (10.71) and (10.72) has a UGES equilibrium $(\mathbf{z}, \omega_s) = (\mathbf{0}, 0)$.

Proof. See Skjetne *et al.* (2002*d*). ■

10.2.2 Adaptive Output Maneuvering

In this section, adaptive backstepping is applied to solve the maneuvering problem for parametrically uncertain nonlinear systems in strict feedback form. Consider the plant with vector relative degree n:

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{G}_1(\bar{\mathbf{x}}_1)\,\mathbf{x}_2 + \mathbf{f}_1(\bar{\mathbf{x}}_1) + \boldsymbol{\varphi}_1(\bar{\mathbf{x}}_1)\,\boldsymbol{\Theta} \\ \dot{\mathbf{x}}_2 &= \mathbf{G}_2(\bar{\mathbf{x}}_2)\,\mathbf{x}_3 + \mathbf{f}_2(\bar{\mathbf{x}}_2) + \boldsymbol{\varphi}_2(\bar{\mathbf{x}}_2)\,\boldsymbol{\Theta} \\ &\vdots \\ \dot{\mathbf{x}}_n &= \mathbf{G}_n(\bar{\mathbf{x}}_n)\,\mathbf{u} + \mathbf{f}_n(\bar{\mathbf{x}}_n) + \boldsymbol{\varphi}_n(\bar{\mathbf{x}}_n)\,\boldsymbol{\Theta} \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}_1) \end{aligned} \tag{10.73}$$

where $\mathbf{x}_i \in \mathbb{R}^m$ $(i = 1, \ldots, n)$ are the states, $\mathbf{y} \in \mathbb{R}^m$ is the output, $\mathbf{u} \in \mathbb{R}^m$ is the control, $\boldsymbol{\varphi}_i$ are the regressor matrices, and $\boldsymbol{\Theta}$ is the vector of constant unknown parameters. $\bar{\mathbf{x}}_i$ denotes the vector $\bar{\mathbf{x}}_i = \left[\mathbf{x}_1^\top, \mathbf{x}_2^\top, \ldots, \mathbf{x}_i^\top\right]^\top$. The matrices $\mathbf{G}_i(\bar{\mathbf{x}}_i)$ and $\nabla\mathbf{h}(\mathbf{x}_1) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_1}(\mathbf{x}_1)$ are invertible for all $\bar{\mathbf{x}}_i$, $\mathbf{h}$ is a diffeomorphism, and $\mathbf{G}_i$, $\mathbf{f}_i$, $\boldsymbol{\varphi}_i$ are locally Lipschitz.

The control objective is to design a maneuvering controller that solves the output maneuvering problem, the geometric task (10.30) and the speed assignment task (10.32), with respect to the desired parametrized output path

$$\mathbf{Y}_d = \{\mathbf{y} \in \mathbb{R}^m : \mathbf{y} = \mathbf{y}_d(\theta),\ \theta \in J_\theta \subseteq \mathbb{R}\} \tag{10.74}$$

where $\mathbf{y}_d$ is n times differentiable w.r.t. θ and the path characterization vector $\mathbf{y}_{cv}(\theta)$ is uniformly bounded in θ.

Recursive Design Procedure

The first steps is presented in detail while the 2nd and *i-th* Steps are given in Tables 10.2 and 10.3, respectively.

Step 1:

Define the error variables

$$\omega_s = v_s - \dot{\theta} \tag{10.75}$$

$$\mathbf{z}_1 = \mathbf{y} - \mathbf{y}_d(\theta) \tag{10.76}$$

$$\mathbf{z}_i = \mathbf{x}_i - \boldsymbol{\alpha}_{i-1}, \qquad (i = 2, \ldots, n) \tag{10.77}$$

$$\tilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta} - \hat{\boldsymbol{\Theta}} \tag{10.78}$$

where $\hat{\boldsymbol{\Theta}}$ is the parameter estimate, v_s is a bounded C^{n-1} signal, and $\boldsymbol{\alpha}_{i-1}$ are virtual controls to be specified later. Differentiating (10.76) with respect to time results in:

$$\begin{aligned}\dot{\mathbf{z}}_1 &= \dot{\mathbf{y}} - \mathbf{y}_d'\dot{\theta} \\ &= \nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + \nabla\mathbf{h}\mathbf{G}_1\boldsymbol{\alpha}_1 + \nabla\mathbf{h}\mathbf{f}_1 + \nabla\mathbf{h}\boldsymbol{\varphi}_1\tilde{\boldsymbol{\Theta}} + \nabla\mathbf{h}\boldsymbol{\varphi}_1\hat{\boldsymbol{\Theta}} - \boldsymbol{\nu}_1 v_s + \boldsymbol{\nu}_1\omega_s\end{aligned} \tag{10.79}$$

where $\boldsymbol{\nu}_1 = \mathbf{y}_d'(\theta)$. Choose $\mathbf{A}_1$ Hurwitz and let $\mathbf{P}_1 = \mathbf{P}_1^\top > 0$ be the solution to $\mathbf{P}_1\mathbf{A}_1 + \mathbf{A}_1^\top\mathbf{P}_1 = -\mathbf{Q}_1 < 0$. Define the first Lyapunov function as:

$$V_1 = \mathbf{z}_1^\top\mathbf{P}_1\mathbf{z}_1 + \frac{1}{2}\tilde{\boldsymbol{\Theta}}^\top\boldsymbol{\Gamma}^{-1}\tilde{\boldsymbol{\Theta}} \tag{10.80}$$

whose time derivative is:

$$\begin{aligned}\dot{V}_1 = {} & 2\mathbf{z}_1^\top\mathbf{P}_1\left[\nabla\mathbf{h}\mathbf{G}_1\boldsymbol{\alpha}_1 + \nabla\mathbf{h}\mathbf{f}_1 + \nabla\mathbf{h}\boldsymbol{\varphi}_1\hat{\boldsymbol{\Theta}} - \boldsymbol{\nu}_1 v_s\right] \\ & + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\boldsymbol{\varphi}_1\tilde{\boldsymbol{\Theta}} + 2\mathbf{z}_1^\top\mathbf{P}_1\boldsymbol{\nu}_1\omega_s - \tilde{\boldsymbol{\Theta}}^\top\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\end{aligned}$$

where $\dot{\tilde{\boldsymbol{\Theta}}} = -\dot{\hat{\boldsymbol{\Theta}}}$. The first virtual control law is picked as:

$$\begin{aligned}\boldsymbol{\alpha}_1 &= \mathbf{G}_1^{-1}(\nabla\mathbf{h})^{-1}\left[\mathbf{A}_1\mathbf{z}_1 - \nabla\mathbf{h}\mathbf{f}_1 - \nabla\mathbf{h}\boldsymbol{\varphi}_1\hat{\boldsymbol{\Theta}} + \boldsymbol{\nu}_1 v_s\right] \\ &= \mathbf{f}_{\alpha_1}(\mathbf{x}_1, \hat{\boldsymbol{\Theta}}, \theta, t)\end{aligned} \tag{10.81}$$

Let $\boldsymbol{\Delta}_1 = \nabla\mathbf{h}\boldsymbol{\varphi}_1$, and define the first tuning function, $\tau_1 \in \mathbb{R}$, and the adaptive tuning function, $\boldsymbol{\rho}_1 \in \mathbb{R}^p$, as:

$$\tau_1 = 2\mathbf{z}_1^\top\mathbf{P}_1\boldsymbol{\nu}_1 \tag{10.82}$$

$$\boldsymbol{\rho}_1 = 2\boldsymbol{\Delta}_1^\top\mathbf{P}_1\mathbf{z}_1 \tag{10.83}$$

Hence, the result of Step 1 is:

$$\dot{V}_1 = -\mathbf{z}_1^\top\mathbf{Q}_1\mathbf{z}_1 + \tau_1\omega_s + \tilde{\boldsymbol{\Theta}}^\top\left[\boldsymbol{\rho}_1 - \boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\right] + 2\mathbf{z}_1^\top\mathbf{P}_1\nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 \tag{10.84}$$

$$\dot{\mathbf{z}}_1 = \mathbf{A}_1\mathbf{z}_1 + \nabla\mathbf{h}\mathbf{G}_1\mathbf{z}_2 + \boldsymbol{\nu}_1\omega_s + \boldsymbol{\Delta}_1\tilde{\boldsymbol{\Theta}} \tag{10.85}$$

where the terms containing $\mathbf{z}_2$, ω_s, and $\tilde{\boldsymbol{\Theta}}$ are left for the next step. To aid the next step, let:

$$\dot{\boldsymbol{\alpha}}_1 = \boldsymbol{\sigma}_1 + \boldsymbol{\nu}_2\dot{\theta} + \boldsymbol{\chi}_{1,1}\tilde{\boldsymbol{\Theta}} + \boldsymbol{\chi}_{1,2}\dot{\hat{\boldsymbol{\Theta}}} \tag{10.86}$$

Table 10.2: Adaptive Maneuvering: Step 2.

$$\dot{\mathbf{z}}_2=\dot{\mathbf{x}}_2-\dot{\boldsymbol{\alpha}}_1=\mathbf{G}_2\left(\bar{\mathbf{x}}_2\right)\mathbf{z}_3+\mathbf{G}_2\left(\bar{\mathbf{x}}_2\right)\boldsymbol{\alpha}_2+\mathbf{f}_2\left(\bar{\mathbf{x}}_2\right)$$
$$+\boldsymbol{\varphi}_2\left(\bar{\mathbf{x}}_2\right)\tilde{\boldsymbol{\Theta}}+\boldsymbol{\varphi}_2\left(\bar{\mathbf{x}}_2\right)\hat{\boldsymbol{\Theta}}-\boldsymbol{\sigma}_1-\boldsymbol{\nu}_2 v_s+\boldsymbol{\nu}_2\omega_s-\boldsymbol{\chi}_{1,1}\tilde{\boldsymbol{\Theta}}-\boldsymbol{\chi}_{1,2}\dot{\hat{\boldsymbol{\Theta}}}$$

$$V_2=V_1+\mathbf{z}_2^{\top}\mathbf{P}_2\mathbf{z}_2$$
$$\dot{V}_2=-\mathbf{z}_1^{\top}\mathbf{Q}_1\mathbf{z}_1+\tilde{\boldsymbol{\Theta}}^{\top}\left[\boldsymbol{\rho}_1-\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\right]$$
$$+2\mathbf{z}_2^{\top}\left\{\mathbf{G}_1^{\top}\left(\nabla\mathbf{h}\right)^{\top}\mathbf{P}_1\mathbf{z}_1+\mathbf{P}_2\left[\mathbf{G}_2\boldsymbol{\alpha}_2+\mathbf{f}_2+\boldsymbol{\varphi}_2\hat{\boldsymbol{\Theta}}-\boldsymbol{\sigma}_1-\boldsymbol{\nu}_2 v_s-\boldsymbol{\chi}_{1,2}\dot{\hat{\boldsymbol{\Theta}}}\right]\right\}$$
$$+2\mathbf{z}_2^{\top}\mathbf{P}_2\left[\boldsymbol{\varphi}_2-\boldsymbol{\chi}_{1,1}\right]\tilde{\boldsymbol{\Theta}}+2\mathbf{z}_2^{\top}\mathbf{P}_2\boldsymbol{\nu}_2\omega_s+\tau_1\omega_s+2\mathbf{z}_2^{\top}\mathbf{P}_2\mathbf{G}_2\mathbf{z}_3$$

$$\boldsymbol{\Delta}_2=\boldsymbol{\varphi}_2-\boldsymbol{\chi}_{1,1}$$
$$\tau_2=\tau_1+2\mathbf{z}_2^{\top}\mathbf{P}_2\boldsymbol{\nu}_2 \quad \text{2nd Tuning Function}$$
$$\boldsymbol{\rho}_2=\boldsymbol{\rho}_1+2\boldsymbol{\Delta}_2^{\top}\mathbf{P}_2\mathbf{z}_2 \quad \text{2nd Adaptive Tuning Function}$$

$$\boldsymbol{\alpha}_2=\mathbf{G}_2^{-1}\left[\mathbf{A}_2\mathbf{z}_2-\mathbf{P}_2^{-1}\mathbf{G}_1^{\top}\left(\nabla h\right)\mathbf{P}_1\mathbf{z}_1-\mathbf{f}_2-\boldsymbol{\varphi}_2\hat{\boldsymbol{\Theta}}+\boldsymbol{\sigma}_1+\boldsymbol{\nu}_2 v_s+\boldsymbol{\chi}_{12}\boldsymbol{\Gamma}\boldsymbol{\rho}_2\right]$$
$$=\mathbf{f}_{\alpha_2}\left(\bar{\mathbf{x}}_2,\hat{\boldsymbol{\Theta}},\theta,t\right)$$
$$\mathbf{P}_2\mathbf{A}_2+\mathbf{A}_2^{\top}\mathbf{P}_2=-\mathbf{Q}_2<0$$
$$\boldsymbol{\varpi}_2^{\top}=2\mathbf{z}_2^{\top}\mathbf{P}_2\boldsymbol{\chi}_{1,2}$$

$$\Downarrow$$

$$\dot{V}_2=-\mathbf{z}_1^{\top}\mathbf{Q}_1\mathbf{z}_1-\mathbf{z}_2^{\top}\mathbf{Q}_2\mathbf{z}_2+\tau_2\omega_s+\tilde{\boldsymbol{\Theta}}^{\top}\left[\boldsymbol{\rho}_2-\boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\right]+\boldsymbol{\varpi}_2^{\top}\left[\boldsymbol{\Gamma}\boldsymbol{\rho}_2-\dot{\hat{\boldsymbol{\Theta}}}\right]+2\mathbf{z}_2^{\top}\mathbf{P}_2\mathbf{G}_2\mathbf{z}_3$$
$$\dot{\mathbf{z}}_2=-\mathbf{P}_2^{-1}\mathbf{G}_1^{\top}\left(\nabla\mathbf{h}\right)^{\top}\mathbf{P}_1\mathbf{z}_1+\mathbf{A}_2\mathbf{z}_2+\mathbf{G}_2\mathbf{z}_3+\boldsymbol{\nu}_2\omega_s+\boldsymbol{\Delta}_2\tilde{\boldsymbol{\Theta}}+\boldsymbol{\chi}_{1,2}\left[\boldsymbol{\Gamma}\boldsymbol{\rho}_2-\dot{\hat{\boldsymbol{\Theta}}}\right]$$

$$\dot{\boldsymbol{\alpha}}_2=\boldsymbol{\sigma}_2+\boldsymbol{\nu}_3\dot{\theta}+\boldsymbol{\chi}_{2,1}\tilde{\boldsymbol{\Theta}}+\boldsymbol{\chi}_{2,2}\dot{\hat{\boldsymbol{\Theta}}}$$
$$\boldsymbol{\sigma}_2=\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_1}\left[\mathbf{G}_1\mathbf{x}_2+\mathbf{f}_1+\boldsymbol{\varphi}_1\hat{\boldsymbol{\Theta}}\right]+\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_2}\left[\mathbf{G}_2\mathbf{x}_3+\mathbf{f}_2+\boldsymbol{\varphi}_2\hat{\boldsymbol{\Theta}}\right]+\tfrac{\partial\boldsymbol{\alpha}_2}{\partial t}$$
$$\boldsymbol{\nu}_3=\boldsymbol{\alpha}_2'=\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\theta}$$
$$\boldsymbol{\chi}_{2,1}=\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_1}\boldsymbol{\varphi}_1+\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_2}\boldsymbol{\varphi}_2$$
$$\boldsymbol{\chi}_{2,2}=\tfrac{\partial\boldsymbol{\alpha}_2}{\partial\hat{\boldsymbol{\Theta}}}$$

where

$$\boldsymbol{\sigma}_1=\frac{\partial\boldsymbol{\alpha}_1}{\partial\mathbf{x}_1}\left[\mathbf{G}_1\mathbf{x}_2+\mathbf{f}_1+\boldsymbol{\varphi}_1\hat{\boldsymbol{\Theta}}\right]+\frac{\partial\boldsymbol{\alpha}_1}{\partial t} \tag{10.87}$$

$$\boldsymbol{\nu}_2=\boldsymbol{\alpha}_1'=\frac{\partial\boldsymbol{\alpha}_1}{\partial\theta} \tag{10.88}$$

$$\boldsymbol{\chi}_{1,1}=\frac{\partial\boldsymbol{\alpha}_1}{\partial\mathbf{x}_1}\boldsymbol{\varphi}_1 \tag{10.89}$$

$$\boldsymbol{\chi}_{1,2}=\frac{\partial\boldsymbol{\alpha}_1}{\partial\hat{\boldsymbol{\Theta}}} \tag{10.90}$$

Table 10.3: Adaptive Maneuvering: Steps i=3,...,n.

$$\dot{\mathbf{z}}_i = \dot{\mathbf{x}}_i - \dot{\boldsymbol{\alpha}}_{i-1} = \mathbf{G}_i(\bar{\mathbf{x}}_i)\mathbf{z}_{i+1} + \mathbf{G}_i(\bar{\mathbf{x}}_i)\boldsymbol{\alpha}_i + \mathbf{f}_i(\bar{\mathbf{x}}_i) + \boldsymbol{\varphi}_i(\bar{\mathbf{x}}_i)\tilde{\boldsymbol{\Theta}} + \boldsymbol{\varphi}_i(\bar{\mathbf{x}}_i)\hat{\boldsymbol{\Theta}} - \boldsymbol{\sigma}_{i-1} - \boldsymbol{\nu}_i v_s + \boldsymbol{\nu}_i \omega_s - \boldsymbol{\chi}_{i-1,1}\tilde{\boldsymbol{\Theta}} - \boldsymbol{\chi}_{i-1,2}\dot{\hat{\boldsymbol{\Theta}}}$$

$$V_i = V_{i-1} + \mathbf{z}_i^\top \mathbf{P}_i \mathbf{z}_i$$

$$\dot{V}_i = -\sum_{j=1}^{i-1} \mathbf{z}_j^\top \mathbf{Q}_j \mathbf{z}_j + \tilde{\boldsymbol{\Theta}}^\top \left[\boldsymbol{\rho}_{i-1} - \boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\right] + \boldsymbol{\varpi}_{i-1}^\top \left[\boldsymbol{\Gamma}\boldsymbol{\rho}_{i-1} - \dot{\hat{\boldsymbol{\Theta}}}\right] + 2\mathbf{z}_i^\top \left\{\mathbf{G}_{i-1}^\top \mathbf{P}_{i-1}\mathbf{z}_{i-1} + \mathbf{P}_i\left[\mathbf{G}_i\boldsymbol{\alpha}_i + \mathbf{f}_i + \boldsymbol{\varphi}_i\hat{\boldsymbol{\Theta}} - \boldsymbol{\sigma}_{i-1} - \boldsymbol{\nu}_i v_s - \boldsymbol{\chi}_{i-1,2}\dot{\hat{\boldsymbol{\Theta}}}\right]\right\} + 2\mathbf{z}_i^\top \mathbf{P}_i\left(\boldsymbol{\varphi}_i - \boldsymbol{\chi}_{i-1,1}\right)\tilde{\boldsymbol{\Theta}} + \tau_{i-1}\omega_s + 2\mathbf{z}_i^\top \mathbf{P}_i \boldsymbol{\nu}_i \omega_s + 2\mathbf{z}_i^\top \mathbf{P}_i \mathbf{G}_i \mathbf{z}_{i+1}$$

$$\boldsymbol{\Delta}_i = \boldsymbol{\varphi}_i - \boldsymbol{\chi}_{i-1,1}$$
$$\tau_i = \tau_{i-1} + 2\mathbf{z}_i^\top \mathbf{P}_i \boldsymbol{\nu}_i \quad \text{i-th Tuning Function}$$
$$\boldsymbol{\rho}_i = \boldsymbol{\rho}_{i-1} + 2\boldsymbol{\Delta}_i^\top \mathbf{P}_i \mathbf{z}_i \quad \text{i-th Adaptive Tuning Function}$$

$$\boldsymbol{\alpha}_i = \mathbf{G}_i^{-1}[\mathbf{A}_i\mathbf{z}_i - \mathbf{P}_i^{-1}\mathbf{G}_{i-1}^\top \mathbf{P}_{i-1}\mathbf{z}_{i-1} - \mathbf{f}_i - \boldsymbol{\varphi}_i\hat{\boldsymbol{\Theta}} + \boldsymbol{\sigma}_{i-1} + \boldsymbol{\nu}_i v_s + \boldsymbol{\chi}_{i-1,2}\boldsymbol{\Gamma}\boldsymbol{\rho}_i + \boldsymbol{\Delta}_i\boldsymbol{\Gamma}\boldsymbol{\varpi}_{i-1}] = \mathbf{f}_{\alpha_i}\left(\bar{\mathbf{x}}_i, \hat{\boldsymbol{\Theta}}, \theta, t\right)$$

$$\mathbf{P}_i\mathbf{A}_i + \mathbf{A}_i^\top \mathbf{P}_i = -\mathbf{Q}_i < 0$$
$$\boldsymbol{\varpi}_i^\top = \boldsymbol{\varpi}_{i-1}^\top + 2\mathbf{z}_i^\top \mathbf{P}_i \boldsymbol{\chi}_{i-1,2}$$

$$\Downarrow$$

$$\dot{V}_i = -\sum_{j=1}^{i} \mathbf{z}_j^\top \mathbf{Q}_j \mathbf{z}_j + \tau_i\omega_s + \tilde{\boldsymbol{\Theta}}^\top \left[\boldsymbol{\rho}_i - \boldsymbol{\Gamma}^{-1}\dot{\hat{\boldsymbol{\Theta}}}\right] + \boldsymbol{\varpi}_i^\top \left[\boldsymbol{\Gamma}\boldsymbol{\rho}_i - \dot{\hat{\boldsymbol{\Theta}}}\right] + 2\mathbf{z}_i^\top \mathbf{P}_i \mathbf{G}_i \mathbf{z}_{i+1}$$
$$\dot{\mathbf{z}}_i = -\mathbf{P}_i^{-1}\mathbf{G}_{i-1}^\top \mathbf{P}_{i-1}\mathbf{z}_{i-1} + \mathbf{A}_i\mathbf{z}_i + \mathbf{G}_i\mathbf{z}_{i+1} + \boldsymbol{\nu}_i\omega_s + \boldsymbol{\Delta}_i\tilde{\boldsymbol{\Theta}} + \boldsymbol{\chi}_{i-1,2}\left[\boldsymbol{\Gamma}\boldsymbol{\rho}_i - \dot{\hat{\boldsymbol{\Theta}}}\right]$$

$$\dot{\boldsymbol{\alpha}}_i = \boldsymbol{\sigma}_i + \boldsymbol{\nu}_{i+1}\dot{\theta} + \boldsymbol{\chi}_{i,1}\tilde{\boldsymbol{\Theta}} + \boldsymbol{\chi}_{i,2}\dot{\hat{\boldsymbol{\Theta}}}$$
$$\boldsymbol{\sigma}_i = \sum_{j=1}^{i} \frac{\partial \boldsymbol{\alpha}_i}{\partial \mathbf{x}_j}\left[\mathbf{G}_j\mathbf{x}_{j+1} + \mathbf{f}_j + \boldsymbol{\varphi}_j\hat{\boldsymbol{\Theta}}\right] + \frac{\partial \boldsymbol{\alpha}_i}{\partial t}$$
$$\boldsymbol{\nu}_{i+1} = \boldsymbol{\alpha}_i' = \frac{\partial \boldsymbol{\alpha}_i}{\partial \theta}$$
$$\boldsymbol{\chi}_{i,1} = \sum_{j=1}^{i} \frac{\partial \boldsymbol{\alpha}_i}{\partial \mathbf{x}_j}\boldsymbol{\varphi}_j$$
$$\boldsymbol{\chi}_{i,2} = \frac{\partial \boldsymbol{\alpha}_i}{\partial \hat{\boldsymbol{\Theta}}}$$

Step i:

Step 2 and Steps $i = 3, \ldots, n$ are summarized in Tables 10.2 and 10.3. Notice the introduction of the terms $\boldsymbol{\varpi}_i$ $(i = 2, \ldots, n)$ which from adaptive backstepping theory is known to occur for plants of relative degree of 3 or higher (Krstic *et al.* 1995).

Step n:

Upon the completion of Step n the control law and the update laws are designed as:

$$\begin{aligned}\mathbf{u} &= \boldsymbol{\alpha}_n\left(\bar{\mathbf{x}}_n, \hat{\boldsymbol{\Theta}}, \theta, t\right) \\ &= \mathbf{G}_n^{-1}[\mathbf{A}_n\mathbf{z}_n - \mathbf{P}_n^{-1}\mathbf{G}_{n-1}^{\top}\mathbf{P}_{n-1}\mathbf{z}_{n-1} - \mathbf{f}_n - \boldsymbol{\varphi}_n\hat{\boldsymbol{\Theta}} + \boldsymbol{\sigma}_{n-1} + \boldsymbol{\nu}_n\upsilon_s \\ &\quad + \boldsymbol{\chi}_{n-1,2}\boldsymbol{\Gamma}\boldsymbol{\rho}_n + \boldsymbol{\Delta}_n\boldsymbol{\Gamma}\boldsymbol{\varpi}_{n-1}] \qquad (10.91)\end{aligned}$$

$$\dot{\hat{\boldsymbol{\Theta}}} = \boldsymbol{\Gamma}\boldsymbol{\rho}_n \qquad (10.92)$$

$$\dot{\omega}_s = -\lambda\omega_s - \mu_1\tau_n \qquad (10.93)$$

The augmented Lyapunov function:

$$V = V_n + \frac{1}{2\mu_1}\omega_s^2 = \mathbf{z}^{\top}\mathbf{P}\mathbf{z} + \frac{1}{2}\tilde{\boldsymbol{\Theta}}^{\top}\boldsymbol{\Gamma}^{-1}\tilde{\boldsymbol{\Theta}} + \frac{1}{2\mu_1}\omega_s^2 \qquad (10.94)$$

where $\mathbf{P} = \mathrm{diag}\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_n\}$, results in a negative semi-definite time derivative:

$$\dot{V} = -\mathbf{z}^{\top}\mathbf{Q}\mathbf{z} - \frac{\lambda}{\mu_1}\omega_s^2 \leq 0 \qquad (10.95)$$

It follows by application of the LaSalle-Yoshizawa Theorem that $(\mathbf{z}(t), \omega_s(t)) \rightarrow (\mathbf{0}, 0)$ as $t \rightarrow \infty$, which solves the geometric task (10.30) and the speed assignment task (10.32).

Error Dynamics

To rewrite the closed-loop system in vector form, let $\mathbf{A}_z \in \mathbb{R}^{nm\times nm}$, $\mathbf{F} \in \mathbb{R}^{nm\times nm}$ and $\mathbf{H} \in \mathbb{R}^{nm\times nm}$ be defined by:

$$\mathbf{A}_z\left(\bar{\mathbf{x}}_n, \hat{\boldsymbol{\Theta}}, \theta, t\right) =$$

$$\begin{bmatrix} \mathbf{A}_1 & \nabla\mathbf{h}\mathbf{G}_1 & \mathbf{0} & 0 & \cdots & \mathbf{0} \\ -\mathbf{P}_2^{-1}\mathbf{G}_1^{\top}(\nabla\mathbf{h})^{\top}\mathbf{P}_1 & \mathbf{A}_2 & \mathbf{G}_2+\boldsymbol{\kappa}_{2,3} & \boldsymbol{\kappa}_{2,4} & \cdots & \boldsymbol{\kappa}_{2,n} \\ \mathbf{0} & -\mathbf{P}_3^{-1}\mathbf{G}_2^{\top}\mathbf{P}_2 & \mathbf{A}_3 & \mathbf{G}_3+\boldsymbol{\kappa}_{3,4} & \cdots & \boldsymbol{\kappa}_{3,n} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & -\mathbf{P}_n^{-1}\mathbf{G}_{n-1}^{\top}\mathbf{P}_{n-1} & \mathbf{A}_n \end{bmatrix}$$

where $\boldsymbol{\kappa}_{i,j} = -2\boldsymbol{\chi}_{i-1,2}\boldsymbol{\Gamma}\boldsymbol{\Delta}_j^{\top}\mathbf{P}_j$ $(i = 2, \ldots, n-1,\ j = i+1, \ldots, n)$ and:

$$\mathbf{F}(\bar{\mathbf{x}}_n, \hat{\boldsymbol{\Theta}}, \theta, t) = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \frac{\partial\boldsymbol{\alpha}_1}{\partial\mathbf{y}_d} & \frac{\partial\boldsymbol{\alpha}_1}{\partial\mathbf{y}_d'} & \mathbf{0} & \cdots & \mathbf{0} \\ \frac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{y}_d} & \frac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{y}_d'} & \frac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{y}_d''} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{y}_d} & \frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{y}_d'} & \frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{y}_d''} & \cdots & \frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{y}_d^{(n)}} \end{bmatrix} \qquad (10.96)$$

$$\mathbf{H}\left(\bar{\mathbf{x}}_{n-1}, \hat{\boldsymbol{\Theta}}, \theta, t\right) = \begin{bmatrix} \nabla\mathbf{h} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -\frac{\partial\boldsymbol{\alpha}_1}{\partial\mathbf{x}_1} & \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ -\frac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_1} & -\frac{\partial\boldsymbol{\alpha}_2}{\partial\mathbf{x}_2} & \mathbf{I} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \mathbf{0} \\ -\frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{x}_1} & -\frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{x}_2} & \cdots & -\frac{\partial\boldsymbol{\alpha}_{n-1}}{\partial\mathbf{x}_{n-1}} & \mathbf{I} \end{bmatrix} \qquad (10.97)$$

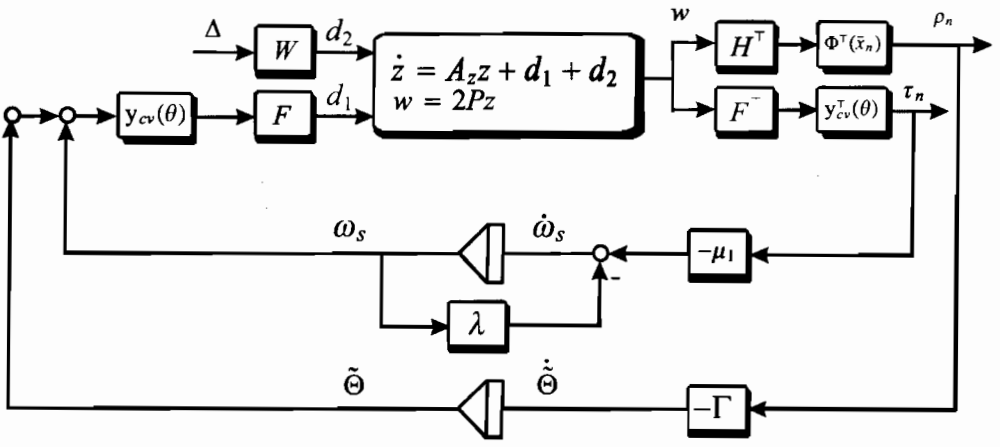


Figure 10.4: Closed-loop adaptive maneuvering system.

Let the overall regressor matrix $\bar{\mathbf{\Phi}} \in \mathbb{R}^{nm \times p}$ be defined as:

$$\mathbf{\Phi}(\bar{\mathbf{x}}_n) = \left[\boldsymbol{\varphi}_1^\top(\bar{\mathbf{x}}_1), \boldsymbol{\varphi}_2^\top(\bar{\mathbf{x}}_2), \boldsymbol{\varphi}_3^\top(\bar{\mathbf{x}}_3), \ldots, \boldsymbol{\varphi}_n^\top(\bar{\mathbf{x}}_n)\right]^\top \tag{10.98}$$

and $\mathbf{y}_{cv}(\theta)$ is given by (10.36). Then the tuning function τ_n and adaptive tuning function $\boldsymbol{\rho}_n$ are written:

$$\tau_n = 2\mathbf{y}_{cv}^\top \mathbf{F}^\top \mathbf{P}\mathbf{z} \tag{10.99}$$

$$\boldsymbol{\rho}_n = 2\mathbf{\Phi}^\top \mathbf{H}^\top \mathbf{P}\mathbf{z} \tag{10.100}$$

and the closed-loop equations are given by:

$$\dot{\mathbf{z}} = \mathbf{A}_z \mathbf{z} + \mathbf{F}\mathbf{y}_{cv}^\top \omega_s + \mathbf{H}\mathbf{\Phi}\tilde{\mathbf{\Theta}} \tag{10.101}$$

$$\dot{\omega}_s = -2\mu_1 \mathbf{y}_{cv}^\top \mathbf{F}^\top \mathbf{P}\mathbf{z} - \lambda\omega_s \tag{10.102}$$

$$\dot{\tilde{\mathbf{\Theta}}} = -2\mathbf{\Gamma}\mathbf{\Phi}^\top \mathbf{H}^\top \mathbf{P}\mathbf{z} \tag{10.103}$$

The interconnection structure is shown in Figure 10.4.

Theorem 10.2 (Adaptive Maneuvering)
The closed-loop output maneuvering system, consisting of the parametrically uncertain plant (10.73), the dynamic controller consisting of the update laws(10.67) and (10.93), the adaptive update law (10.92), and the control law (10.91), has a UGS equilibrium $\left(\mathbf{z}, \tilde{\mathbf{\Theta}}, \omega_s\right) = \mathbf{0}$ *where* $(\mathbf{z}(t), \omega_s(t)) \to 0$ *as* $t \to \infty$. *This means that the maneuvering problem, with respect to the path (10.35), is solved with limits:*

$$\lim_{t\to\infty} \left[\mathbf{y}(t) - \mathbf{y}_d(\theta(t))\right] = \mathbf{0}$$

$$\lim_{t\to\infty} \left[\dot{\theta}(t) - v_s(\theta(t), t)\right] = 0$$

Proof. See Skjetne *et al.* (2002*d*). ■

Example 10.1 (Fully Actuated Container Ship)
Consider a high-speed container ship in 3 DOF with data taken from Appendix E.1.3 of Fossen (1994). The ship is of length $L = 175$ (m)*, displacement volume of* 21.222 (m^3)*, and it is actuated by two rudders and a forward thrust propeller. Let* $\boldsymbol{\eta} = [n, e, \psi]^\top$ *denote the North-East positions and yaw angle, and* $\boldsymbol{\nu} = [u, v, r]^\top$ *be the velocity vector. The kinematic equation is:*

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{10.104}$$

where $\mathbf{R}(\psi)$ *is the rotation matrix in yaw. Due to port-starboard symmetry of the ship, the surge speed equation and the steering equations (sway and yaw) are assumed to be decoupled. The speed equation is given by:*

$$(m - X_{\dot{u}})\,\dot{u} - X_u u - X_{|u|u}\,|u|\,u = (1 - t_d)\,T$$

where $X_{(\cdot)} < 0$ *are the hydrodynamic coefficients,* $0 < t_d < 1$ *is the thrust deduction number, and* T *is the propeller thrust. The steering equations are given by*

$$\begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} = \frac{U}{L}\begin{bmatrix} \bar{a}_{11} & L\bar{a}_{12} \\ \frac{1}{L}\bar{a}_{21} & \bar{a}_{22} \end{bmatrix}\begin{bmatrix} v \\ r \end{bmatrix} + \frac{U^2}{L}\begin{bmatrix} \bar{b}_{11} & \bar{b}_{12} \\ \frac{1}{L}\bar{b}_{21} & \frac{1}{L}\bar{b}_{22} \end{bmatrix}\begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}$$

where δ_1, δ_2 *are the rudder angles, the hydrodynamic coefficients* $\bar{a}_{ij}, \bar{b}_{ij}$ *are nondimensional, and* $U = \sqrt{u^2 + v^2}$ *is the total speed. The environmental forces (except 1st-order wave-induced disturbances which need to be filtered out) are assumed to be constant (or at least slowly varying), and represented by an additive constant* w *denoting the environmental force. This gives the dynamic equation:*

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}(\boldsymbol{\nu})\boldsymbol{\nu} = \mathbf{B}(\boldsymbol{\nu})\,\mathbf{u} + \mathbf{R}^\top(\psi)\mathbf{w} \tag{10.105}$$

where $\mathbf{u} = [T, \delta_1, \delta_2]^\top$ *is the control vector and:*

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & \mathbf{0}_{1\times 2} \\ \mathbf{0}_{2\times 1} & \mathbf{I}_{2\times 2} \end{bmatrix}$$

$$\mathbf{N}(\boldsymbol{\nu}) = \begin{bmatrix} -X_u - X_{|u|u}\,|u| & \mathbf{0}_{1\times 2} \\ \mathbf{0}_{2\times 1} & -\frac{U}{L}\begin{bmatrix} \bar{a}_{11} & L\bar{a}_{12} \\ \frac{1}{L}\bar{a}_{21} & \bar{a}_{22} \end{bmatrix} \end{bmatrix}$$

$$\mathbf{B}(\boldsymbol{\nu}) = \begin{bmatrix} 1 - t_d & \mathbf{0}_{2\times 1} \\ \mathbf{0}_{1\times 2} & \frac{U^2}{L}\begin{bmatrix} \bar{b}_{11} & \bar{b}_{12} \\ \frac{1}{L}\bar{b}_{21} & \frac{1}{L}\bar{b}_{22} \end{bmatrix} \end{bmatrix}$$

The control objectives are: (1) to force the ship to converge to and follow the path, and (2) let the desired surge speed u_d *be adjustable online. The desired output path is chosen as:*

$$\boldsymbol{\eta}_d(\theta) = [x_d(\theta), y_d(\theta), atan(y'_d/x'_d)]^\top \tag{10.106}$$

where $x_d(\theta) = \theta$ *and* $y_d(\theta) = 500\sin(\frac{2\pi}{4000}\theta)$*. The first part of the control objective is solved as* $\lim_{t\to\infty}|z_1(t)| = 0$ *is guaranteed. To solve the speed specification in the second part, we note the relationship:*

$$u_d(t) = \sqrt{x'_d(\theta(t))^2 + y'_d(\theta(t))^2}\,v_s(\theta(t), t)$$

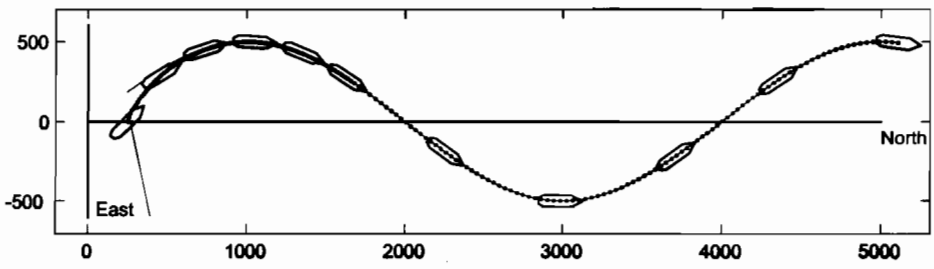


Figure 10.5: Output response of the container ship following the desired sinusoidal path on the ocean surface, for the adaptive system.

Let $u_d(t) = u_0$ be a constant desired set-point for u_d, then:

$$\begin{aligned} v_s(\theta) &= \frac{u_0}{\sqrt{x_d'(\theta)^2 + y_d'(\theta)^2}} \\ v_s'(\theta) &= -\frac{x_d'(\theta)x_d''(\theta) + y_d'(\theta)y_d''(\theta)}{\left[x_d'(\theta)^2 + y_d'(\theta)^2\right]^{3/2}} u_0 \quad \textit{and} \quad \frac{\partial v_s(\theta,t)}{\partial t} = 0 \end{aligned} \tag{10.107}$$

where the commanded surge speed is:

$$u_0 = \begin{cases} 4\,(\mathrm{m/s}), & 0 \le t < 400\ (\mathrm{s}) \\ 10\,(\mathrm{m/s}), & 400 \le t \le 800\ (\mathrm{s}) \end{cases} \tag{10.108}$$

In the model (10.105), the unknown constant parameter vector $\boldsymbol{\Theta}$ and the corresponding regressor matrix $\boldsymbol{\varphi}$ are:

$$\boldsymbol{\Theta} = [\bar{X}_u, \bar{X}_{|u|u}, \bar{a}_{11}, \bar{a}_{12}, \bar{a}_{21}, \bar{a}_{22}, \mathbf{w}^\top]^\top \tag{10.109}$$

$$\boldsymbol{\varphi}(\boldsymbol{\eta}, \boldsymbol{\nu}) = \begin{bmatrix} \frac{1}{2}\rho L^2 U u & \frac{1}{2}\rho L^2 |u| u & 0 & 0 & 0 & 0 & \\ 0 & 0 & \frac{U}{L}v & Ur & 0 & 0 & \mathbf{R}^\top(\psi) \\ 0 & 0 & 0 & 0 & \frac{U}{L^2}v & \frac{U}{L}r & \end{bmatrix}$$

where $\bar{X}_u$, $\bar{X}_{|u|u}$, and $\bar{a}_{ij}$ are nondimensional coefficients, and are given units by the corresponding multiplying terms in the regressor where ρ is the sea water density. The dynamic equation (10.105) becomes

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}\mathbf{B}(\boldsymbol{\nu})\,\mathbf{u} + \mathbf{M}^{-1}\boldsymbol{\varphi}(\boldsymbol{\eta}, \boldsymbol{\nu})\,\boldsymbol{\Theta} \tag{10.110}$$

and together with the kinematic equation (10.104) the plant is in the form of (10.73). This gives:

$$\begin{aligned} \mathbf{z}_1 &= \boldsymbol{\eta} - \boldsymbol{\eta}_d(\theta) \\ \mathbf{z}_2 &= \boldsymbol{\nu} - \boldsymbol{\alpha}_1(\boldsymbol{\eta}, \boldsymbol{\eta}_d(\theta), \boldsymbol{\eta}_d'(\theta), v_s(\theta,t)) \\ \boldsymbol{\alpha}_1 &= \mathbf{R}^\top(\psi)\left[-\mathbf{K}_p\mathbf{z}_1 + \boldsymbol{\eta}_d'(\theta)v_s(\theta,t)\right] \\ \boldsymbol{\sigma}_1 &= \dot{\mathbf{R}}^\top(r)\mathbf{R}(\psi)\boldsymbol{\alpha}_1 + \mathbf{R}^\top(\psi)\left[-\mathbf{K}_p\mathbf{R}(\psi)\boldsymbol{\nu} + \boldsymbol{\eta}_d'(\theta)\frac{\partial v_s(\theta,t)}{\partial t}\right] \\ \boldsymbol{\nu}_2 &= \mathbf{R}^\top(\psi)\left[\mathbf{K}_p\boldsymbol{\eta}_d'(\theta) + \boldsymbol{\eta}_d''(\theta)v_s(\theta,t) + \boldsymbol{\eta}_d'(\theta)v_s'(\theta,t)\right] \end{aligned}$$

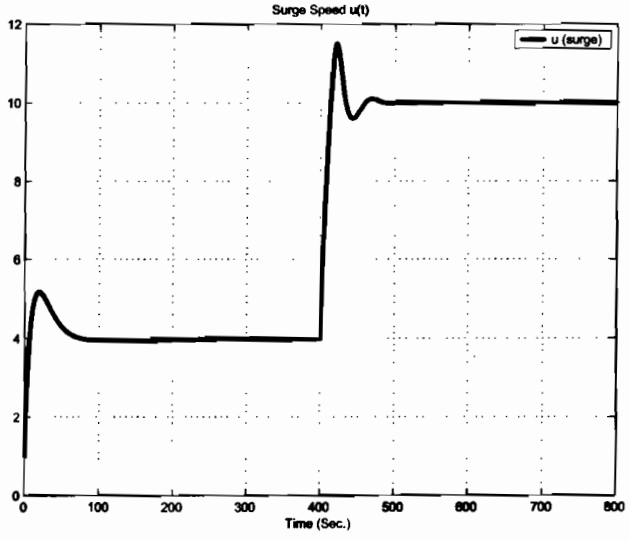


Figure 10.6: Plot of the surge speed u as a function of time t.

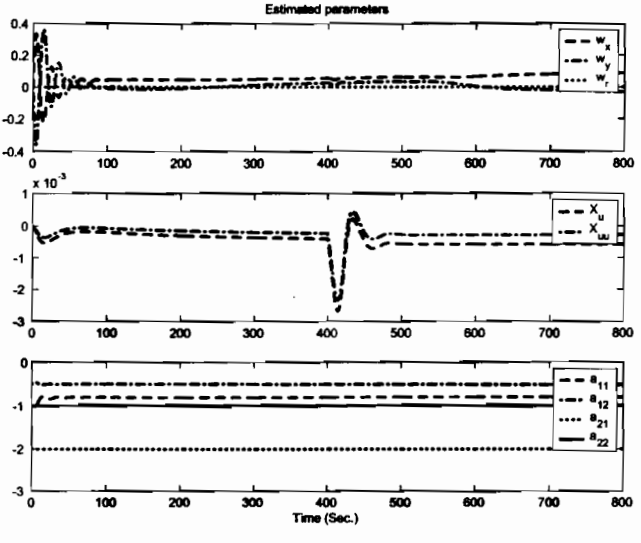


Figure 10.7: The parameter estimates $\hat{\Theta}$ as a function of time t.

The control law, the adaptive update law, and the maneuvering update law are then given by the closed-loop system according to Figure 10.2:

$$\textit{Plant} \;:\; \begin{cases} \dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \\ \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{N}\boldsymbol{\nu} = \mathbf{B}\mathbf{u} + \mathbf{R}^{\top}(\psi)\mathbf{w} \\ \quad \text{inputs} = [\mathbf{u}, \mathbf{w}] \\ \quad \text{outputs} = [\boldsymbol{\eta}, \boldsymbol{\nu}] \end{cases}$$

$$\textit{Guidance} \;:\; \begin{cases} \begin{bmatrix} \dot{\theta} \\ \dot{\omega}_s \end{bmatrix} = \begin{bmatrix} v_s(\theta,t) - \omega_s \\ -2\mu_1 \left[p_1 \mathbf{z}_1^{\top} \boldsymbol{\eta}_d'(\theta) + p_2 \mathbf{z}_2^{\top} \boldsymbol{\nu}_2 \right] - \lambda \omega_s \end{bmatrix} \\ \quad \text{inputs} = [\theta, t] \\ \quad \text{outputs} = [\boldsymbol{\eta}_d(\theta), \mathbf{y}_{cv}(\theta), v_s(\theta,t), \frac{\partial v_s(\theta,t)}{\partial t}, \frac{\partial v_s(\theta,t)}{\partial \theta}] \end{cases}$$

$$\textit{Control} \;:\; \begin{cases} \dot{\hat{\boldsymbol{\Theta}}} = 2p_2 \boldsymbol{\Gamma} \boldsymbol{\varphi}^{\top} \mathbf{M}^{-\top} \mathbf{z}_2 \\ \mathbf{u} = \mathbf{B}^{-1}\mathbf{M}[-\frac{p_1}{p_2}\mathbf{R}^{\top}(\psi)\mathbf{z}_1 - \mathbf{K}_d \mathbf{z}_2 - \mathbf{M}^{-1}\boldsymbol{\varphi}\hat{\boldsymbol{\Theta}} + \boldsymbol{\sigma}_1 + \boldsymbol{\nu}_2 v_s(\theta,t)] \\ \quad \text{inputs} = [\boldsymbol{\eta}, \boldsymbol{\nu}, \boldsymbol{\eta}_d(\theta), \mathbf{y}_{cv}(\theta), v_s(\theta,t)] \\ \quad \text{outputs} = [\mathbf{u}, \hat{\boldsymbol{\Theta}}] \end{cases}$$

The numerical values are taken from Appendix E.1.3 in Fossen (1994); $m = 21.2 \cdot 10^6$, $X_{\dot{u}} = -6.38 \cdot 10^5$, $\bar{b}_{11} = \bar{b}_{12} = -0.2081$, $\bar{b}_{21} = -\bar{b}_{22} = -1.5238$. *The true unknown parameter vector is:*

$$\boldsymbol{\Theta} = [-4.226 \cdot 10^{-4}, -4.226 \cdot 10^{-4}, -0.7072, -0.2860, -4.1078, -2.6619,\ 0.1, 0, 0]^{\top}$$

and the controller settings are: $\mathbf{K}_p = \text{diag}\{0.02,\ 0.02,\ 0.2\}, \mathbf{K}_d = \text{diag}\{0.1,\ 0.1,\ 10\}$, $\mathbf{P}_1 = 0.01 \cdot \mathbf{I}$, $\mathbf{P}_2 = 10 \cdot \mathbf{I}$, $\lambda = \mu_1 = 200$ *and* $\boldsymbol{\Gamma} = 10^{-7} \cdot \text{diag}\{1, 1, 10^5, 10^5, 10^5, 10^5, 5 \cdot 10^4, 5 \cdot 10^4, 10^4\}$. *The initial conditions are* $\boldsymbol{\eta}(0) = [250, 0, \frac{\pi}{4}]^{\top}$, $\boldsymbol{\nu}(0) = [1, 0, 0]^{\top}$, $\theta(0) = 250$, $\omega_s(0) = 0$, *and* $\hat{\boldsymbol{\Theta}}(0) = [0, 0, -1, -0.5, -2, -1, 0, 0, 0]^{\top}$.

Figure 10.5 shows how the ship smoothly converges to and accurately follows the path. The smooth entrance to the path is characteristic for maneuvering systems. In Figure 10.6 it is shown how the surge speed u is first controlled to $u_0 = 4$ (m/s) and then at $t = 400$ (s) to $u_0 = 10$ (m/s). Hence, the surge speed is adjustable online by set-points specified by the operator. The time series of the parameter estimates $\hat{\boldsymbol{\Theta}}(t)$ is shown in Figure 10.7. We notice that all parameters stay bounded, and that some converge close to their true values.

Example 10.2 (Comparison of Tracking and Maneuvering Control)
In this experiment the resulting response of a tracking design and a maneuvering design is compared when the controls saturate. This is done by deliberately commanding a forward speed that is faster than the ship is able to follow. For simplicity, the weather disturbance is $\mathbf{w} = \mathbf{0}$, *and the experiment is executed as a nonadaptive case. The maneuvering controller is simply obtained from Example 10.1 with* $\boldsymbol{\Gamma} = \mathbf{0}$ *and* $\hat{\boldsymbol{\Theta}}(0) = \boldsymbol{\Theta}$. *The tracking controller is implemented by letting* $\omega_s \equiv 0$ *so that* $\dot{\theta} = v_s(\theta)$.

The saturation levels are $|\delta_i| \leq 20°$ *and* $|T| \leq 10^6$ (N). *This means, in practise, that the ship can move with maximum speed* $u \approx 8$ (m/s), *while the commanded surge speed is again specified according to (10.108) and is therefore infeasible for* $t \geq 400$ (s). *The controller settings are:* $\mathbf{K}_p = \text{diag}\{0.2,\ 0.2,\ 0.4\}$, $\mathbf{K}_d = \text{diag}\{1, 1,\ 10\}$, $\mathbf{P}_1 = 0.005 \cdot \mathbf{I}$, $\mathbf{P}_2 = 10 \cdot \mathbf{I}$, *and* $\lambda = \mu_1 = 200$. *The initial conditions are* $\boldsymbol{\eta}(0) = [0, 0, \frac{\pi}{4}]^{\top}$, $\boldsymbol{\nu}(0) = [4, 0, 0]^{\top}$, $\theta(0) = 0$, *and* $\omega_s(0) = 0$.

The position responses are shown in Figure 10.8. While, not surprisingly, the tracking

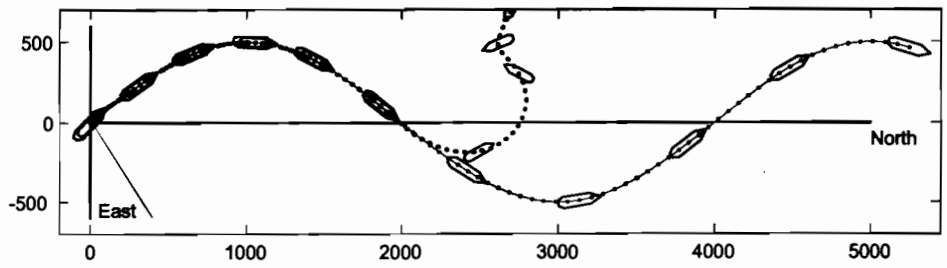


Figure 10.8: Position response of the tracking system (small ship) and the maneuvering system (large ship). As expected, the ship based on the tracking controller goes unstable when an unfeasible desired surge speed is commanded. The ship based on the maneuvering system, on the other hand, is seen to keep following the path.

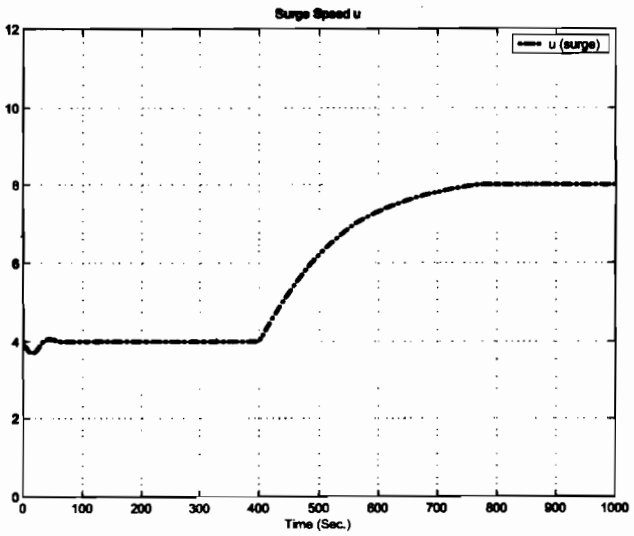


Figure 10.9: Surge speed u of the maneuvering based system. Note how the speed, due to saturation of the control thrust, converges to maximum possible speed ($\approx 8\,\mathrm{m/s}$) instead of the specified speed $u_0 = 10\,\mathrm{m/s}$.

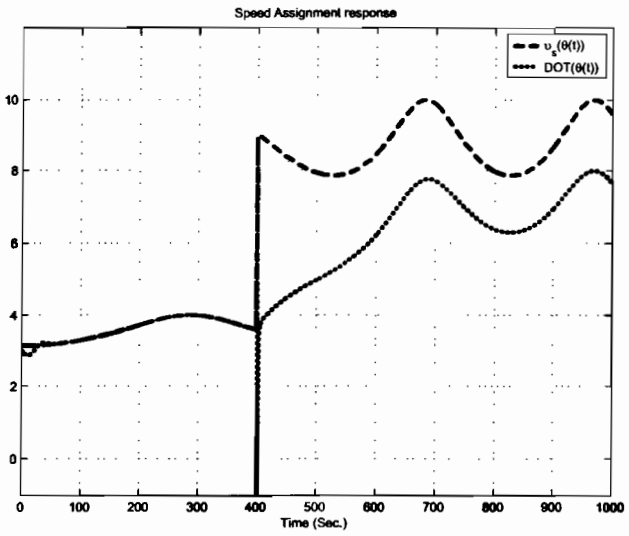


Figure 10.10: Plots of $v_s(\theta(t))$ and $\dot{\theta}(t)$. While the specified speed assignment $v_s(\theta(t))$ corresponds to the infeasible surge speed $u_0 = 10\,\mathrm{m/s}$, the resulting response of $\dot{\theta}(t)$ tracks a slower value according to the state of the ship.

system goes unstable, it is observed that the maneuvering system moves along the path, in spite of the infeasible u_0. The surge speed u of the maneuvering system is seen in Figure 10.9 to converge to the maximum possible speed of 8 (m/s). Figure 10.10 reveals some of the secret. The speed assignment $v_s(\theta(t))$ strictly corresponds to the specified speed u_0 according to (10.107). However, $\dot{\theta}(t)$ follows at a value about 2 (m/s) slower, and the result is that $\theta(t)$, and thus $\boldsymbol{\eta}_d(\theta(t))$, moves not faster than the ship is able to follow.

10.2.3 Maneuvering Control of Underactuated Ships

The maneuvering controller of the previous section assumes that the ship is fully actuated–i.e. $\tau = [\tau_{\text{surge}}, \tau_{\text{sway}}, \tau_{\text{yaw}}]^\top$. For ships that only have two controls:

- forward thrust τ_{surge} (main propeller)
- yaw moment τ_{yaw} (rudder)

a different approach must be applied since $\tau_{\text{sway}} = 0$. One way to solve this problem is to define an output mapping of dimension two which can be stabilized by using two controls. Two approaches for this are:

Surge-Yaw Maneuvering Control using only two Controls

In Fossen *et al.* (2003) the output mapping is defined as the LOS angle ψ using position measurements (x, y) and the way-point database (x_k, y_k) for $k = 1, \ldots, n$, see Section 5.3.

In this approach the sway velocity v is proven to be uniformly bounded while the two controls τ_{surge} and τ_{yaw} are used to control (ψ, u). The equilibrium pint $(\psi - \psi_d, u - u_d) = (0, 0)$ of the closed-loop system is UGAS. This type of controller can be implemented for ships equipped with a conventional rudder and a main propeller. A case study exploring this is found in Fossen *et al.* (2003).

Surge-Sway Maneuvering Control using only two Controls

In Skjetne *et al.* (2002*a*) a maneuvering controller for underactuated ships is derived by using the inputs τ_{surge} and τ_{yaw} to control the positions (x, y). Hence, speed is controlled indirectly using $U = \sqrt{\dot{x}^2 + \dot{y}^2}$. The main problem in doing this, is that the ship must be prevented from turning around. This idea is based on the joint work of Lindegaard (1997), Fossen *et al.* (1998) and Berge *et al.* (1999) where the yaw dynamics is stabilized by locating a body-fixed coordinate frame in the bow or ahead of the ship. The point in which the coordinate system is located is referred to as the *Vessel Reference Point* (VRP). Hence, the ship bow, will be forced to track the desired path, while the aft part of the ship may deviate from the path. This will of course only work if the yaw mode is stable. Fortunately, this is easy to achieve, since introduction of a VRP implies that the ship behaves like a pendulum in the gravity field, where the gravitational force corresponds to the mean environmental disturbances. Hence, moving the VRP ahead corresponds to increasing the length of the rope in which the pendulum hangs. This stabilizes the yaw dynamics. A complete case study showing the performance of the maneuvering controller in cruise condition and docking is given in Skjetne *et al.* (2002*a*)

10.3 Exercises

Exercise 10.1 *Simulate the LOS and linear quadratic optimal cross-tracking algorithms in Sections 10.1.2 and 10.1.3 using one the of ship models in the GNC toolbox. Comment on the performance of the two algorithms. What happens if you add a constant wind disturbance in the sway direction? Will both algorithms behave in a similar manner?*

Exercise 10.2 *In Section 10.1.1 it was shown that the transfer function between the cross-track error and the rudder angle was:*

$$e_2(s) = h_\delta(s)\delta(s) + h_b(s)r_b(s) \tag{10.111}$$

where:

$$h_\delta(s) = \frac{e_2}{\delta}(s) = \frac{KU}{s^2(1+Ts)}, \quad h_b(s) = \frac{e_2}{r_b}(s) = \frac{U}{s^2(1+Ts)} \tag{10.112}$$

Consider the PID control law:

$$\delta = -K_p\left(1 + T_d s + \frac{1}{T_i s}\right)e_2 = -K_p\left(\frac{1 + T_i s + T_d T_i s^2}{T_i s}\right)e_2 \tag{10.113}$$

and show that:

$$\lim_{t\to\infty} e_2(t) = 0 \tag{10.114}$$

for r_b =constant. Is this possible for a PD-controller? (Hint: Use the final value theorem).

Chapter 11

Positioning Systems

This chapter discusses control systems for station-keeping and low-speed maneuvering. These systems are commonly known as dynamic positioning (DP) systems. The Norwegian classification society, DnV (1990) defines a dynamically positioned vessel as a free-floating vessel which maintains its position (fixed location or predetermined track) exclusively by means of thruster. It is, however, possible to exploit rudder forces in DP also by using the propeller to generate rudder lift forces (Lindegaard and Fossen 2002).

For ships that are anchored, additional spring forces are introduced into the control model. These systems are referred to as position mooring (PM) systems.

Finally, optimality with respect to changing weather conditions will be discussed using the concept of weather optimal positioning control (WOPC).

11.1 Models for Station-Keeping

For DP and PM systems a low-frequency (LF) control model will be employed for feedback since dynamics at higher frequencies are negligible in station-keeping.

11.1.1 Vessel Kinematics and Dynamics

The vessel kinematics and dynamics are (see Section 3.5.1):

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{11.1}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} + \mathbf{K}\boldsymbol{\eta} = \boldsymbol{\tau} \tag{11.2}$$

where $\boldsymbol{\nu} = [u, v, r]^\top$, $\boldsymbol{\eta} = [n, e, \psi]^\top$ (North-East positions and heading), and:

$$\mathbf{M} = \mathbf{M}^\top = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix} \tag{11.3}$$

$$\mathbf{D} = \mathbf{D}^\top = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix} \tag{11.4}$$

$$\mathbf{K} = \text{diag}\{k_{11}, k_{22}, k_{33}\} \tag{11.5}$$

The LF assumption implies that added mass and damping can be computed at zero wave frequency and that linear damping is sufficient. The spring stiffness matrix $\mathbf{K}$ is only used in PM systems. It is assumed that spring forces and moment in *surge, sway,* and *yaw* are decoupled. The models (11.1)–(11.2) can be transformed to VP coordinates $\boldsymbol{\eta}_p = \mathbf{R}^\top(\psi)\boldsymbol{\eta}$ by using the results in Section 3.3.2. Consequently:

$$\dot{\boldsymbol{\eta}}_p = \boldsymbol{\nu} \tag{11.6}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} + \mathbf{K}\boldsymbol{\eta}_p = \boldsymbol{\tau} \tag{11.7}$$

In order to incorporate the limitations of the propellers, the model is augmented by actuator dynamics. The simplest way of doing this is to define three time constants in *surge, sway,* and *yaw* such that:

$$\dot{\boldsymbol{\tau}} = \mathbf{A}_{\text{thr}}(\boldsymbol{\tau} - \boldsymbol{\tau}_{\text{com}}) \tag{11.8}$$

where $\boldsymbol{\tau}_{\text{com}}$ is the commanded thrust and $\mathbf{A}_{\text{thr}} = -\text{diag}\{1/T_{\text{surge}}, 1/T_{\text{sway}}, 1/T_{\text{yaw}}\}$ is a diagonal matrix containing the time constants. The resulting state-space model becomes:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\boldsymbol{\tau}_{\text{com}} \tag{11.9}$$

where $\mathbf{x} = [\boldsymbol{\eta}_p^\top, \boldsymbol{\nu}^\top, \boldsymbol{\tau}^\top]^\top$ and:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} & \mathbf{M}^{-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{thr}} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\mathbf{A}_{\text{thr}} \end{bmatrix} \tag{11.10}$$

This model is the basis for the LQ controller.

11.1.2 DP and PM Thrust Models

Most DP ships use thrusters to maintain their position and heading. Both *fixed pitch* (FP) and *controllable pitch* (CP) propellers are used for this purpose; see Section 6.2 in Fossen (1994) for a more detailed discussion on FP and CP propellers.

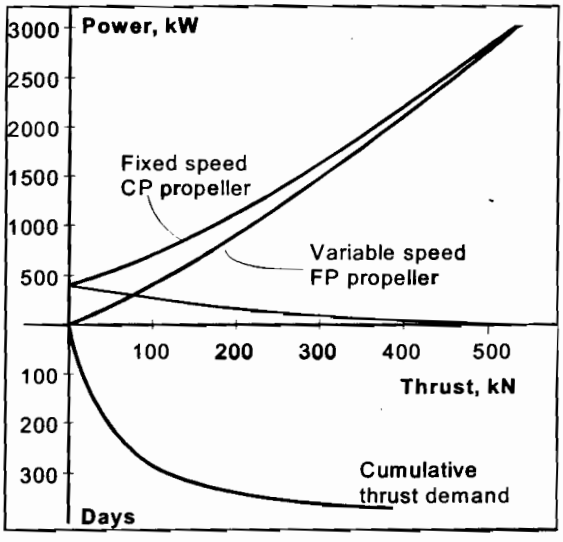


Figure 11.1: Power consumption of fixed speed CP and variable speed FP propellers.

Fixed Speed CP and Variable Speed FP Propellers

The thrust F from a *variable speed FP* propeller can be modelled as:

$$F(n) = Kn\,|n|\,, \qquad (\text{or } F(n) = Kn) \tag{11.11}$$

where K = constant is the thrust coefficient and n (rpm) is the propeller revolution. Some propellers show a linear behavior in n while others are quadratic. Even combinations of the linear and quadratic behavior are observed in practice.

CP propellers are screw blade propellers where the blades can be turned under the control of a hydraulic servo. This introduces a second control variable, pitch p, which is used to obtained the desired thrust F for different propeller revolutions n. If P is the "traveled distance per revolution", D is the propeller diameter then $p = P/D$ represents the pitch ratio.

The thrust from a *fixed speed CP* propeller can be approximated by:

$$F(n,p) = K(n)|(p - p_0)|(p - p_0), \qquad (\text{or } F(n,p) = K(n)(p - p_0)) \tag{11.12}$$

where the force coefficient $K(n)$ now depend on the propeller revolution. Again, thrust is quadratic alternatively linear in $p - p_0$ or combinations of both. The pitch offset is denoted as p_0. For DP ships using fixed speed CP propellers it is common to operate at one or two fixed propeller revolutions such that only p is used for active control by the DP system; see Example 11.1.

For ships in transit a constant demand for thrust and power suggest that a fixed speed CP propeller should be used while low-speed applications like DP operations require little thrust in good weather suggesting that a variable speed FP propeller might be advantageous; see 11.1. Notice that the fixed speed CP propeller require power also at zero thrust.

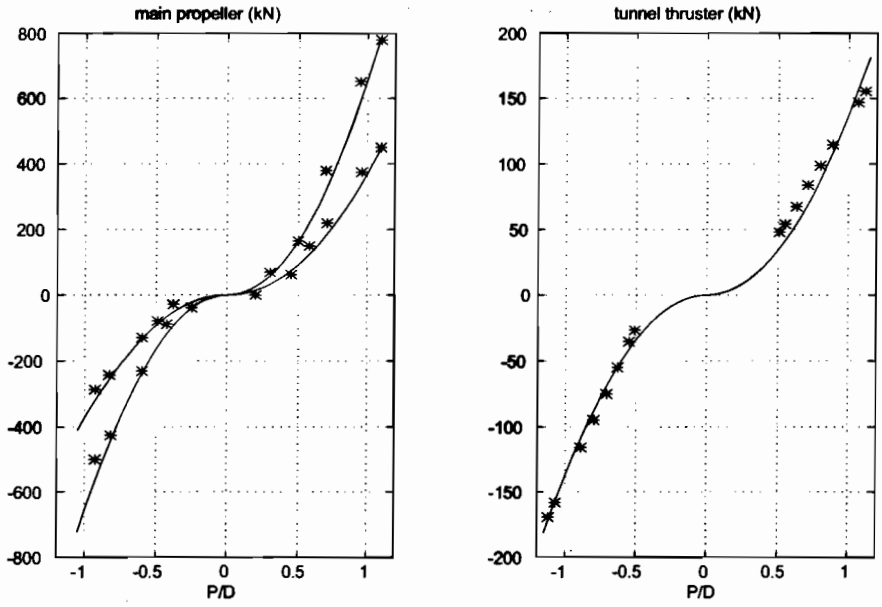


Figure 11.2: Thrust $F(n,p) = K(n)p\,|p|$ versus pitch p for a main propeller (left plot) and a tunnel thruster (right plot). The asterisks are experimental measured values and the solid lines are least-squares fits to a quadratic model.

Example 11.1 (Experimental Thrust Characteristics)
The thrust of a main propeller and a tunnel thruster for a supply vessel is shown in Figure 11.2 (Fossen et al. *1996). The asterisks represent the measured values, while the solid lines are least-squares fits to the quadratic thrust function (11.12). The main propeller operated at $n = 122$ (rpm) and $n = 160$ (rpm), while the tunnel thruster ran at $n = 236$ (rpm) resulting in:*

main propeller	$F(122,p) = 370\,\|p\|\,p$	$F(236,p) = 137\,\|p\|\,p$
tunnel thruster	$F(160,p) = 655\,\|p\|\,p$	

Actuator Configuration and Thrust Coefficient Matrices

The commanded forces and moment $\boldsymbol{\tau}_{\text{com}} \in \mathbb{R}^3$ (*surge, sway,* and *yaw*) can be written (see Section 7.5):

$$\boldsymbol{\tau}_{\text{com}} = \mathbf{T}(\boldsymbol{\alpha}) \underbrace{\mathbf{K}\mathbf{u}}_{\mathbf{f}} \tag{11.13}$$

where $\mathbf{f} = \mathbf{K}\mathbf{u} \in \mathbb{R}^r$ (r = number of thrusters) is the thrust force vector and $\mathbf{u} \in \mathbb{R}^r$ is a DP

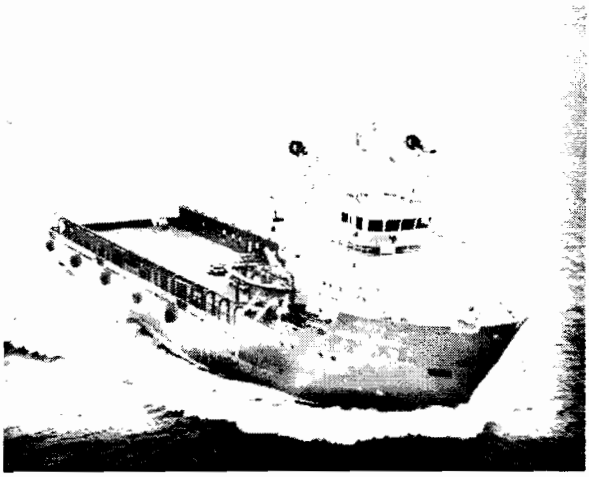


Figure 11.3: Offshore supply vessel.

control variable defined as:

$$\begin{array}{lll} \text{CPP:} & \mathbf{u} = [|p_1|p_1, |p_2|p_2, ..., |p_r|p_r]^\top, & (\text{or } \mathbf{u} = [p_1, p_2, ..., p_r]^\top) \\ \text{FP:} & \mathbf{u} = [|n_1|n_1, |n_2|n_2, ..., |n_r|n_r]^\top, & (\text{or } \mathbf{u} = [n_1, n_2, ..., n_r]^\top) \end{array} \tag{11.14}$$

The *thrust coefficient matrix* $\mathbf{K}$ is a diagonal matrix of thrust coefficients defined as:

$$\mathbf{K} = \text{diag}\{K_1(n_1), K_2(n_2), ..., K_r(n_r)\} \tag{11.15}$$

The *actuator configuration matrix* $\mathbf{T}(\boldsymbol{\alpha}) \in \mathbb{R}^{3\times r}$ only depends on the location of the actuators and possible angles $\boldsymbol{\alpha}$ used for rotatable thrusters (azimuth thruster). Computation of $\mathbf{T}(\boldsymbol{\alpha})$ is best illustrated by considering using a real ship.

Example 11.2 (Actuator Configuration Matrix)
Consider the supply vessel in Figure 11.3. Let us assume that the ship is equipped with two main propellers (aft of the ship), two tunnel thrusters and two azimuth thrusters which can be rotated to arbitrary angles α_1 and α_2, and therefore produce thrust in different directions, see Figure 11.4. Hence, we have 8 control variables (6 rpm set-points and 2 azimuth angles) for 3 DOF. The control variables are assigned according to (clockwise numbering of u_i):

u_1, α_1	*fore azimuth thruster*	u_4	*aft tunnel thruster*
u_2	*fore tunnel thruster*	u_5	*starboard main propeller*
u_3, α_2	*aft azimuth thruster*	u_6	*port main propeller*

Using the results of Section 7.5 implies that:

$$\mathbf{K} = \text{diag}\{K_1, K_2, K_3, K_4, K_5, K_6\} \tag{11.16}$$

$$\mathbf{T}(\boldsymbol{\alpha}) = \begin{bmatrix} \cos\alpha_1 & 0 & \cos\alpha_2 & 0 & 1 & 1 \\ \sin\alpha_1 & 1 & \sin\alpha_2 & 1 & 0 & 0 \\ l_1\sin\alpha_1 & l_2 & -l_3\sin\alpha_2 & -l_4 & -l_5 & l_6 \end{bmatrix} \tag{11.17}$$

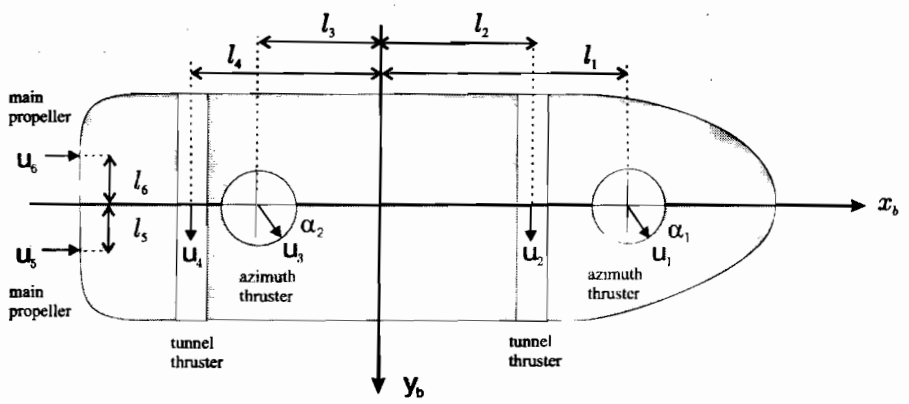


Figure 11.4: Schematic drawing showing the thruster configuration for a typical supply vessel.

where l_i $(i = 1, \ldots, r)$ are the moment arms in yaw. It is also seen that $l_5 = l_6$ (symmetrical location of main propellers). The thrust demands are defined such that a positive thruster force/moment results in positive motion according to the VP axis system.

One of the advantages of the model representation (11.13) is that input uncertainties only appear in the diagonal force coefficient matrix $\mathbf{K}$, since $\mathbf{T}(\alpha)$ will be perfectly known. In fact, this decomposition is highly advantageous since it can be exploited when designing the feedback control system where robust measures for uncertainties in $\mathbf{K}$ must be taken.

11.1.3 Environmental Disturbances

A ship will be exposed to *waves*, *currents*, and *wind*. The observer-controller must be robust and compensate for environmental disturbances. These are the most important design requirements in an industrial ship control system since a full-state feedback controller will not work in bad weather unless the environmental loads are included in the design specifications. In commercial DP systems it is therefore necessary to:

- include *integral action* in the controller to compensate for slowly-varying disturbances (bias) due to ocean currents and wave drift forces (2nd-order wave theory).
- use *feedforward control* to compensate for *mean* wind disturbances. Wind gust cannot be compensated for since the actuators do not the have the capacity of moving a large vessel in the frequency range of the wind gust.
- include *wave filtering* in the state estimator (observer) to avoid 1st-order wave induced oscillations fed back to the control system. Again, the actuators cannot move a large vessel fast enough to suppress the disturbances.

A closed loop DP system is shown in Figure 11.5.

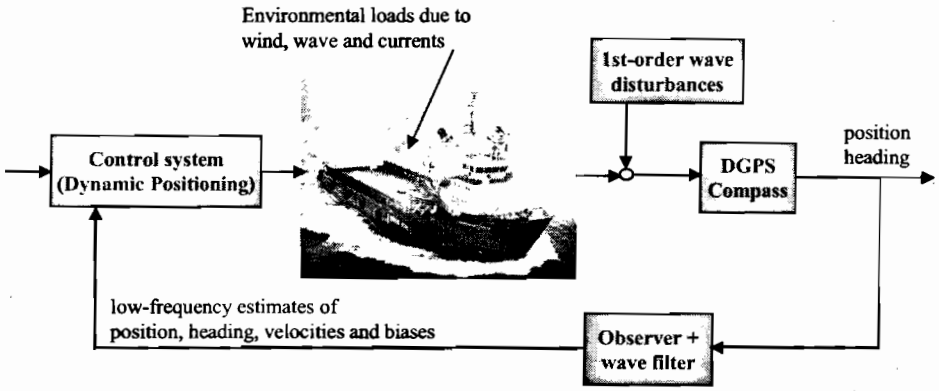


Figure 11.5: Dynamic positioing system.

11.2 Dynamic Positioning (DP) Systems

In the 1960s systems for automatic control of the horizontal position, in addition to the course, were developed. Systems for the simultaneous control of the three horizontal motions (surge, sway, and yaw) are today commonly known as *dynamic positioning (DP) systems.* More recently anchored positioning systems or *position mooring (PM) systems* have been designed; see Section 11.3. For a free floating vessel in DP the thrusters are the prime actuators for station-keeping, while for a PM system the assistance of thrusters are only complementary since most of the position keeping is provided by a deployed anchor system. Different DP applications are described more closely in Strand and Sørensen (2000).

DP systems have traditionally been a *low-speed* application, where the basic DP functionality is either to keep a fixed position and heading or to move slowly from one location to another (*marked positioning*). In addition, specialized tracking functions for cable and pipe-layers, and operations of remotely operated vehicles (ROVs) have been included. The traditional *autopilot* and *way-point tracking* functionalities have also been included in modern DP systems. The trend today is that *high-speed* operation functionality merges with classical DP functionality, resulting in a *unified system* for all speed ranges and types of operations.

The first DP systems were designed using conventional PID controllers in cascade with low pass and/or notch filters to suppress the wave-induced motion components. This was based on the assumption that the interactions were negligible (Sargent and Cowgill 1976 and Morgan 1978). From the middle of the 1970's a new model-based control concept utilizing stochastic optimal control theory and Kalman filtering techniques was employed with the DP problem by Balchen *et al.* (1976). The Kalman filter is used to separate the LF and WF motion components such that only feedback from the LF motion components is used. Later extensions and modifications of this work have been proposed by numerous authors; see Balchen *et al.* (1980a, 1980b), Grimble *et al.* (1980a, 1980b), Fung and Grimble (1983), Sælid *et al.* (1983), and more lately by Fossen *et al.* (1996), Sørensen *et al.* (1996), Fossen and Grøvlen (1998), Fossen and Strand (1999*a*) and Sørensen *et al.* (2000).

Roll and Pitch Damping in DP

Traditionally DP systems have been designed for 3 DOF low-speed trajectory tracking control by means of thrusters and propellers. However, extensions to 5 DOF control for the purpose of roll and pitch damping of semi-submersibles has been proposed by Sørensen and Strand (1998). It is well known that for marine structures with a small-waterplane-area and low metacentric height, which results in relatively low hydrostatic restoring compared to the inertia forces, an unintentional coupling phenomenon between the vertical and the horizontal planes through the thruster action can be invoked. Examples are found in semi-submersibles and SWATHs, which typically have natural periods in roll and pitch in the range of 35–65 (s). If the inherent vertical damping properties are small, the amplitudes of roll and pitch may be emphasized by the thruster's induction by up to 2°–5° in the resonance range. These oscillations have caused discomfort in the vessel's crew and have in some cases limited the operation. Hence, both the horizontal and vertical planes DOF should be considered in the controller design as proposed in Sørensen and Strand (2000).

Optimal Set-Point Chasing in DP for Drilling and Intervention Vessels

Further extension in the development of DP systems includes extended functionality adapted the particular marine operation considered. In Sørensen *et al.* (2001) a function for optimal set-point chasing in DP of drilling and intervention vessels is proposed in order to minimize riser angle offsets at the sea bed and on the vessel.

11.2.1 Thrust Allocation in DP Systems

In order to implement a DP control system a thrust allocation algorithm is needed. Thrust (control) allocation involves computing the thruster inputs $u_i = |p_i|p_i$ and azimuth angles α_i $(i = 1, \ldots, r)$ from:

$$\boldsymbol{\tau}_{\text{com}} = \mathbf{T}(\boldsymbol{\alpha})\mathbf{f} \tag{11.18}$$

$$\mathbf{f} = \mathbf{K}\mathbf{u} \tag{11.19}$$

for given controller commands $\boldsymbol{\tau}_{\text{com}}$. In the unconstrained case (no saturation) with $\boldsymbol{\alpha} =$ constant, a formula for $\mathbf{f}$ can be found by minimizing the thrust vector $\mathbf{f}$. This results in the generalized inverse:

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{T}^{\dagger}(\boldsymbol{\alpha})\boldsymbol{\tau}_{\text{com}}, \qquad \mathbf{T}^{\dagger}(\boldsymbol{\alpha}) = \mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})[\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})]^{-1} \tag{11.20}$$

where $\mathbf{W} = \mathbf{W}^T > 0$ is a positive definite weighting matrix, usually chosen to be diagonal. $\mathbf{W}$ should be selected so that using the tunnel and azimuth thrusters is less expensive (small K_i-value) than using the main propellers (large K_i-value).

Optimal solutions for systems which $\boldsymbol{\alpha}$ is allowed to vary and where thrust $\mathbf{f}$ is limited are found in Section 7.5.

Example 11.3 (Supply Vessel Thrust Allocation)
Consider the supply vessel in Example 11.2. The generalized inverse:

$$\mathbf{T}^{\dagger}(\boldsymbol{\alpha}) = \frac{1}{\det[\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})]}\mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})\,\mathrm{adj}[\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^{\top}(\boldsymbol{\alpha})] \tag{11.21}$$

will be a function of the azimuth angles α_1 and α_2. The expression for the determinant in (11.21) will be non-zero for all combinations of α_1 and α_2, since the vessel has more actuators than needed for 3 DOF stabilization (overactuated). However, for some vessels a singular configuration may exist, that is the determinant becomes zero for certain combinations of α_i ($i = 1, \ldots, r$). The expression for the determinant can also be used to compute optimal angles α_1 and α_2 in a minimum energy sense by simply maximizing the determinant with respect to α_1 and α_2.

11.2.2 Linear Quadratic Optimal Control

The LQ controller is designed under assumption of *full state feedback* requiring that all states are measured or at least estimated. Navigation systems for DP, conventional linear observer design (*Kalman filtering*) with extensions to nonlinear and passive observer design are treated in detail in Section 6.2. The observer is also needed in the case when all states are measured since it is necessary to separate the LF and WF motions (*wave filtering*). The WF estimates (1st-order wave-induced motion) should not be fed back, since this will cause wear and tear on the thruster actuators.

The goal of this section is to design an optimal control law with wind feedforward and integral action. Consider the vessel model (11.9) in the form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\underbrace{(\boldsymbol{\tau}_{\mathrm{LQ}} + \boldsymbol{\tau}_{\mathrm{FF}})}_{\boldsymbol{\tau}_{\mathrm{com}}} \tag{11.22}$$

where we have assumed that the commanded input $\boldsymbol{\tau}_{\mathrm{com}}$ can be divided into two parts; optimal feedback $\boldsymbol{\tau}_{\mathrm{LQ}}$, and wind feedforward $\boldsymbol{\tau}_{\mathrm{FF}}$.

Optimal Feedback Control

The LQ control objective is to obtain $\mathbf{x} = \mathbf{0}$, that is $\boldsymbol{\eta}_p = \boldsymbol{\nu} = \boldsymbol{\tau} = \mathbf{0}$. Hence, we can compute $\boldsymbol{\tau}_{\mathrm{LQ}}$ by minimizing the performance index:

$$J = \min_{\boldsymbol{\tau}_{\mathrm{LQ}}} \left\{ \frac{1}{2} \int_0^T (\mathbf{x}^{\top}\mathbf{Q}\mathbf{x} + \boldsymbol{\tau}_{\mathrm{LQ}}^{\top}\mathbf{R}\boldsymbol{\tau}_{\mathrm{LQ}})\, d\tau \right\} \tag{11.23}$$

where $\mathbf{R} = \mathbf{R}^{\top} > 0$ and $\mathbf{Q} = \mathbf{Q}^{\top} \geq 0$ are two cost matrices. The $\mathbf{Q}$-matrix is defined as $\mathbf{Q} = \mathrm{diag}\{\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_3\}$ where the weights $\mathbf{Q}_1, \mathbf{Q}_2$ and $\mathbf{Q}_3$ put penalty on position/heading $\boldsymbol{\eta}_p$, velocity $\boldsymbol{\nu}$, and actuator dynamics $\boldsymbol{\tau}$, respectively. The optimal control law minimizing (11.23) is (see Section 7.2.1):

$$\boldsymbol{\tau}_{\mathrm{LQ}} = \underbrace{-\mathbf{R}^{-1}\mathbf{B}^{\top}\mathbf{P}_{\infty}}_{\mathbf{G}}\mathbf{x} \tag{11.24}$$

where $\mathbf{P}_{\infty}$ is the solution of the ARE:

$$\mathbf{P}_{\infty}\mathbf{A} + \mathbf{A}^{\top}\mathbf{P}_{\infty} - \mathbf{P}_{\infty}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{P}_{\infty} + \mathbf{Q} = 0 \tag{11.25}$$

Integral Action

In order to obtain zero steady-state errors in *surge*, *sway*, and *yaw*, we must include integral action in the control law. Integral action can be included by using state augmentation. Since we are considering three outputs (x, y, ψ) to be regulated to zero, we can augment no more than three integral states to the system. Define a new state variable:

$$\mathbf{z} = \int_0^t \mathbf{y}(\tau)d\tau \quad \Longrightarrow \quad \dot{\mathbf{z}} = \mathbf{y} \tag{11.26}$$

Here $\mathbf{y}$ is a subspace of $\mathbf{x}$ defined as:

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{11.27}$$

with:

$$\mathbf{C} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{11.28}$$

Next define an augmented model with state vector $\mathbf{x}_a = [\mathbf{z}^{\top}, \mathbf{x}^{\top}]^{\top}$ such that:

$$\dot{\mathbf{x}}_a = \mathbf{A}_a\mathbf{x}_a + \mathbf{B}_a\boldsymbol{\tau}_{\mathrm{com}} \tag{11.29}$$

where

$$\mathbf{A}_a = \begin{bmatrix} \mathbf{0} & \mathbf{C} \\ \mathbf{0} & \mathbf{A} \end{bmatrix}, \quad \mathbf{B}_a = \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix} \tag{11.30}$$

The performance index for the integral controller becomes:

$$J = \min_{\boldsymbol{\tau}_{\mathrm{LQ}}} \left\{ \frac{1}{2}\int_0^T \left(\mathbf{x}_a^{\top}\mathbf{Q}_a\mathbf{x}_a + \boldsymbol{\tau}_{\mathrm{LQ}}^{\top}\mathbf{R}\boldsymbol{\tau}_{\mathrm{LQ}}\right) d\tau \right\} \tag{11.31}$$

where $\mathbf{R} = \mathbf{R}^{\top} > 0$ and:

$$\mathbf{Q}_a = \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_I \end{bmatrix} \geq \mathbf{0} \tag{11.32}$$

The matrix $\mathbf{Q}_I = \mathbf{Q}_I > 0$ is used to specify the integral times in *surge*, *sway*, and *yaw*. The optimal controller is (see Section 7.2.1):

$$\boldsymbol{\tau}_{\mathrm{LQ}} = \mathbf{G}_a\mathbf{x}_a = \mathbf{G}\mathbf{x} + \mathbf{G}_I \underbrace{\int_0^t \mathbf{y}(\tau)d\tau}_{\mathbf{z}} \tag{11.33}$$

where $\mathbf{G}_a = [\mathbf{G}_I, \mathbf{G}]$ and:

$$\mathbf{G}_a = -\mathbf{R}^{-1}\mathbf{B}_a^{\top}\mathbf{P}_{\infty} \tag{11.34}$$

$$\mathbf{P}_{\infty}\mathbf{A}_a + \mathbf{A}_a^{\top}\mathbf{P}_{\infty} - \mathbf{P}_{\infty}\mathbf{B}_a\mathbf{R}^{-1}\mathbf{B}_a^{\top}\mathbf{P}_{\infty} + \mathbf{Q}_a = \mathbf{0} \tag{11.35}$$

Wind Feedforward

It is straightforward to include wind feedforward τ_{FF} in the optimal controller. However, this requires that the wind forces and moment are known as functions of the wind speed and direction, as well as ship hull parameters. Models for this are presented in Section 8.4.4.

The Isherwood (1972) wind forces and moments are given by (4.10)–(4.12) implying that:

$$\tau_{\mathrm{FF}} = \frac{1}{2}\begin{bmatrix} C_X(\gamma_r)\rho_a V_r^2 A_T \\ C_Y(\gamma_r)\rho_a V_r^2 A_L \\ C_N(\gamma_r)\rho_a V_r^2 A_L L \end{bmatrix} \tag{11.36}$$

where C_X, C_Y, and C_N are wind drag and moment coefficients, ρ_a is the density of air, A_T and A_L are the transverse and lateral projected area, and L is the length of the ship. The *relative* wind speed and direction are:

$$V_r = V_w - \sqrt{u^2 + v^2} \tag{11.37}$$

$$\gamma_r = \beta_w - (\psi + \psi_w) \tag{11.38}$$

where V_w is the wind speed and β_w is the wind direction.

LQG Control - Linear Separation Principle

In practise only some of the states are measured. Hence, the optimal integral controller (11.33) should be replaced by:

$$\tau_{\mathrm{LQ}} = \mathbf{G}\hat{\mathbf{x}} + \mathbf{G}_I \mathbf{C} \int_0^t \hat{\mathbf{x}}(\tau) d\tau \tag{11.39}$$

where the state estimate $\hat{\mathbf{x}}$ is computed using a:

- Kalman filter (Section 6.2.3)
- nonlinear passive observer (Section 6.2.4)

For the Kalman filter in cascade with the LQ controller there exists a *linear separation principle* guaranteeing that $\hat{\mathbf{x}} \to \mathbf{x}$ and that $\mathbf{x} \to \mathbf{0}$ (Athans and Falb 1966). This is referred to as LQG control and it was first applied to design DP systems by.Balchen *et al.* (1976, 1980a, 1980b), and Grimble *et al.* (1980a, 1980b).

If the passive observer is applied the *nonlinear separation principle* of Section 11.2.4 can be used to prove stability of the observer-controller. This holds for both a LQ and a PD controller in cascade with the passive observer of Section 6.2.4.

11.2.3 Nonlinear PID Control

A nonlinear MIMO PID controller can be designed for station-keeping using the nonlinear models (11.1)–(11.2). Set-point regulation $\boldsymbol{\eta} = \boldsymbol{\nu} = \mathbf{0}$ is obtained by choosing (see Sections 7.1.3–7.1.4):

$$\boldsymbol{\tau} = -\mathbf{H}_m(s)\dot{\boldsymbol{\nu}} - \mathbf{R}^\top(\psi)\boldsymbol{\tau}_{\text{PID}} \tag{11.40}$$

where:

$$\boldsymbol{\tau}_{\text{PID}} = \mathbf{K}_p\boldsymbol{\eta} + \mathbf{K}_d\boldsymbol{\nu} + \mathbf{K}_i\int_0^t \boldsymbol{\eta}(\tau)d\tau \tag{11.41}$$

and $\mathbf{H}_m(s)$ is an optional transfer matrix for frequency dependent acceleration feedback (Fossen *et al.* 2002). Acceleration feedback for marine craft is discussed in more detail in Section 7.1.5 and by Lindegaard (2003).

For the case $\mathbf{H}_m$ = constant and $\mathbf{K}_i = \mathbf{0}$ (PD-control with acceleration feedback) global asymptotic stability (GAS) of the equilibrium point $\boldsymbol{\nu} = \boldsymbol{\eta} = \mathbf{0}$ follows directly from the CLF:

$$V = \underbrace{\frac{1}{2}\boldsymbol{\nu}^\top(\mathbf{M}+\mathbf{H}_m)\boldsymbol{\nu}}_{\substack{\text{kinetic}\\ \text{energy}}} + \underbrace{\frac{1}{2}\boldsymbol{\eta}^\top\mathbf{K}_p\boldsymbol{\eta}}_{\substack{\text{potential}\\ \text{energy}}} \tag{11.42}$$

$$\dot{V} = -\boldsymbol{\nu}^\top\left[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d\right]\boldsymbol{\nu} \tag{11.43}$$

using Krasovskii-LaSalle's theorem. In the case $\mathbf{K}_i > 0$ only local asymptotic stability can be proven. The nonlinear PID controller can be implemented together with the passive observer in Section 6.2.4 or the extended Kalman filter in Section 6.2.3. In Section 11.2.4 a nonlinear separation principle for cascaded PD-control and observer design is presented.

11.2.4 Nonlinear Separation Principle for PD-Control/Observer Design

For the ship control systems presented so far, slowly-varying environmental disturbances have been compensated for by adding integral action in the controller. In this section it is shown how a globally converging observer and a simple PD-type control law plus a nonlinear term of observer bias estimates can be combined to compensate for slowly-varying environmental disturbances (Loria *et al.* 2000). Moreover, the integral term is removed in the controller and replaced by a bias estimate. The stability proof is based on the *separation principle,* which holds for nonlinear systems. The separation principle is theoretically supported by recent results on cascaded nonlinear systems and standard Lyapunov theory, and it is validated in practice by experimentation with a model ship.

The observer-controller is designed in three steps:

1. design a UGAS state estimator
2. design the control law as if the whole state $\mathbf{x}$ and bias term $\mathbf{b}$ were known (measured) and free of noise
3. implement the control law with the observer estimates $\hat{\mathbf{x}}$ and $\hat{\mathbf{b}}$ and show that the observer-controller error dynamics is GAS

The stability proof of this approach requires that the separation principle hold for nonlinear systems. The method in this section relies on recent Lyapunov theorems on stability of cascaded time-varying systems to prove GAS (Panteley and Loria 1998).

Cascaded Systems

The controller-observer is analyzed by using cascaded system theory. Consider the time-varying systems Σ_1 and Σ_2 (Loria *et al.* 2000):

$$\Sigma_1 \quad : \quad \dot{\mathbf{x}}_1 = \mathbf{f}_1(t, \mathbf{x}_1) + \mathbf{G}(t, \mathbf{x})\mathbf{x}_2 \tag{11.44}$$

$$\Sigma_2 \quad : \quad \dot{\mathbf{x}}_2 = \mathbf{f}_2(t, \mathbf{x}_2) \tag{11.45}$$

where $\mathbf{x}_1 \in \mathbb{R}^n, \mathbf{x}_1 \in \mathbb{R}^m$, and $\mathbf{x} = [\mathbf{x}_1^\top, \mathbf{x}_2^\top]^\top$. The function $\mathbf{f}_1(t, \mathbf{x}_1)$ is continuously differentiable in $(t, \mathbf{x}_1)$, while $\mathbf{f}_2(t, \mathbf{x}_2)$ and $\mathbf{G}(t, \mathbf{x})$ are continuous in their arguments, and locally Lipschitz. The two subsystems Σ_1 and Σ_2 will represent the controller and observer error dynamics, respectively, while $\mathbf{G}(t, \mathbf{x})\mathbf{x}_2$ is the interaction term coupling these two subsystems together. A growth rate condition on $\mathbf{G}(t, \mathbf{x})$ is needed in order to prevent the controller error dynamics Σ_1 from becoming unstable when the true states are replaced with observer estimates.

The cascaded system (11.44)–(11.45) can be proven to be UGAS by reformulating Theorems 1 and 2 in Panteley and Loria (1998) according to:

Theorem 11.1 (UGAS for Cascaded Systems)
The cascaded system (11.44)–(11.45) is UGAS if Assumptions A1–A3 are satisfied:

A1) The system

$$\dot{\mathbf{x}}_1 = \mathbf{f}_1(t, \mathbf{x}_1) \tag{11.46}$$

is UGAS with a Lyapunov function $V = (t, \mathbf{x}_1)$, $V : \mathbb{R}_{\geq 0} \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, *positive definite–i.e.,* $V(t, \mathbf{0}) = \mathbf{0}$ *and* $V(t, \mathbf{x}_1) > 0$ *for all* $x_1 \neq \mathbf{0}$, *and proper (radially unbounded) which satisfies:*

$$\left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \|\mathbf{x}_1\| \leq c_1 V(t, \mathbf{x}_1) \qquad \forall \|\mathbf{x}_1\| \geq \mu \tag{11.47}$$

where c_1, $\mu > 0$. *It is also assumed that* $(\partial V / \partial \mathbf{x}_1)(t, \mathbf{x}_1)$ *is bounded uniformly in* t *for all* $\|\mathbf{x}_1\| < \mu$, *that is, there exists a constant* $c_2 > 0$ *such that for all* $t \geq t_0 \geq 0$:

$$\left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \leq c_2 \qquad \forall \|\mathbf{x}_1\| \leq \mu \tag{11.48}$$

A2) The function $\mathbf{G}(t, \mathbf{x})$ *satisfies:*

$$\|\mathbf{G}(t, \mathbf{x})\| \leq \theta_1 (\|\mathbf{x}_2\|) + \theta_2 (\|\mathbf{x}_2\|) \|\mathbf{x}_1\| \tag{11.49}$$

where $\theta_1, \theta_2 : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ *are continuous.*

A3) Equation $\dot{\mathbf{x}}_2 = \mathbf{f}_2(t, \mathbf{x}_2)$ *is UGAS, and for all* $t_0 \geq 0$:

$$\int_{t_0}^{\infty} \|\mathbf{x}_2(t))\| \, dt \leq \phi (\|\mathbf{x}_2(t_0)\|) \tag{11.50}$$

where the function $\phi(\cdot)$ *is a class* $\mathcal{K}$*-function.*

Main Result

Consider the nonlinear DP model:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{11.51}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} = \boldsymbol{\tau} + \mathbf{R}^{\top}(\psi)\mathbf{b} \tag{11.52}$$

$$\mathbf{y} = \boldsymbol{\eta} + \boldsymbol{\eta}_w \tag{11.53}$$

where $\mathbf{b} \in \mathbb{R}^3$ is a bias term representing slowly-varying environmental disturbances and $\mathbf{y} \in \mathbb{R}^3$ represent the measurements. Instead of using integral action to compensate for $\mathbf{b}$, a PD-controller:

$$\boldsymbol{\tau} = -\mathbf{R}^{\top}(\psi)\mathbf{K}_p\mathbf{e} - \mathbf{K}_d\boldsymbol{\nu} - \mathbf{R}^{\top}(\psi)\mathbf{b}, \qquad \mathbf{e} = \boldsymbol{\eta} - \boldsymbol{\eta}_d \tag{11.54}$$

can be implemented under the assumption that $\mathbf{b}$ is known (perfect compensation) and $\dot{\boldsymbol{\eta}}_d = \mathbf{0}$. However, it is impossible to measure $\mathbf{b}$ so a state observer is needed. For this purpose the passive observer (6.105)–(6.109) in Section 6.2.4 can be used to generate estimates of $\boldsymbol{\eta}, \boldsymbol{\nu}$, and $\mathbf{b}$ and at the same time provide wave filtering. Application of a nonlinear separation principle implies that the controller can be implemented using the estimated states $\hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}$, and $\hat{\mathbf{b}}$, that is:

$$\boldsymbol{\tau} = -\mathbf{R}^{\top}(\psi)\mathbf{K}_p\hat{\mathbf{e}} - \mathbf{K}_d\hat{\boldsymbol{\nu}} - \mathbf{R}^{\top}(\psi)\hat{\mathbf{b}}, \qquad \hat{\mathbf{e}} = \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_d \tag{11.55}$$

The proof needed to show that the passive observer with the controller (11.55) is UGAS is done in three steps corresponding to Assumptions A1–A3 in theorem 11.1.

Step 1: Observer Error Dynamics

Since the observer error dynamics:

$$\Sigma_2 : \dot{\mathbf{x}}_2 = \mathbf{f}_2(t, \mathbf{x}_2) \tag{11.56}$$

is UGES (and consequently UGAS) when considering the passive observer in Section 6.2.4, there exists positive constants λ_1 and λ_2 such that:

$$\|\mathbf{x}_2(t)\| \leq \lambda_1 \|\mathbf{x}_2(t_0)\| e^{-\lambda_2(t-t_0)} \tag{11.57}$$

and therefore Assumption A3 in Theorem 11.1 is satisfied with $\phi(\|\mathbf{x}_2(t_0)\|) = (\lambda_1/\lambda_2)\|\mathbf{x}_2(t_0)\|$

Step 2: Regulator Error Dynamics

The *full state* feedback controller (11.54) when applied to (11.51)–(11.52) gives:

$$\dot{\mathbf{e}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{11.58}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + (\mathbf{D} + \mathbf{K}_d)\boldsymbol{\nu} + \mathbf{R}^{\top}(\psi)\mathbf{K}_p\mathbf{e} = \mathbf{0} \tag{11.59}$$

This system is GAS according to LaSalle-Krasovskii's theorem since:

$$V = \frac{1}{2}\left(\boldsymbol{\nu}^\top \mathbf{M}\boldsymbol{\nu} + \mathbf{e}^\top \mathbf{K}_p \mathbf{e} + \mathbf{b}^\top \mathbf{T}\mathbf{b}\right) > 0, \qquad \forall \boldsymbol{\nu} \neq \mathbf{0}, \mathbf{e} \neq \mathbf{0}, \mathbf{b} \neq \mathbf{0} \tag{11.60}$$

and

$$\dot{V} = -\boldsymbol{\nu}^\top (\mathbf{D} + \mathbf{K}_d)\boldsymbol{\nu} - \mathbf{b}^\top \mathbf{b} \leq 0 \tag{11.61}$$

This analysis is done with $\dot{\mathbf{b}} = -\mathbf{T}^{-1}\mathbf{b}$ but is also possible to use $\dot{\mathbf{b}} = \mathbf{0}$. This implies that the system $\dot{\mathbf{x}}_1 = \mathbf{f}_1(\mathbf{x}_1, t)$, see Assumption A1 in Theorem 11.1, is UGAS. Next, a constant c_1 is easily found by considering:

$$\left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \|\mathbf{x}_1\| \leq \max\{m_M, k_M, 1\} \|\mathbf{x}_1\|^2 \qquad \forall \|\mathbf{x}_1\| \geq \mu \tag{11.62}$$

where $m_M = \lambda_{\max}(\mathbf{M})$ and $k_M = \lambda_{\max}(\mathbf{K}_p)$. Hence, (11.47) is satisfied with:

$$c_1 = \frac{\max\{m_M, k_M, 1\}}{\min\{m_m, k_m, 1\}} \tag{11.63}$$

where $m_m = \lambda_{\min}(\mathbf{M})$ and $k_m = \lambda_{\min}(\mathbf{K}_p)$. Also from (11.62) it is clear that (11.48) is satisfied by:

$$c_2 = \max\{m_M, k_M, 1\}\mu \tag{11.64}$$

Step 3: Growth Rate Condition

Finally, it can be shown that the growth rate condition (11.49) on $\mathbf{x}_1$ is satisfied by choosing $\theta_1 =$ constant and $\theta_2 = 0$ such that:

$$\|\mathbf{G}(t, \mathbf{x})\| \leq \theta_1 (\|\mathbf{x}_2\|) \tag{11.65}$$

The details in this analysis is found in Loria *et al.* (2000).

Experimental Results

The nonlinear controller (11.55) and passive observer of Section 6.2.4 have been tested experimentally using CyberShip I in the GNC Laboratory at NTNU. In this experiment wind disturbances were generated using a fan and a simple wave maker. More details regarding the experiment are found in Loria *et al.* (2000).

In the experiments the desired position and heading of the ship during DP were chosen as:

$$x_d = 208 \text{ (m)} \tag{11.66}$$
$$y_d = 334 \text{ (m)} \tag{11.67}$$
$$\psi_d = -150 \text{ (deg)} \tag{11.68}$$

The experiment was carried out for a ship scale 1:70, but the results have been transformed to full scale. The development of the experiment is as follows:

1. During the first 350 seconds there are no environmental loads perturbing the ship.

   Comments: From Figure 11.7 it is seen that the bias estimate $\hat{b}$ and the WF estimate $\hat{\eta}_w$ both are approximately zero as expected the first 350 seconds. The non-zero values of $\hat{b}$ are due to the water motion generated by the propellers. It is also seen that the regulation and estimation errors are very small during this phase; see upper plots in Figures 11.6 and 11.8.

2. After 350 seconds wind loads are generated by using a ducted fan directed approximately 30 degrees off the port side bow of the ship.

   Comments: When turned on, the fan produces a step input disturbance to the system, notice the peaks in Figures 11.6 and 11.8. This step is an unrealistic situation (in full-scale applications, no abrupt changes in the bias occur) however, it can be generated in the laboratory to show the performance of our observer-based controller. The bias estimates from the observer, $\hat{b}$, are integrated to cancel the off-sets resulting from the change in magnitude of the wind force, see 11.7. These estimated values are used in the output feedback control law to obtain perfect regulation, which validates the separation principle, see Figure 11.8. Most of the wind disturbance is compensated by the control input, and therefore the regulation errors converge to zero in $100 - 150$ seconds; see the first three plots of Figure 11.6. However, since the wind disturbance is a step, the observer needs some time for the bias estimate to converge to its true value, after which the controller compensates for the bias, hence keeping the boat almost still.

3. After 800 seconds the wave generator is turned on:

   Comments: This results in an oscillatory wave frequency motion η_w which builds up over time. The estimated wave frequency motion $\hat{\eta}_w$ is shown in the upper plots of Figure 11.7. Their effect in the position measurements is shown in the upper plots of Figure 11.6. In order to avoid $\hat{\eta}_w$ entering the feedback loop, this signal is filtered out from the position measurement. This wave filtering results in more smooth controls; see bottom plots of Figure 11.7. The low frequency estimates are clearly shown in the upper plots of Figure 11.6.

4. After 1700 seconds both the wind and wave generators are turned off.

   Comments: Turning off the fan, produces a second step input disturbance while the wave-induced motion decays more slowly. We see from Figure 11.7 that the bias estimates drop to approximately their initial values in $100 - 150$ seconds while the amplitudes of the WF motion estimates drop quite slowly. Again, almost perfect regulation to zero is obtained as soon as the bias estimates have converged to their true values. This clearly demonstrates the separation principle. In a full-scale implementation the wind force will build up quite slowly. Hence the step inputs do not constitute a problem.

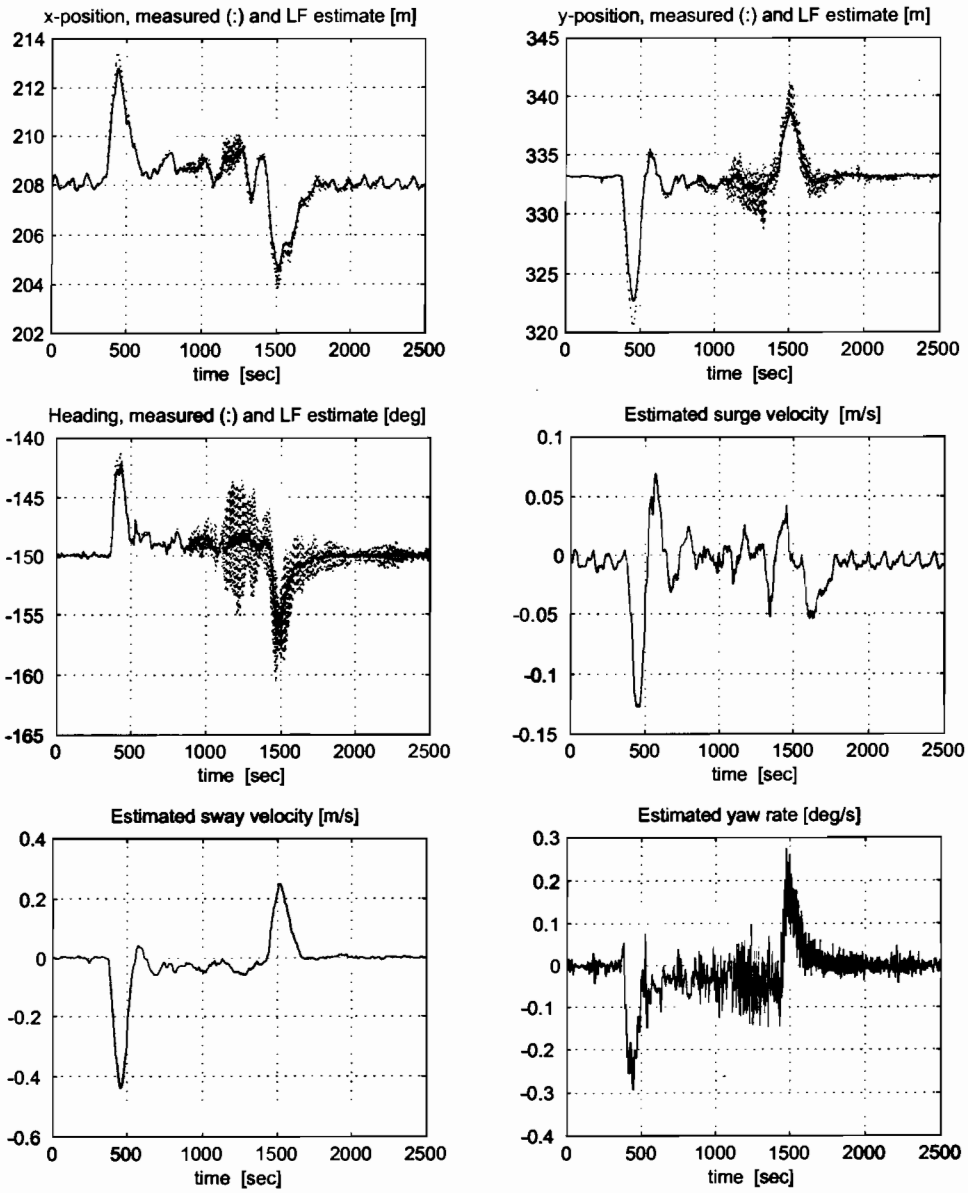


Figure 11.6: Plots 1–3 show the components of the measurement vector $\mathbf{y} = [x + x_w, y + y_w, \psi + \psi_w]$ and the wave filtered (LF-estimate) $\hat{\boldsymbol{\eta}} = [\hat{x}, \hat{y}, \hat{\psi}]^\top$ during DP. Plots 4–6 show the estimated LF velocity components $\hat{\boldsymbol{\nu}} = [\hat{u}, \hat{v}, \hat{r}]^\top$ versus time.

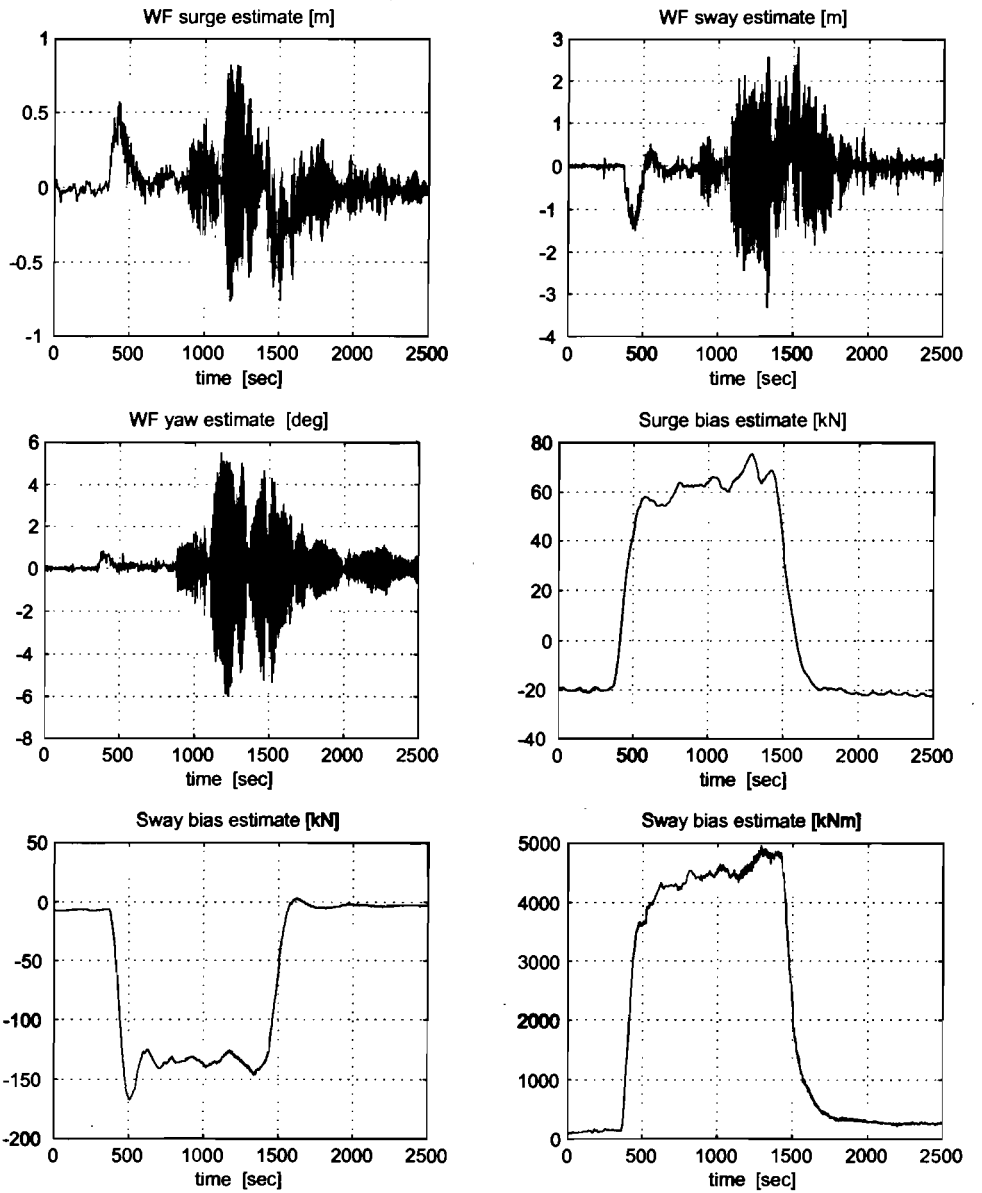


Figure 11.7: Plots 1–3 show the estimated wave frequency (WF) motion components $\hat{\boldsymbol{\eta}}_w = [\hat{x}_w, \hat{y}_w, \hat{\psi}_w]^\top$ while plots 4–6 show the bias estimates $\hat{\mathbf{b}} = [\hat{b}_1, \hat{b}_2, \hat{b}_3]^\top$ versus time.

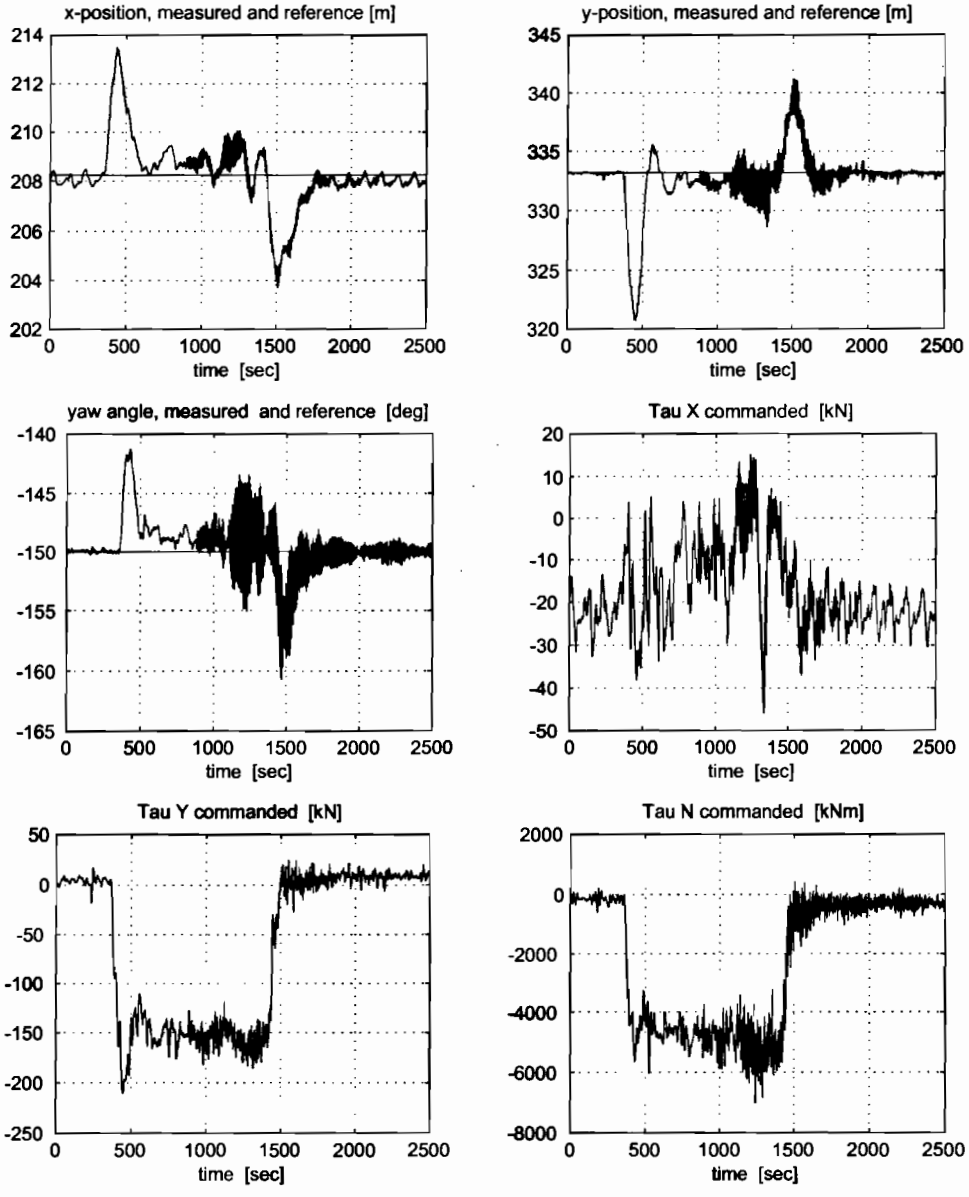


Figure 11.8: **Plots** 1–3 show the components of the measured position $\mathbf{y} = [x + x_w, y + y_w, \psi + \psi_w]^\top$ together with the desired position $\boldsymbol{\eta_d} = [x_d, y_d, \psi_d]^\top$ while plots 4–6 are the control inputs $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3]^\top$ versus time.

11.2.5 Nonlinear Observer Backstepping

This section presents a nonlinear DP and PM output feedback controller using vectorial backstepping (Fossen and Grøvlen 1998). Linearization of the kinematics is avoided by using a nonlinear state observer for velocity. Observer backstepping is used to provide nonlinear feedback from the state estimates (Krstic *et al.* 1995). The results of Krstic *et al.* (1995) are further improved by replacing the measured output with a filtered output when designing the feedback control law. Hence, the control inputs are generated by using filtered estimates of both the velocities and positions. Finally, GES is proven for the total system–i.e., is ship model, observer and control system. The control law was initially derived in component form by using 6 steps (3 for both the position and velocity components) when performing the backstepping (Grøvlen and Fossen 1996). Later the number of steps was reduced to 2 by using vectorial backstepping (Fossen and Grøvlen 1998).

Consider the DP and PM models (11.1)–(11.2) in the form:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{11.69}$$

$$\dot{\boldsymbol{\nu}} = \mathbf{A}_1\boldsymbol{\eta} + \mathbf{A}_2\boldsymbol{\nu} + \mathbf{B}\boldsymbol{\tau} \tag{11.70}$$

where

$$\mathbf{A}_1 = -\mathbf{M}^{-1}\mathbf{K}, \quad \mathbf{A}_2 = -\mathbf{M}^{-1}\mathbf{D}, \quad \mathbf{B} = \mathbf{M}^{-1} \tag{11.71}$$

It is assumed that only the North-East positions (n, e), and yaw angle ψ are measured, that is:

$$\mathbf{y} = \boldsymbol{\eta} = [n, e, \psi]^\top \tag{11.72}$$

Nonlinear Observer Design

The nonlinear observer is found by using Lyapunov theory, which places constraints on the choice of the filter gains. This is based on Fossen and Fjellstad (1995), where a nonlinear model-based observer for an underwater vehicle with filter gains that are functions of the measured attitude is proposed.

An observer for (11.69) and (11.70) is constructed as:

$$\dot{\hat{\boldsymbol{\eta}}} = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} + \mathbf{K}_1\tilde{\boldsymbol{\eta}} \tag{11.73}$$

$$\dot{\hat{\boldsymbol{\nu}}} = \mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{A}_2\hat{\boldsymbol{\nu}} + \mathbf{B}\boldsymbol{\tau} + \mathbf{K}_2\tilde{\boldsymbol{\eta}} \tag{11.74}$$

where $\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta} - \hat{\boldsymbol{\eta}}$ is the position estimation error. Defining $\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu} - \hat{\boldsymbol{\nu}}$, the error dynamics can be written:

$$\dot{\tilde{\boldsymbol{\eta}}} = \mathbf{R}(\psi)\tilde{\boldsymbol{\nu}} - \mathbf{K}_1\tilde{\boldsymbol{\eta}} \tag{11.75}$$

$$\dot{\tilde{\boldsymbol{\nu}}} = (\mathbf{A}_1 - \mathbf{K}_2)\tilde{\boldsymbol{\eta}} + \mathbf{A}_2\tilde{\boldsymbol{\nu}} \tag{11.76}$$

The matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ in (11.73) and (11.74) can be chosen such that the observer is GES. This is obtained by defining a Lyapunov function candidate:

$$V_{\text{obs}} = \frac{1}{2}\left(\tilde{\boldsymbol{\eta}}^\top\mathbf{P}_1\tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\nu}}^\top\mathbf{P}_2\tilde{\boldsymbol{\nu}}\right) > 0, \quad \forall\tilde{\boldsymbol{\eta}} \neq \mathbf{0}, \tilde{\boldsymbol{\nu}} \neq \mathbf{0} \tag{11.77}$$

where $\mathbf{P}_1 = \mathbf{P}_1^\top$ and $\mathbf{P}_2 = \mathbf{P}_2^\top$ are positive definite matrices. Hence:

$$\begin{aligned}\dot{V}_{\text{obs}} &= \dot{\tilde{\boldsymbol{\eta}}}^\top \mathbf{P}_1 \tilde{\boldsymbol{\eta}} + \frac{1}{2}\left(\dot{\tilde{\boldsymbol{\nu}}}^\top \mathbf{P}_2 \tilde{\boldsymbol{\nu}} + \tilde{\boldsymbol{\nu}}^\top \mathbf{P}_2 \dot{\tilde{\boldsymbol{\nu}}}\right) \\ &= \left(\mathbf{R}(\mathbf{y})\tilde{\boldsymbol{\nu}} - \mathbf{K}_1\tilde{\boldsymbol{\eta}}\right)^\top \mathbf{P}_1 \tilde{\boldsymbol{\eta}} \\ &\quad + \frac{1}{2}\tilde{\boldsymbol{\nu}}^\top \left(\mathbf{P}_2\mathbf{A}_2 + \mathbf{A}_2^\top \mathbf{P}_2\right)\tilde{\boldsymbol{\nu}} + \frac{1}{2}\tilde{\boldsymbol{\eta}}^\top \left(\mathbf{A}_1 - \mathbf{K}_2\right)^\top \mathbf{P}_2 \tilde{\boldsymbol{\nu}} + \frac{1}{2}\tilde{\boldsymbol{\nu}}^\top \mathbf{P}_2 \left(\mathbf{A}_1 - \mathbf{K}_2\right)\tilde{\boldsymbol{\eta}} \\ &= -\tilde{\boldsymbol{\eta}}^\top \mathbf{K}_1^\top \mathbf{P}_1 \tilde{\boldsymbol{\eta}} + \frac{1}{2}\tilde{\boldsymbol{\nu}}^\top \left(\mathbf{P}_2\mathbf{A}_2 + \mathbf{A}_2^\top \mathbf{P}_2\right)\tilde{\boldsymbol{\nu}} + \tilde{\boldsymbol{\nu}}^\top \left(\mathbf{R}^\top(\psi)\mathbf{P}_1 + \mathbf{P}_2\left(\mathbf{A}_1 - \mathbf{K}_2\right)\right)\tilde{\boldsymbol{\eta}}\end{aligned}$$

V_{obs} can be made negative definite by defining:

$$\mathbf{R}^\top(\psi)\mathbf{P}_1 + \mathbf{P}_2\left(\mathbf{A}_1 - \mathbf{K}_2\right) \;:=\; \mathbf{0} \tag{11.78}$$

$$\mathbf{K}_1^\top \mathbf{P}_1 \;:=\; \mathbf{Q}_1 \tag{11.79}$$

$$\frac{1}{2}\left(\mathbf{P}_2\mathbf{A}_2 + \mathbf{A}_2^\top \mathbf{P}_2\right) \;:=\; -\mathbf{Q}_2 \tag{11.80}$$

where $\mathbf{Q}_1 = \mathbf{Q}_1^\top$ and $\mathbf{Q}_2 = \mathbf{Q}_2^\top$ are positive definite design matrices and $\mathbf{A}_2 = -\mathbf{M}^{-1}\mathbf{D}$ is assumed to be *Hurwitz*. This implies that the ship must be course-stable. An extension to course-unstable ships can, however, be made by applying the approach of Robertson and Johansson (1998). Hence:

$$\dot{V}_{\text{obs}} = -\tilde{\boldsymbol{\eta}}^\top \mathbf{Q}_1 \tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top \mathbf{Q}_2 \tilde{\boldsymbol{\nu}} < 0, \quad \forall \tilde{\boldsymbol{\eta}} \neq \mathbf{0}, \tilde{\boldsymbol{\nu}} \neq \mathbf{0} \tag{11.81}$$

GES of the observer can be proven by defining $\mathbf{x} = [\tilde{\boldsymbol{\eta}}^\top, \tilde{\boldsymbol{\nu}}^\top]^\top$ and:

$$\mathbf{P} = \text{diag}\{\mathbf{P}_1, \mathbf{P}_2\}, \qquad \mathbf{Q} = \text{diag}\{\mathbf{Q}_1, \mathbf{Q}_2\} \tag{11.82}$$

and noticing that:

$$\begin{aligned}V_{\text{obs}} &= \frac{1}{2}\left(\tilde{\boldsymbol{\eta}}^\top \mathbf{P}_1 \tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\nu}}^\top \mathbf{P}_2 \tilde{\boldsymbol{\nu}}\right) \\ &= \frac{1}{2}\mathbf{x}^\top \mathbf{P}\mathbf{x} \leq \frac{\lambda_{\max}(\mathbf{P})}{2}\mathbf{x}^\top\mathbf{x}\end{aligned} \tag{11.83}$$

$$\begin{aligned}\dot{V}_{\text{obs}} &= -\tilde{\boldsymbol{\eta}}^\top \mathbf{Q}_1 \tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top \mathbf{Q}_2 \tilde{\boldsymbol{\nu}} \\ &= -\mathbf{x}^\top \mathbf{Q}\mathbf{x} \leq -\lambda_{\min}(\mathbf{Q})\mathbf{x}^\top\mathbf{x}\end{aligned} \tag{11.84}$$

where $\lambda_{\min}$ and $\lambda_{\max}$ denote the minimum and maximum eigenvalue, respectively. Hence:

$$V_{\text{obs}}(t) \leq e^{-2\alpha t} V_{\text{obs}}(0) \tag{11.85}$$

where $\alpha = \lambda_{\min}(\mathbf{Q})/\lambda_{\max}(\mathbf{P}) > 0$ can be interpreted as the *convergence rate*. The definitions (11.78) and (11.79) implies that:

$$\mathbf{K}_1 = \mathbf{P}_1^{-1}\mathbf{Q}_1 \tag{11.86}$$

$$\mathbf{K}_2(\psi) = \mathbf{P}_2^{-1}\mathbf{R}^\top(\psi)\mathbf{P}_1 - \mathbf{A}_1 \tag{11.87}$$

where $\mathbf{P}_2$ is computed from the Lyapunov equation (11.80) and, $\mathbf{P}_1$, $\mathbf{Q}_1$, and $\mathbf{Q}_2$ are constant positive definite design matrices.

Observer Backstepping

The observer backstepping problem is solved in two steps by using vectorial backstepping. The control objective is tracking of a smooth reference trajectory $\boldsymbol{\eta}_d = [n_d, e_d, \psi_d]^\top$ satisfying:

$$\ddot{\boldsymbol{\eta}}_d, \dot{\boldsymbol{\eta}}_d, \boldsymbol{\eta}_d \in \mathcal{L}_\infty \tag{11.88}$$

Station-keeping implies that $\boldsymbol{\eta}_d$ = constant.

Step 1:
Since $\boldsymbol{\eta}$ is measured with sensor noise, the tracking error, $\boldsymbol{\eta} - \boldsymbol{\eta}_d$, is rewritten in terms of the estimate of $\boldsymbol{\eta}$. Moreover, the tracking error, $\hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_d$, is used for observer backstepping since the observer guarantees that $\hat{\boldsymbol{\eta}} \to \boldsymbol{\eta}$. Hence, the resulting control law will use feedback from the filtered measurements $\hat{\boldsymbol{\eta}}$ instead of $\boldsymbol{\eta}$. Defining the error variable:

$$\mathbf{z}_1 = \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_d \tag{11.89}$$

implies that:

$$\dot{\mathbf{z}}_1 = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} + \mathbf{K}_1\tilde{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d \tag{11.90}$$

The main idea of backstepping is to choose one of the state variables as virtual control. It turns out that:

$$\boldsymbol{\xi}_1 = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} := \mathbf{z}_2 + \boldsymbol{\alpha}_1 \tag{11.91}$$

is an excellent choice for the virtual control. Notice that the virtual control $\boldsymbol{\xi}_1$ is defined as the sum of the next error variable $\mathbf{z}_2$ and $\boldsymbol{\alpha}_1$ which can be interpreted as a stabilizing function. Hence:

$$\dot{\mathbf{z}}_1 = \mathbf{z}_2 + \boldsymbol{\alpha}_1 + \mathbf{K}_1\tilde{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}_d \tag{11.92}$$

The stabilizing function is chosen as:

$$\boldsymbol{\alpha}_1 = -\mathbf{C}_1\mathbf{z}_1 - \mathbf{D}_1\mathbf{z}_1 + \dot{\boldsymbol{\eta}}_d \tag{11.93}$$

where $\mathbf{C}_1$ is a strictly positive constant feedback design matrix usually chosen to be diagonal and $\mathbf{D}_1$ is a positive diagonal damping matrix defined according to:

$$\mathbf{D}_1 = \begin{bmatrix} d_1\mathbf{k}_1^T\mathbf{k}_1 & 0 & 0 \\ 0 & d_2\mathbf{k}_2^T\mathbf{k}_2 & 0 \\ 0 & 0 & d_3\mathbf{k}_3^T\mathbf{k}_3 \end{bmatrix} \tag{11.94}$$

where $d_i > 0$ $(i = 1, \ldots, 3)$ and $\mathbf{k}_i$ $(i = 1, \ldots, 3)$ are the column vectors of:

$$\mathbf{K}_1^\top = [\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_3] \tag{11.95}$$

The motivation for the damping term $-\mathbf{D}_1\mathbf{z}_1$ is that $\mathbf{K}_1\tilde{\boldsymbol{\eta}}$ can be treated as a disturbance term in (11.90) to be compensated for by adding damping. The final equation for $\dot{\mathbf{z}}_1$ is then:

$$\dot{\mathbf{z}}_1 = -(\mathbf{C}_1 + \mathbf{D}_1)\mathbf{z}_1 + \mathbf{z}_2 + \mathbf{K}_1\tilde{\boldsymbol{\eta}} \tag{11.96}$$

Notice that $\mathbf{D}_1$ is only used to compensate for the "disturbance" term $\mathbf{K}_1\tilde{\boldsymbol{\eta}}$. In the next step, the desired dynamics of $\mathbf{z}_2$ will be specified.

Step 2:
Time differentiation of (11.91) yields:

$$\dot{\mathbf{z}}_2 = \dot{\boldsymbol{\xi}}_1 - \dot{\boldsymbol{\alpha}}_1 \tag{11.97}$$

which can be written according to:

$$\begin{aligned} \dot{\mathbf{z}}_2 &= \mathbf{R}(\psi)\dot{\hat{\boldsymbol{\nu}}} + \dot{\mathbf{R}}(\psi)\hat{\boldsymbol{\nu}} + \mathbf{C}_1\dot{\mathbf{z}}_1 + \mathbf{D}_1\dot{\mathbf{z}}_1 - \ddot{\boldsymbol{\eta}}_d \\ &\Updownarrow \\ \dot{\mathbf{z}}_2 &= -(\mathbf{C}_1 + \mathbf{D}_1)^2\mathbf{z}_1 \\ &\quad +(\mathbf{C}_1 + \mathbf{D}_1)(\mathbf{z}_2 + \mathbf{K}_1\tilde{\boldsymbol{\eta}}) - \ddot{\boldsymbol{\eta}}_d + \dot{\mathbf{R}}(\psi)\hat{\boldsymbol{\nu}} \\ &\quad +\mathbf{R}(\psi)(\mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{A}_2\hat{\boldsymbol{\nu}} + \mathbf{B}\boldsymbol{\tau} + \mathbf{K}_2\tilde{\boldsymbol{\eta}}) \end{aligned} \tag{11.98}$$

Defining:

$$\boldsymbol{\rho} = \begin{bmatrix} 0 \\ 0 \\ r \end{bmatrix}, \qquad \mathbf{S}(\boldsymbol{\rho}) = \begin{bmatrix} 0 & -r & 0 \\ r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.99}$$

and $\tilde{\boldsymbol{\rho}} = \boldsymbol{\rho} - \hat{\boldsymbol{\rho}}$. Hence the time derivative of $\mathbf{J}(\mathbf{y})$ can be written:

$$\dot{\mathbf{J}}(\mathbf{y}) = \mathbf{R}(\psi)\mathbf{S}(\boldsymbol{\rho}) = \mathbf{R}(\psi)\mathbf{S}(\tilde{\boldsymbol{\rho}}) + \mathbf{R}(\psi)\mathbf{S}(\hat{\boldsymbol{\rho}}) \tag{11.100}$$

Hence the product $\dot{\mathbf{R}}(\psi)\hat{\boldsymbol{\nu}}$ in (11.98) can be rewritten as:

$$\begin{aligned} \dot{\mathbf{R}}(\psi)\hat{\boldsymbol{\nu}} &= \mathbf{R}(\psi)\mathbf{S}(\tilde{\boldsymbol{\rho}})\hat{\boldsymbol{\nu}} + \mathbf{R}(\psi)\mathbf{S}(\hat{\boldsymbol{\rho}})\hat{\boldsymbol{\nu}} \\ &= \mathbf{R}(\psi)\mathbf{T}(\hat{\boldsymbol{\nu}})\tilde{\boldsymbol{\nu}} + \mathbf{R}(\psi)\mathbf{S}(\hat{\boldsymbol{\rho}})\hat{\boldsymbol{\nu}} \end{aligned} \tag{11.101}$$

where

$$\mathbf{T}(\hat{\boldsymbol{\nu}}) = \begin{bmatrix} 0 & 0 & -\hat{v} \\ 0 & 0 & \hat{u} \\ 0 & 0 & 0 \end{bmatrix} \tag{11.102}$$

Substituting (11.101) into (11.98) yields:

$$\begin{aligned} \dot{\mathbf{z}}_2 &= -(\mathbf{C}_1 + \mathbf{D}_1)^2\mathbf{z}_1 + (\mathbf{C}_1 + \mathbf{D}_1)(\mathbf{z}_2 + \mathbf{K}_1\tilde{\boldsymbol{\eta}}) - \ddot{\boldsymbol{\eta}}_d \\ &\quad +\mathbf{R}(\psi)(\mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{A}_2\hat{\boldsymbol{\nu}} + \mathbf{B}\boldsymbol{\tau} + \mathbf{K}_2\tilde{\boldsymbol{\eta}}) + \mathbf{R}(\psi)\mathbf{T}(\hat{\boldsymbol{\nu}})\tilde{\boldsymbol{\nu}} + \mathbf{R}(\psi)\mathbf{S}(\hat{\boldsymbol{\rho}})\hat{\boldsymbol{\nu}} \end{aligned} \tag{11.103}$$

Collecting terms in $\tilde{\boldsymbol{\eta}}$ and $\tilde{\boldsymbol{\nu}}$,yields:

$$\begin{aligned} \dot{\mathbf{z}}_2 &= ((\mathbf{C}_1 + \mathbf{D}_1)\mathbf{K}_1 + \mathbf{R}(\psi)\mathbf{K}_2)\,\tilde{\boldsymbol{\eta}} \\ &\quad +\mathbf{R}(\psi)\mathbf{T}(\hat{\boldsymbol{\nu}})\tilde{\boldsymbol{\nu}} + \boldsymbol{\varphi}(\ddot{\boldsymbol{\eta}}_d, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\rho}}, \mathbf{y}) + \mathbf{R}(\psi)\mathbf{B}\boldsymbol{\tau} \end{aligned} \tag{11.104}$$

where:

$$\begin{aligned} \boldsymbol{\varphi}(\ddot{\boldsymbol{\eta}}_d, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\rho}}, \mathbf{y}) &= -(\mathbf{C}_1 + \mathbf{D}_1)^2\mathbf{z}_1 + (\mathbf{C}_1 + \mathbf{D}_1)\mathbf{z}_2 - \ddot{\boldsymbol{\eta}}_d \\ &\quad +\mathbf{R}(\psi)\mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{R}(\psi)(\mathbf{A}_2 + \mathbf{S}(\hat{\boldsymbol{\rho}}))\hat{\boldsymbol{\nu}} \end{aligned} \tag{11.105}$$

The following choice of feedback is made:

$$\boldsymbol{\tau} = -\mathbf{B}^{-1}\mathbf{R}^{\top}(\psi)\left(\boldsymbol{\varphi}(\ddot{\boldsymbol{\eta}}_d, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\rho}}, \mathbf{y}) + \mathbf{C}_2\mathbf{z}_2 + \mathbf{D}_2\mathbf{z}_2 + \mathbf{z}_1\right) \tag{11.106}$$

where $\mathbf{C}_2$ is a strictly positive feedback design matrix usually chosen to be diagonal. The resulting $\mathbf{z}_2$-error dynamics is:

$$\dot{\mathbf{z}}_2 = -\mathbf{C}_2\mathbf{z}_2 - \mathbf{D}_2\mathbf{z}_2 - \mathbf{z}_1 + \boldsymbol{\Omega}_1\tilde{\boldsymbol{\eta}} + \boldsymbol{\Omega}_2\tilde{\boldsymbol{\nu}} \tag{11.107}$$

where

$$\boldsymbol{\Omega}_1 \;:\; = (\mathbf{C}_1 + \mathbf{D}_1)\mathbf{K}_1 + \mathbf{R}(\psi)\mathbf{K}_2 \tag{11.108}$$

$$\boldsymbol{\Omega}_2 \;:\; = \mathbf{R}(\psi)\mathbf{T}(\tilde{\nu}) \tag{11.109}$$

The matrix $\mathbf{D}_2$ is defined in terms of the elements of $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$, that is:

$$\boldsymbol{\Omega}_1^\top = [\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \boldsymbol{\omega}_3], \qquad \boldsymbol{\Omega}_2^\top = [\boldsymbol{\omega}_4, \boldsymbol{\omega}_5, \mathbf{0}] \tag{11.110}$$

with:

$$\mathbf{D}_2 = \text{diag}\{d_4(\boldsymbol{\omega}_1^\top\boldsymbol{\omega}_1 + \boldsymbol{\omega}_4^\top\boldsymbol{\omega}_4), d_5(\boldsymbol{\omega}_2^\top\boldsymbol{\omega}_2 + \boldsymbol{\omega}_5^\top\boldsymbol{\omega}_5), d_6\boldsymbol{\omega}_3^\top\boldsymbol{\omega}_3)\} \tag{11.111}$$

where $d_i > 0\,(i = 4, \ldots, 6)$.

Error Dynamics

The resulting error dynamics can be written:

$$\dot{\mathbf{z}} = -(\mathbf{C}_z + \mathbf{D}_z + \mathbf{E})\mathbf{z} + \mathbf{W}_1\tilde{\boldsymbol{\eta}} + \mathbf{W}_2\tilde{\boldsymbol{\nu}} \tag{11.112}$$

$$\dot{\tilde{\boldsymbol{\eta}}} = \mathbf{R}(\psi)\tilde{\boldsymbol{\nu}} - \mathbf{K}_1\tilde{\boldsymbol{\eta}} \tag{11.113}$$

$$\dot{\tilde{\boldsymbol{\nu}}} = (\mathbf{A}_1 - \mathbf{K}_2)\tilde{\boldsymbol{\eta}} + \mathbf{A}_2\tilde{\boldsymbol{\nu}} \tag{11.114}$$

where $\mathbf{z} = [\mathbf{z}_1^\top, \mathbf{z}_2^\top]^\top$ and:

$$\mathbf{C}_z = \begin{bmatrix} \mathbf{C}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 \end{bmatrix}, \quad \mathbf{D}_z = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{bmatrix} \tag{11.115}$$

$$\mathbf{E} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{bmatrix}, \quad \mathbf{W}_1 = \begin{bmatrix} \mathbf{K}_1 \\ \boldsymbol{\Omega}_1 \end{bmatrix}, \quad \mathbf{W}_2 = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\Omega}_2 \end{bmatrix} \tag{11.116}$$

Lyapunov Stability Analysis

A Lyapunov function candidate for the control law is:

$$V_{\text{con}} = \frac{1}{2}\mathbf{z}^\top\mathbf{z} > 0, \quad \forall \mathbf{z} \neq \mathbf{0} \tag{11.117}$$

Hence, a Lyapunov function candidate for both the control law and observer can be defined as:

$$V = V_{\text{con}} + V_{\text{obs}}$$

$$\Updownarrow \tag{11.118}$$

$$V = \frac{1}{2}\left(\mathbf{z}^\top\mathbf{z} + \tilde{\boldsymbol{\eta}}^\top\mathbf{P}_1\tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\nu}}^\top\mathbf{P}_2\tilde{\boldsymbol{\nu}}\right) \tag{11.119}$$

Time differentiation of V along the trajectories of $\mathbf{z}, \tilde{\boldsymbol{\eta}}$ and $\tilde{\boldsymbol{\nu}}$, yields:

$$\dot{V} = \mathbf{z}^\top \dot{\mathbf{z}} + \dot{V}_{\text{obs}} \tag{11.120}$$

where the expression for $\dot{V}_{obs}$ is given by (11.81). Consequently:

$$\dot{V} = \mathbf{z}^\top \dot{\mathbf{z}} - \tilde{\boldsymbol{\eta}}^\top \mathbf{Q}_1 \tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top \mathbf{Q}_2 \tilde{\boldsymbol{\nu}} \tag{11.121}$$

Substituting (11.112) into (11.121), and using the fact that $\mathbf{z}^\top \mathbf{E}\mathbf{z} = 0$, $\forall \mathbf{z}$, yields:

$$\dot{V} = \mathbf{z}^\top \left(-\mathbf{C}_z \mathbf{z} - \mathbf{D}_z \mathbf{z} + \mathbf{W}_1 \tilde{\boldsymbol{\eta}} + \mathbf{W}_2 \tilde{\boldsymbol{\nu}}\right) - \tilde{\boldsymbol{\eta}}^\top \mathbf{Q}_1 \tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top \mathbf{Q}_2 \tilde{\boldsymbol{\nu}} \tag{11.122}$$

Adding the zero terms:

$$\frac{1}{4}\left(\tilde{\boldsymbol{\eta}}^\top \mathbf{G}_1 \tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\eta}}^\top \mathbf{G}_1 \tilde{\boldsymbol{\eta}}\right) = 0 \tag{11.123}$$

$$\frac{1}{4}\left(\tilde{\boldsymbol{\nu}}^\top \mathbf{G}_2 \tilde{\boldsymbol{\nu}} - \tilde{\boldsymbol{\nu}}^\top \mathbf{G}_2 \tilde{\boldsymbol{\nu}}\right) = 0 \tag{11.124}$$

to (11.122) yields:

$$\begin{aligned} \dot{V} = {} & -\mathbf{z}^\top \mathbf{C}_z \mathbf{z} - \mathbf{z}^\top \mathbf{D}_z \mathbf{z} + \mathbf{z}^\top \mathbf{W}_1 \tilde{\boldsymbol{\eta}} + \mathbf{z}^\top \mathbf{W}_2 \tilde{\boldsymbol{\nu}} \\ & -\frac{1}{4}(\tilde{\boldsymbol{\eta}}^\top \mathbf{G}_1 \tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\nu}}^\top \mathbf{G}_2 \tilde{\boldsymbol{\nu}}) - \tilde{\boldsymbol{\eta}}^\top (\mathbf{Q}_1 - \frac{1}{4}\mathbf{G}_1)\tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top (\mathbf{Q}_2 - \frac{1}{4}\mathbf{G}_2)\tilde{\boldsymbol{\nu}} \end{aligned} \tag{11.125}$$

The matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ are defined as:

$$\mathbf{G}_1 := g_1 \mathbf{I}, \qquad \mathbf{G}_2 := g_2 \mathbf{I} \tag{11.126}$$

where

$$g_1 = \sum_{i=1}^{6} \frac{1}{d_i}, \qquad g_2 = \frac{1}{d_4} + \frac{1}{d_5} \tag{11.127}$$

In Appendix I in Fossen and Grøvlen (1998) it was shown that:

$$-\mathbf{z}^\top \mathbf{D}_z \mathbf{z} + \mathbf{z}^\top \mathbf{W}_1 \tilde{\boldsymbol{\eta}} + \mathbf{z}^\top \mathbf{W}_2 \tilde{\boldsymbol{\nu}} - \frac{1}{4}(\tilde{\boldsymbol{\eta}}^\top \mathbf{G}_1 \tilde{\boldsymbol{\eta}} + \tilde{\boldsymbol{\nu}}^\top \mathbf{G}_2 \tilde{\boldsymbol{\nu}}) \leq 0 \tag{11.128}$$

Hence:

$$\dot{V} \leq -\mathbf{z}^\top \mathbf{C}_z \mathbf{z} - \tilde{\boldsymbol{\eta}}^\top (\mathbf{Q}_1 - \frac{1}{4}\mathbf{G}_1)\tilde{\boldsymbol{\eta}} - \tilde{\boldsymbol{\nu}}^\top (\mathbf{Q}_2 - \frac{1}{4}\mathbf{G}_2)\tilde{\boldsymbol{\nu}} < 0, \quad \forall \tilde{\boldsymbol{\eta}} \neq \mathbf{0}, \tilde{\boldsymbol{\nu}} \neq \mathbf{0} \tag{11.129}$$

It is then clear that $\dot{V}$ can be made negative definite by choosing the positive definite weight matrices $\mathbf{Q}_1 = \mathbf{Q}_1^\top$ and $\mathbf{Q}_2 = \mathbf{Q}_2^\top$ such that $\|\mathbf{Q}_1\| > g_1/4$ and $\|\mathbf{Q}_2\| > g_2/4$. Notice that $\mathbf{G}_1$ and $\mathbf{G}_2$ are not needed for implementation.

Hence, according to Lyapunov stability theory, the ship model with control law (11.105)–(11.106) and observer (11.73)–(11.74) is GES.

Resulting Control Law

The resulting control law is given below.

Control law:

$$\boldsymbol{\tau} = -\mathbf{B}^{-1}\mathbf{R}^{\top}(\psi)(\boldsymbol{\varphi}(\ddot{\boldsymbol{\eta}}_d, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\rho}}, \mathbf{y}) + \mathbf{C}_2\mathbf{z}_2 + \mathbf{D}_2\mathbf{z}_2 + \mathbf{z}_1)$$

$$\boldsymbol{\varphi}(\ddot{\boldsymbol{\eta}}_d, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\rho}}, \mathbf{y}) = -\ddot{\boldsymbol{\eta}}_d - (\mathbf{C}_1 + \mathbf{D}_1)^2\mathbf{z}_1 + (\mathbf{C}_1 + \mathbf{D}_1)\mathbf{z}_2 + \mathbf{R}(\psi)\mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{J}(\mathbf{y})(\mathbf{A}_2 + \mathbf{S}(\hat{\boldsymbol{\rho}}))\hat{\boldsymbol{\nu}}$$

$$\mathbf{z}_1 = \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}_d, \qquad \mathbf{z}_2 = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} - \boldsymbol{\alpha}_1$$

Stabilizing function:

$$\boldsymbol{\alpha}_1 = -\mathbf{C}_1\mathbf{z}_1 - \mathbf{D}_1\mathbf{z}_1 + \dot{\boldsymbol{\eta}}_d$$

Observer:

$$\dot{\hat{\mathbf{n}}} = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} + \mathbf{K}_1\tilde{\boldsymbol{\eta}}$$

$$\dot{\hat{\boldsymbol{\nu}}} = \mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{A}_2\hat{\boldsymbol{\nu}} + \mathbf{B}\boldsymbol{\tau} + \mathbf{K}_2(\mathbf{y})\tilde{\boldsymbol{\eta}}$$

Observer gains:

$$\mathbf{K}_1 = \mathbf{P}_1^{-1}\mathbf{Q}_1$$

$$\mathbf{K}_2(\psi) = \mathbf{P}_2^{-1}\mathbf{R}^{\top}(\psi)\mathbf{P}_1 - \mathbf{A}_1$$

$$\tfrac{1}{2}\left(\mathbf{P}_2\mathbf{A}_2 + \mathbf{A}_2^{\top}\mathbf{P}_2\right) = -\mathbf{Q}_2$$

Nonlinear damping:

$$\boldsymbol{\Omega}_1 = (\mathbf{C}_1 + \mathbf{D}_1)\mathbf{K}_1 + \mathbf{R}(\psi)\mathbf{K}_2(\psi), \quad \boldsymbol{\Omega}_2 = \mathbf{R}(\psi)\mathbf{T}(\hat{\boldsymbol{\nu}})$$

$$\boldsymbol{\Omega}_1^{\top} = [\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \boldsymbol{\omega}_3], \quad \boldsymbol{\Omega}_2^{\top} = [\boldsymbol{\omega}_4, \boldsymbol{\omega}_5, \mathbf{0}]$$

$$\mathbf{D}_1 = \operatorname{diag}\{d_1\mathbf{k}_1^{\top}\mathbf{k}_1, d_2\mathbf{k}_2^{\top}\mathbf{k}_2, d_3\mathbf{k}_3^{\top}\mathbf{k}_3\}$$

$$\mathbf{D}_2 = \operatorname{diag}\{d_4(\boldsymbol{\omega}_1^{\top}\boldsymbol{\omega}_1 + \boldsymbol{\omega}_4^{\top}\boldsymbol{\omega}_4), d_5(\boldsymbol{\omega}_2^{\top}\boldsymbol{\omega}_2 + \boldsymbol{\omega}_5^{\top}\boldsymbol{\omega}_5), d_6\boldsymbol{\omega}_3^{\top}\boldsymbol{\omega}_3\}$$

$$\mathbf{K}_1^{\top} = [\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_3]$$

Alternative Analysis using Contraction Theory

In Jouffroy and Lottin (2002*b*) the observer-controller by Fossen and Grøvlen (1998) has been analyzed using contraction theory. Contraction theory is a recent tool that can be used to study the stability of nonlinear system trajectories with respect to each other. The original definition of contraction requires the *uniform negative definiteness* of the *Jacobian* of the system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \tag{11.130}$$

Contracting behavior is determined upon the exact differential relation, that is:

$$\delta\dot{\mathbf{x}} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x},t)\delta\mathbf{x} \tag{11.131}$$

where $\delta\mathbf{x}$ is a virtual displacement–i.e., an infinitesimal displacement at fixed time and $\partial\mathbf{f}/\partial\mathbf{x}$ is recognized as the *Jacobian* of the system. Define the local transformation:

$$\delta\mathbf{z} = \boldsymbol{\Theta}(\mathbf{x},t)\delta\mathbf{x} \tag{11.132}$$

where $\boldsymbol{\Theta}(\mathbf{x},t)$ is a coordinate transformation matrix. The *generalized Jacobian* of the $\delta\mathbf{z}$-system is:

$$\mathbf{F}(\mathbf{x},t) = \left(\dot{\boldsymbol{\Theta}}(\mathbf{x},t) + \boldsymbol{\Theta}(\mathbf{x},t)\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x},t)\right)\boldsymbol{\Theta}^{-1}(\mathbf{x},t) \tag{11.133}$$

The main definition and theorem of contraction are taken from Lohmiller and Slotine (1998).

Definition 11.1 (Contraction Region)
A region of the state space is called a contraction region with respect to a uniformly positive metric:

$$\mathbf{M}(\mathbf{x},t) = \boldsymbol{\Theta}^{\top}(\mathbf{x},t)\boldsymbol{\Theta}(\mathbf{x},t) \tag{11.134}$$

where $\boldsymbol{\Theta}(\mathbf{x},t)$ is a differential coordinate transformation matrix, if equivalently:

$$\mathbf{F}(\mathbf{x},t) = \left(\dot{\boldsymbol{\Theta}}(\mathbf{x},t) + \boldsymbol{\Theta}(\mathbf{x},t)\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x},t)\right)\boldsymbol{\Theta}^{-1}(\mathbf{x},t) \tag{11.135}$$

or:

$$\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x},t)\right)^{\top}\mathbf{M}(\mathbf{x},t) + \dot{\mathbf{M}}(\mathbf{x},t) + \mathbf{M}(\mathbf{x},t)\left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x},t)\right) \tag{11.136}$$

are uniformly negative definite.

Theorem 11.2 (Exponential Convergence in Contracting Systems)
Any trajectory, which starts in a ball of constant radius with respect to the metric $\mathbf{M}(\mathbf{x},t)$, centered at a given trajectory and contained at all times in a contraction region, remains in that ball and converges exponentially to this trajectory.

As a consequence of this, two systems contracting under possibly different metrics, and connected in feedback form as:

$$\begin{bmatrix} \delta\dot{\mathbf{z}}_1 \\ \delta\dot{\mathbf{z}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1 & \mathbf{G} \\ -\mathbf{G}^{\top} & \mathbf{F}_2 \end{bmatrix}\begin{bmatrix} \delta\mathbf{z}_1 \\ \delta\mathbf{z}_2 \end{bmatrix} \tag{11.137}$$

where $\delta\mathbf{z}_i = \boldsymbol{\Theta}(\mathbf{x}_i,t)\delta\mathbf{x}_i$ $(i = 1,2)$, will represent a new system that is contracting. This is recognized as the well known feedback interconnection structure which is used in passivity and backstepping analysis.

Again consider the nonlinear observer of Fossen and Grøvlen (1998) :

$$\dot{\hat{\boldsymbol{\eta}}} = \mathbf{R}(\psi)\hat{\boldsymbol{\nu}} + \mathbf{K}_1(\boldsymbol{\eta} - \hat{\boldsymbol{\eta}}) := \mathbf{f}_1(\hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\eta}}, \boldsymbol{\eta}) \tag{11.138}$$

$$\dot{\hat{\boldsymbol{\nu}}} = \mathbf{A}_1\hat{\boldsymbol{\eta}} + \mathbf{A}_2\hat{\boldsymbol{\nu}} + \mathbf{B}\boldsymbol{\tau} + \mathbf{K}_2(\boldsymbol{\eta} - \hat{\boldsymbol{\eta}}) := \mathbf{f}_2(\hat{\boldsymbol{\nu}}, \hat{\boldsymbol{\eta}}, \boldsymbol{\eta}, \boldsymbol{\tau}) \tag{11.139}$$

The incremental equations and the Jacobian of this system with respect to $(\hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\nu}})$ are:

$$\begin{bmatrix} \delta\dot{\mathbf{x}}_1 \\ \delta\dot{\mathbf{x}}_2 \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_1 & \mathbf{R}(\psi) \\ \mathbf{A}_1 - \mathbf{K}_2 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}_1 \\ \delta\mathbf{x}_2 \end{bmatrix} \tag{11.140}$$

This system is contracting, that is $\mathbf{F}^\top + \mathbf{F} > 0$, if:

$$\mathbf{K}_1 > 0 \tag{11.141}$$

$$\mathbf{K}_2 = \mathbf{A}_1 - \mathbf{R}^\top(\psi) \tag{11.142}$$

It is seen that this design is somewhat restrictive because only $\mathbf{K}_1 > 0$ can be tuned arbitrarily. To relax the condition on $\mathbf{K}_2$, define $\delta\mathbf{z}_i = \boldsymbol{\Theta}_i\delta\mathbf{x}_i$ $(i = 1, 2)$ such that:

$$\begin{bmatrix} \delta\dot{\mathbf{z}}_1 \\ \delta\dot{\mathbf{z}}_2 \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\Theta}_1\mathbf{K}_1\boldsymbol{\Theta}_1^{-1} & \boldsymbol{\Theta}_1\mathbf{R}(\psi)\boldsymbol{\Theta}_2^{-1} \\ \boldsymbol{\Theta}_2(\mathbf{A}_1 - \mathbf{K}_2)\boldsymbol{\Theta}_1^{-1} & \boldsymbol{\Theta}_2\mathbf{A}_2\boldsymbol{\Theta}_2^{-1} \end{bmatrix} \begin{bmatrix} \delta\mathbf{z}_1 \\ \delta\mathbf{z}_2 \end{bmatrix} \tag{11.143}$$

The conditions for this system to be contracting are:

$$-\boldsymbol{\Theta}_1\mathbf{K}_1\boldsymbol{\Theta}_1^{-1} < 0 \tag{11.144}$$

$$\boldsymbol{\Theta}_2\mathbf{A}_2\boldsymbol{\Theta}_2^{-1} < 0 \tag{11.145}$$

$$-\left(\boldsymbol{\Theta}_2(\mathbf{A}_1 - \mathbf{K}_2)\boldsymbol{\Theta}_1^{-1}\right)^\top = \boldsymbol{\Theta}_1\mathbf{R}(\psi)\boldsymbol{\Theta}_2^{-1} \tag{11.146}$$

Finally, choosing $\mathbf{P}_i = \boldsymbol{\Theta}_i^\top\boldsymbol{\Theta}_i$ $(i = 1, 2)$, yields:

$$\mathbf{K}_1^\top\mathbf{P}_1 + \mathbf{P}_1\mathbf{K}_1 > 0 \tag{11.147}$$

$$\mathbf{A}_2^\top\mathbf{P}_2 + \mathbf{P}_2\mathbf{A}_2 > 0 \tag{11.148}$$

$$\mathbf{K}_2(\psi) = \mathbf{P}_2^{-1}\mathbf{R}^\top(\psi)\mathbf{P}_1 + \mathbf{A}_1 \tag{11.149}$$

where $\mathbf{K}_1 > 0$. These conditions are recognized as the UGES conditions of the observer backstepping method. This shows that contracting theory yields a UGES stable error system similar to what is obtained when using backstepping. In addition, additional design freedom is obtained in that the metrics $\boldsymbol{\Theta}_1$ and $\boldsymbol{\Theta}_2$ can be specified rather arbitrarily. For the interested reader, backstepping designs in terms of contraction theory are analyzed in Jouffroy and Lottin (2002*a*).

Case Studies

Two case studies will be presented to illustrate the performance of the proposed controller. In the first one, thruster assisted PM of a tanker is discussed. The second case study considers

dynamic positioning of a supply vessel. In both cases the control law and observer parameters are chosen according to:

$$\begin{aligned}
\mathbf{P}_1 &= \text{diag}\{3.0, 3.0, 1.0\} \\
\mathbf{Q}_1 &= \text{diag}\{1.0, 1.0, 1.0\} \\
\mathbf{Q}_2 &= \text{diag}\{1.0, 1.0, 1.0\} \\
\mathbf{C}_1 &= 0.1 \cdot \text{diag}\{1.0, 1.0, 1.0\} \\
\mathbf{C}_2 &= 0.1 \cdot \text{diag}\{1.0, 1.0, 1.0\} \\
d_1 &= d_2 = 10.0, \qquad d_3 = d_4 = d_5 = d_6 = 1.0
\end{aligned}$$

The sampling time is 0.1 (s). In addition to this, Gaussian white noise was added to the measurements and the ship dynamics, in order to demonstrate the filtering properties of the observer.

Example 11.4 (Thruster Assisted PM of a Tanker)
The Bis-scaled system matrices for the moored tanker in Fossen and Grøvlen (1998) are:

$$\mathbf{M}'' = \begin{bmatrix} 1.0852 & 0 & 0 \\ 0 & 2.0575 & -0.4087 \\ 0 & -0.4087 & 0.2153 \end{bmatrix}, \quad \mathbf{D}'' = \begin{bmatrix} 0.0865 & 0 & 0 \\ 0 & 0.0762 & 0.1510 \\ 0 & 0.0151 & 0.0031 \end{bmatrix}$$

$$\mathbf{K}'' = diag\{0.0389, 0.0266, 0\}$$

Notice that $K_{33} = 0$ (no mooring moment in yaw). The non-dimensional eigenvalues of the system matrix:

$$\mathbf{A}_2 = -(\mathbf{M}'')^{-1}\mathbf{D}''$$

are $\lambda_1 = -0.07\,97, \lambda_2 = -0.3498$, and $\lambda_3 = 0.0\,212$. Hence, the tanker is course-unstable since the non-dimensional eigenvalue in yaw is positive. Consequently, the requirement that $\mathbf{A}_2$ is Hurwitz is violated. This did not have an effect of the performance of the ship mainly due to the stabilizing effect and robustness of the control law. It should be noted that if stabilization is a problem, the non-Hurwitz solution for the observer gain matrices should be applied (Robertson and Johansson 1998). The dimensional time constants are computed according to:

$$T_i = -\frac{1}{\lambda_i}\sqrt{L/g} \qquad (i = 1, \ldots, 3) \tag{11.150}$$

where $L = 200.6$ (m) is the length of the ship hull and $g = 9.81$ (m/s^2). Hence the time constants in surge, sway and yaw are found to be $56.7, 12.9$, and -213.5 (s). The performance of the nonlinear control law is shown in Figures 11.9 where the desired yaw angle (heading) command are $10, 5$ and 0 (deg). The desired (n, e)-positions are shifted from $(-10, 10)$ to $(0, 0)$ during the course-changing maneuver. Smooth reference trajectories in surge, sway and yaw are generated by using three 2nd-order low-pass filters with relative damping ratios equal to 1.0 and natural frequencies equal to 0.5 (rad/s). The computer simulations show that the output feedback controller is highly robust for noise contaminated measurements.

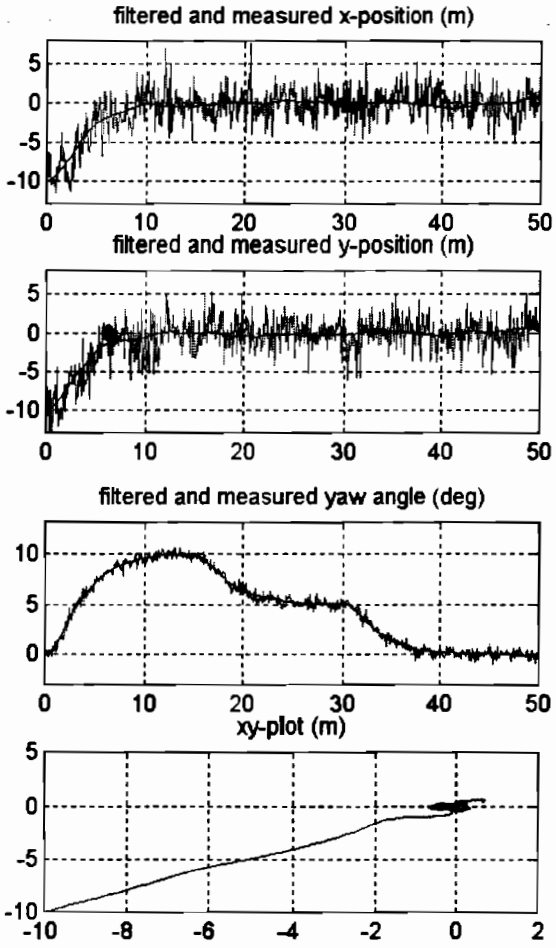


Figure 11.9: Thruster assisted mooring of a tanker. The desired positions are changed from (-10,-10) to (0,0). The desired yaw angle is changed between 10, 5 and 0 (deg).

Example 11.5 (Dynamic Positioning of a Supply Vessel)
The Bis-scaled system matrices for the supply vessel in Fossen and Grøvlen (1998) are:

$$\mathbf{M}''=\begin{bmatrix} 1.1274 & \mathbf{0} & 0 \\ 0 & 1.8902 & -0.0744 \\ 0 & -0.0744 & 0.1278 \end{bmatrix}, \quad \mathbf{D}''=\begin{bmatrix} 0.0358 & 0 & \mathbf{0} \\ 0 & 0.1183 & -0.0124 \\ 0 & -0.0041 & 0.0308 \end{bmatrix}$$

Notice that $\mathbf{K}''=\mathbf{0}$ *for the supply vessel (no mooring forces). The model parameters of the supply vessel have been identified by performing full-scale sea-trials in the North Sea, see Fossen* et al. *(1996) for details. The non-dimensional eigenvalues of the system matrix* $\mathbf{A}_2=-(\mathbf{M}'')^{-1}\mathbf{D}''$ *are* $\lambda_1=-0.2429$, $\lambda_2=-0.0627$ *and* $\lambda_2=-0.0318$. *The dimensional time constants are computed according to (11.150) with* $L=76.2$ *(m) resulting in* $11.5, 44.5$ *and* 87.8 *(s) for surge, sway and yaw, respectively. The performance of the nonlinear control*

law is shown in Figures 11.10 where a time-varying smooth reference trajectory is used in both surge and yaw whereas the desired sway position is zero. Again, excellent performance is demonstrated for noise contaminated position measurements.

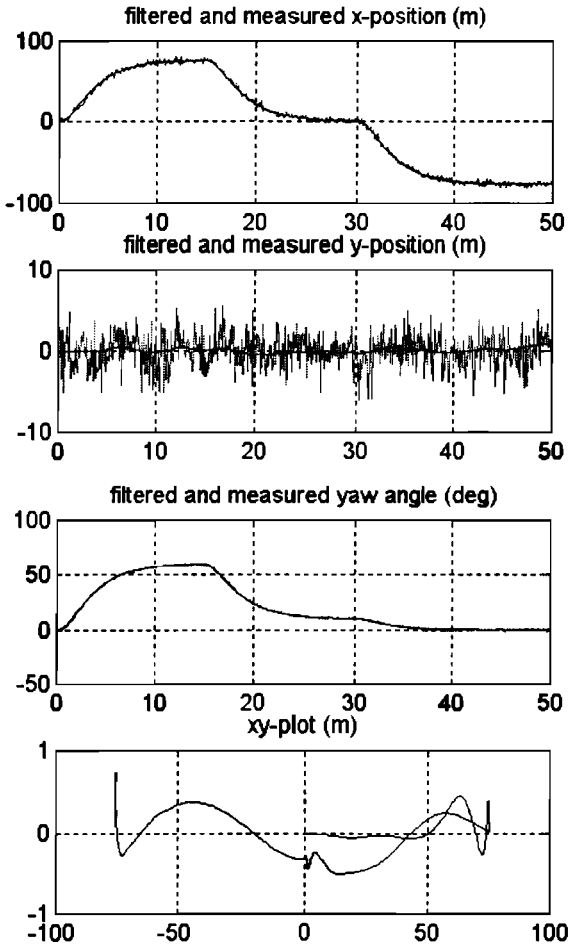


Figure 11.10: DP of a supply vessel. Tracking of a time-varying reference trajectory in surge and yaw whereas the desired sway position is zero.

11.2.6 Nonlinear Inverse Optimal Control

Backstepping designs can be related to optimal control theory by using the concept of *inverse optimal control*, see Strand and Fossen (1998), Strand *et al.* (1998*b*), and Strand (1999). This is based on Ezal (1998). A tutorial on nonlinear backstepping designs for ship control is found in Fossen and Strand (1998).

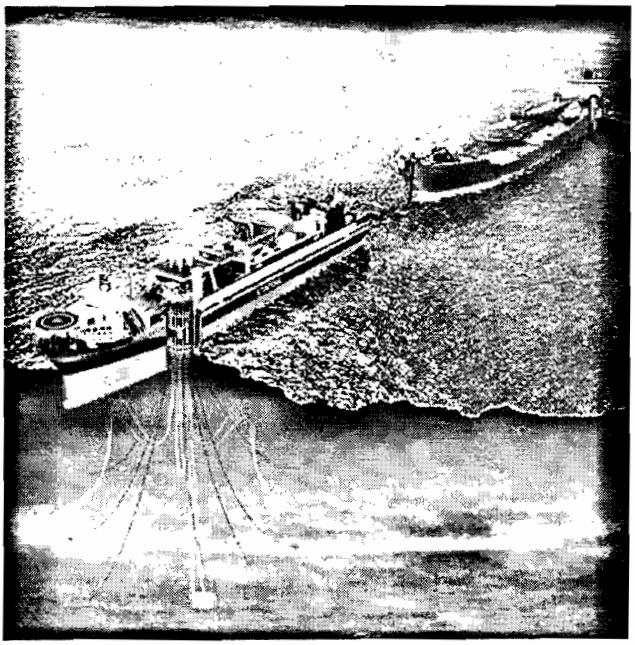


Figure 11.11: Positioning mooring system.

11.2.7 Underactuated Stabilization

An underactuated ship belongs to a class of systems that cannot be asymptotically stabilized by a static feedback control law $\mathbf{u} = \boldsymbol{\alpha}(\mathbf{x})$. The stabilization problem is, however, not solvable using linear control theory or "classical" nonlinear control theory like static feedback linearization. However, it is shown in Pettersen and Egeland (1996) that the ship is still small time locally controllable from any equilibrium point. By Coron (1995) this implies that it is possible to asymptotically stabilize the ship using a periodic time-varying feedback control law $\mathbf{u} = \boldsymbol{\beta}(\mathbf{x}, t)$. Therefore, to avoid the negative stabilizability result, one approach has been to introduce explicit time-dependence in the feedback control law, $\mathbf{u} = \boldsymbol{\beta}(\mathbf{x}, t)$, an approach first used by Samson (1991) for the control of mobile robots. Another approach has been the use of discontinuous feedback control laws, which give convergence to the desired equilibrium point, though not Lyapunov stability.

In Wichlund *et al.* (1995) a continuous feedback control law is proposed that instead asymptotically stabilizes an equilibrium manifold. The desired equilibrium point is then stable as all the system variables are bounded by the initial conditions of the system. Furthermore, the position variables with this approach converge exponentially to their desired values. The yaw angle, however, converges to some constant value, but not necessarily to zero. In Reyhanoglu (1996) a discontinuous feedback control law is proposed, and this provides exponential convergence to the desired equilibrium point, under certain assumptions on the initial value. In Pettersen and Egeland (1996) a time-varying feedback control law is proposed that provides exponential stability of the desired equilibrium point. However the feedback law only locally stabilizes the desired equilibrium point. In Pettersen and Nijmeijer

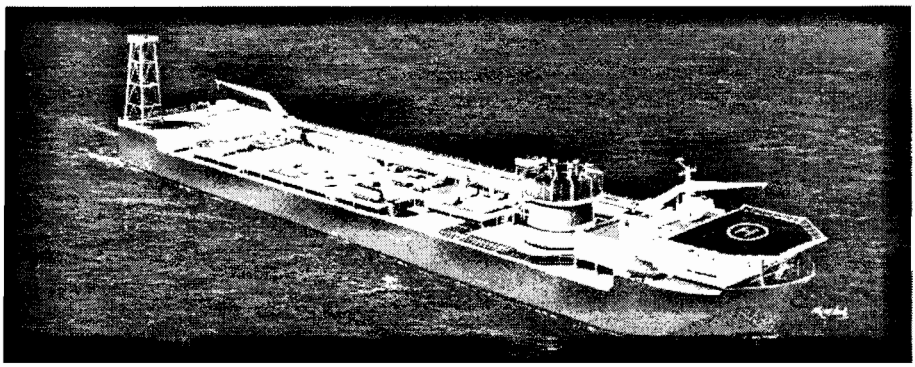


Figure 11.12: Anchored drilling vessel.

(1999*a*) a time-varying feedback control law is proposed that provides semiglobal practical exponential stability of a simplified model of the ship.

Control of a hovercraft is discussed by Fantoni *et al.* (2000) while a geometric framework for controllability analysis and motion control is proposed for mechanical systems on Lie groups, including the howercraft, by Bullo *et al.* (1999).

Even though there have been many theoretical developments in the area of underactuated systems, there have been relatively few experimental results reported that makes use of the developed theory. However, in Pettersen and Fossen (2000) experimental results for an underactuated ship are presented. In Pettersen and Egeland (1996), using averaging theory and homogeneity properties, a time-varying feedback control law was developed that exponentially stabilizes n, e, and ψ using only two control inputs. The experimental results are reported in Pettersen and Fossen (2000) where it was shown that the ship under the exponentially stabilizing feedback control law of Pettersen and Egeland (1996), experienced some stationary oscillations. Simulations indicated that the main reason for the oscillatory behavior was the environmental disturbances. It was thus pointed out that a topic of future research should be the inclusion of a constant environmental force in the analysis and control design, possibly including adaptation in the control scheme, in order to reduce or eliminate the stationary errors and oscillations.

11.3 Position Mooring (PM) Systems

In Section 11.1.1 a linear model for anchored and moored ships was presented. This model is written:

$$\dot{\boldsymbol{\eta}}_p = \boldsymbol{\nu} \tag{11.151}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{D}\boldsymbol{\nu} + \mathbf{K}\boldsymbol{\eta}_p = \boldsymbol{\tau} \tag{11.152}$$

where the term $\mathbf{K}\boldsymbol{\eta}_p$ represents the spring forces due to the mooring system. In thruster assisted position mooring (PM) systems the thrusters are complementary to the mooring sys-

tem and the main idea is to provide the system with additional damping, e.g. by using a D-controller:

$$\boldsymbol{\tau} = -\mathbf{K}_d \boldsymbol{\nu} \tag{11.153}$$

The mooring term $\mathbf{K}\boldsymbol{\eta}_p$ is in fact a P-controller but additional spring forces can be included by position feedback. Integral action is not used in PM systems, since the ship is allowed to move within a limited radius around the equilibrium point.

PM systems have been commercially available since the 1980's, and provide a flexible solution for floating structures for drilling and oil and gas exploitation on the smaller and marginal fields (Sørensen *et al.* 2000). Modeling and control of turret-moored ships are complicated problems since the mooring forces and moments are inherently nonlinear (Strand *et al.* 1998*a*). Control design of PM using nonlinear theory is addressed by Strand (1999).

11.4 Weather Optimal Positioning Control (WOPC)

Conventional DP systems for ships and free-floating rigs are usually designed for station-keeping by specifying a desired constant position (n_d, e_d) and a desired constant heading angle ψ_d. In order to minimize the ship fuel consumption, the desired heading ψ_d should in many operations be chosen such that the yaw moment is zero. For vessels with port/starboard symmetry, this means that the mean environmental force due to wind, waves and currents act through the center line of the vessel. Then the ship must be rotated until the yaw moment is zero.

Unfortunately, it is impossible to measure or compute the direction of the mean environmental force with sufficient accuracy. Hence, the desired heading ψ_d is usually taken to be the measurement of the mean wind direction, which can be easily measured. In practise, however, this can result in large off-sets from the true mean direction of the total environmental force. The main reason for this is the unmeasured current force component and waves that do not coincide with the wind direction. Hence, the DP system can be operated under highly non-optimal conditions if fuel saving is the issue. A small off-set in the optimal heading angle will result in a large use of thrust.

One attractive method for computing the weather optimal heading ψ_d is to monitor the resulting thruster forces in the x- and y-directions. Hence, the bow of the ship can be turned in one direction until the thruster force in the y-direction approaches zero. This method is appealing but the main catch in doing this is that the total resulting thruster forces in the x- and y-directions have to be computed since there are no sensors doing this job directly. The sensors only measure the angular speed and pitch angle of the propellers. Hence, the thrust for each propeller must be computed by using a model of the thruster characteristic resulting in a pretty rough estimate of the total thruster force in each direction.

Another principle, proposed by Pinkster and Nienhuis (1996), is to control the x- and y position using a PID feedback controller, in addition to feedback from the yaw velocity, such that the vessel tends towards the optimal heading. This principle, however, requires that the rotation point of the vessel be located a certain distance forward of the centre of gravity, or even fore of the bow, and it also puts restrictions on the thruster configuration and the number of thrusters installed.

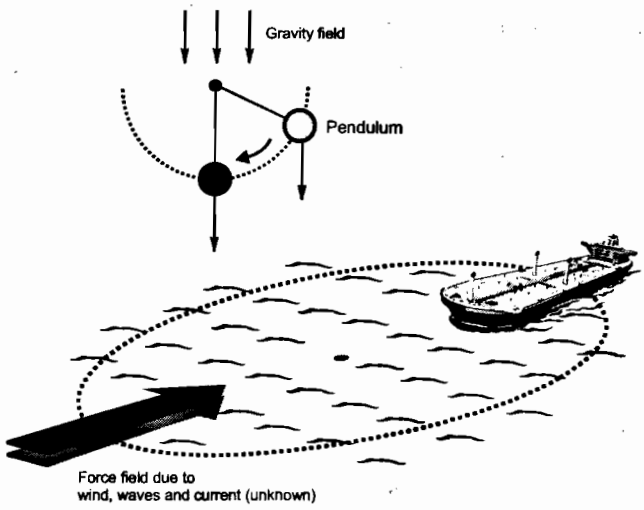


Figure 11.13: The principle of WOPC using the equivalence to a pendulum in the gravity field where gravity is the unmeasured quantity.

This chapter describes the design of a new concept for WOPC of ships and free-floating rigs. The WOPC controller was developed by Fossen and Strand (2001) . The control objective is that the vessel heading should adjust automatically to the mean environmental disturbances (wind, waves and currents) such that a minimum amount of energy is used in order to save fuel and reduce NO_x/CO_x-emissions without using any environmental sensors. This is particular useful for shuttle tankers and FPSOs, which can be located at the same position for a long time. Also DP operated supply vessels which must keep their position for days in loading/off-loading operations have a great WOPC fuel saving potential.

The ship can be exponentially stabilized on a circle arc with constant radius by letting the bow of the ship point towards the origin of the circle. In order to maintain a fixed position at the same time, a translatory circle center control law is designed. The circle center is translated on-line such that the Cartesian position is constant, while the bow of the ship is automatically turned up against the mean environmental force (weathervaning). This approach is motivated by a pendulum in the gravity field where gravity is the unmeasured quantity. The circular motion of the controlled ship, where the mean environmental force can be interpreted as an unknown force field, copies the dynamics of a pendulum in the gravity field; see Figure 11.13.

11.4.1 3 DOF Equations of Motion using Polar Coordinates

Consider the 3 DOF ship model:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{11.154}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau} + \mathbf{w} \tag{11.155}$$

where the North-East positions (n, e) and heading ψ is represented by $\boldsymbol{\eta} = [n, e, \psi]^\top$ and the vessel-fixed vessel velocities are represented by $\boldsymbol{\nu} = [u, v, r]^\top$. It is assumed that $\mathbf{M} =$

$\mathbf{M}^\top > 0$, $\dot{\mathbf{M}} = \mathbf{0}$, and $\mathbf{D}(\boldsymbol{\nu}) > 0$. Unmodeled external forces and moment due to wind, currents and waves are lumped together into an vessel-fixed disturbance vector $\mathbf{w} \in \mathbb{R}^3$ to be interpreted later.

Polar Coordinates

The Cartesian coordinates (n, e) is related to the *polar coordinates* by:

$$n = n_0 + \rho\cos\gamma, \qquad e = e_0 + \rho\sin\gamma, \tag{11.156}$$

where (n_0, e_0) is the origin of a circle with radius ρ and polar angle γ:

$$\rho = \sqrt{(n-n_0)^2 + (e-e_0)}, \quad \gamma = \text{atan2}\left((e-e_0),(n-n_0)\right). \tag{11.157}$$

Time differentiation of (11.156), yields:

$$\dot{n} = \dot{n}_0 + \dot{\rho}\cos\gamma - \rho\sin\gamma\,\dot{\gamma}, \tag{11.158}$$

$$\dot{e} = \dot{e}_0 + \dot{\rho}\sin\gamma + \rho\cos\gamma\,\dot{\gamma}. \tag{11.159}$$

Define the state vectors:

$$\mathbf{p_0} \triangleq [n_0, e_0]^\top, \qquad \mathbf{x} \triangleq [\rho, \gamma, \psi]^\top. \tag{11.160}$$

From (11.158) and (11.159) a new kinematic relationship can be written in terms of the vectors $\mathbf{p_0}$ and $\mathbf{x}$ as:

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\gamma)\mathbf{H}(\rho)\dot{\mathbf{x}} + \mathbf{L}\dot{\mathbf{p}}_0, \tag{11.161}$$

where:

$$\mathbf{H}(\rho) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \rho & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}. \tag{11.162}$$

From (11.161) the Cartesian kinematics (11.154) can be replaced by a differential equation for the polar coordinates:

$$\dot{\mathbf{x}} = \mathbf{T}(\mathbf{x})\boldsymbol{\nu} - \mathbf{T}(\mathbf{x})\mathbf{R}^\top(\psi)\mathbf{L}\dot{\mathbf{p}}_0, \tag{11.163}$$

where

$$\mathbf{T}(\mathbf{x}) = \mathbf{H}^{-1}(\rho)\underbrace{\mathbf{R}^\top(\gamma)\mathbf{R}(\psi)}_{\mathbf{R}^\top(\gamma-\psi)} \tag{11.164}$$

Note that the conversion between Cartesian and polar coordinates is only a local diffeomorphism, since the radius must be kept larger than a minimum value, i.e. $\rho > \rho_{\min} > 0$ in order to avoid the singular point $\rho = 0$.

Ship model transformation

The ship model (11.155) can be represented by polar coordinates by using (11.163) and substituting

$$\boldsymbol{\nu} = \mathbf{T}^{-1}(\mathbf{x})\dot{\mathbf{x}} + \mathbf{R}^{\top}\mathbf{L}\dot{\mathbf{p}}_0 \tag{11.165}$$

$$\dot{\boldsymbol{\nu}} = \mathbf{T}^{-1}(\mathbf{x})\ddot{\mathbf{x}} + \dot{\mathbf{T}}^{-1}(\mathbf{x})\dot{\mathbf{x}} + \mathbf{R}^{\top}\mathbf{L}\ddot{\mathbf{p}}_0 + \dot{\mathbf{R}}^{\top}\mathbf{L}\dot{\mathbf{p}}_0, \tag{11.166}$$

such that:

$$\begin{aligned} \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} &= \boldsymbol{\tau} + \mathbf{w} \\ &\Updownarrow \quad \rho > 0 \\ \mathbf{M}_x(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}_x(\boldsymbol{\nu},\mathbf{x})\dot{\mathbf{x}} + \mathbf{D}_x(\boldsymbol{\nu},\mathbf{x})\dot{\mathbf{x}} &= \mathbf{T}^{-\top}[\mathbf{q}(\boldsymbol{\nu},\mathbf{x},\dot{\mathbf{p}}_0,\ddot{\mathbf{p}}_0) + \boldsymbol{\tau} + \mathbf{w}] \end{aligned} \tag{11.167}$$

where:

$$\begin{aligned} \mathbf{M}_x(\mathbf{x}) &= \mathbf{T}^{-\top}(\mathbf{x})\mathbf{M}\mathbf{T}^{-1}(\mathbf{x}) \\ \mathbf{C}_x(\boldsymbol{\nu},\mathbf{x}) &= \mathbf{T}^{-\top}(\mathbf{x})\left(\mathbf{C}(\boldsymbol{\nu}) - \mathbf{M}\mathbf{T}^{-1}(\mathbf{x})\dot{\mathbf{T}}(\mathbf{x})\right)\mathbf{T}^{-1}(\mathbf{x}) \\ \mathbf{D}_x(\boldsymbol{\nu},\mathbf{x}) &= \mathbf{T}^{-\top}(\mathbf{x})\mathbf{D}(\boldsymbol{\nu})\mathbf{T}^{-1}(\mathbf{x}) \\ \mathbf{q}(\boldsymbol{\nu},\mathbf{x},\dot{\mathbf{p}}_0,\ddot{\mathbf{p}}_0) &= \mathbf{M}\mathbf{R}^{\top}(\psi)\mathbf{L}\ddot{\mathbf{p}}_0 + \mathbf{M}\dot{\mathbf{R}}^{\top}(\psi)\mathbf{L}\dot{\mathbf{p}}_0 + [\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu})]\mathbf{R}^{\top}(\psi)\mathbf{L}\dot{\mathbf{p}}_0 \end{aligned}$$

Here $\mathbf{M}_x(\mathbf{x})$, $\mathbf{C}_x(\boldsymbol{\nu},\mathbf{x})$, and $\mathbf{D}_x(\boldsymbol{\nu},\mathbf{x})$ can be shown to satisfy:

$$\mathbf{M}_x(\mathbf{x}) = \mathbf{M}_x^{\top}(\mathbf{x}) > 0\,, \quad \mathbf{D}_x(\boldsymbol{\nu},\mathbf{x}) > 0, \quad \forall \mathbf{x}.$$

The ship dynamics does also satisfy the skew-symmetric property:

$$\mathbf{z}^{\top}\left(\dot{\mathbf{M}}_x - 2\mathbf{C}_x\right)\mathbf{z} = 0,\ \forall \mathbf{z},\mathbf{x}. \tag{11.168}$$

Disturbance Modeling

The steady-state LF motion of the ship and also the ship's equilibrium position depend on the *unknown* environmental loads acting on the vessel. Let the environmental loads due to *wind, waves,* and *currents* be represented by:

- a slowly-varying mean force F_e which attacks the ship in a point (l_x, l_y) in body-fixed coordinates.
- a slowly-varying mean direction β_e, relative to the Earth-fixed frame; see Figure 11.14.

The WF motion is assumed to be filtered out of the measurements by using a wave filter; see Chapter 6.

Since there are no sensors which can be used to measure (F_e, β_e) and (l_x, l_y) with sufficient accuracy, it is impossible to use feedforward from the environmental disturbances. This leads to the following assumptions:

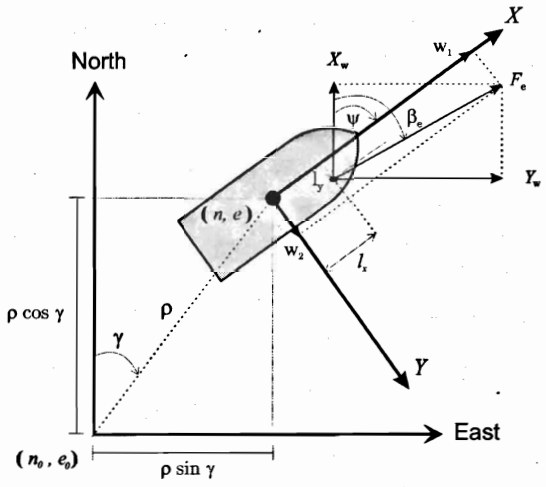


Figure 11.14: Environmental force F_e decomposed into the components w_1 and w_2.

A1: *The unknown mean environmental force F_e and it's direction β_e are assumed to be constant or at least slowly-varying.*

A2: *The unknown attack point (l_x, l_y) is constant for each constant F_e.*

Discussion: *These are good assumptions since the ship control system is only supposed to counteract the slowly-varying motion components of the environmental disturbances.*

From Figure 11.14 the body-fixed environmental load vector $\mathbf{w} \in \mathbb{R}^3$ can be expressed as:

$$\mathbf{w} = \begin{bmatrix} w_1(\psi) \\ w_2(\psi) \\ w_3(\psi) \end{bmatrix} = \begin{bmatrix} F_e \cos(\beta_e - \psi) \\ F_e \sin(\beta_e - \psi) \\ l_x F_e \sin(\beta_e - \psi) - l_y F_e \cos(\beta_e - \psi) \end{bmatrix}. \tag{11.169}$$

Notice that the environmental loads vary with the heading angle ψ of the ship. Consequently:

$$F_e = \sqrt{w_1^2 + w_2^2}, \quad \beta_e = \psi + \tan^{-1}(w_2/w_1). \tag{11.170}$$

The environmental forces X_w and Y_w with attack point $(l_x(\psi), l_y(\psi))$ are shown in Figure 11.14. Note that the attack point will change with the yaw angle ψ. This relationship will be a complicated function of hull and superstructure geometries.

11.4.2 Weather Optimal Control Objectives

The weather optimal control objectives make use of the following definitions (Fossen and Strand 2001):

Definition 11.2 (Weather Optimal Heading)
The weather optimal heading angle ψ_{opt} is given by the equilibrium state where the yaw moment $w_3(\psi_{\text{opt}}) = 0$ at the same time as the bow of the ship is turned up against weather

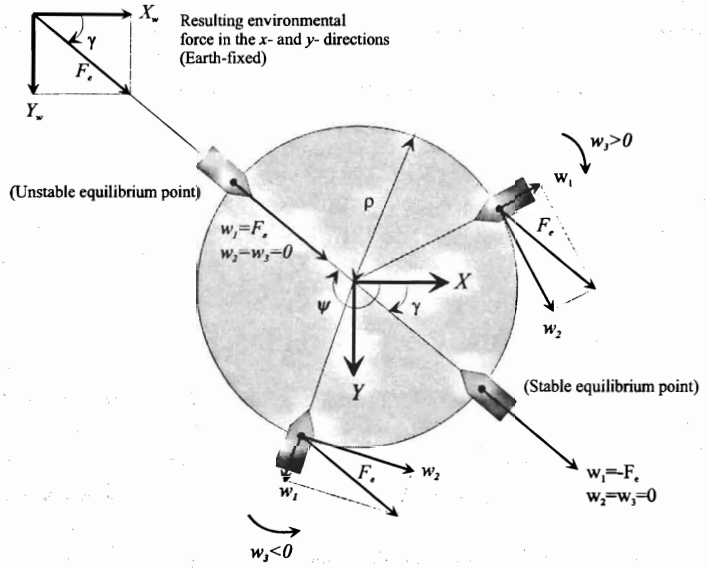


Figure 11.15: Stable and unstable equilibrium points for WOPC.

(mean environmental disturbances), that is $w_2(\psi_{\text{opt}}) = 0$. *This implies that the moment arms* $l_x(\psi_{\text{opt}}) =$ *constant and* $l_y(\psi_{\text{opt}}) = 0$, *and:*

$$\mathbf{w}(\psi_{\text{opt}}) = \begin{bmatrix} w_1(\psi_{\text{opt}}) \\ w_2(\psi_{\text{opt}}) \\ w_3(\psi_{\text{opt}}) \end{bmatrix} = \begin{bmatrix} -F_e \\ 0 \\ 0 \end{bmatrix}.$$

Hence, the mean environmental force attacks the ship in the bow (minimum drag coefficient for water and wind loads)

Definition 11.3 (Weather Optimal Positioning)
Weather optimal positioning (station keeping) is defined as the equilibrium state where ψ_{opt} *satisfies:*

$$w_1(\psi_{\text{opt}}) = -F_e, \qquad w_2(\psi_{\text{opt}}) = w_3(\psi_{\text{opt}}) = l_y(\psi_{\text{opt}}) = 0 \tag{11.171}$$

and the position $(n, e) = (n_d, e_d)$ *is kept constant.*

These definitions motivates the following two control objectives:

O1: **Weather Optimal Heading Control (WOHC):** This is obtained by restricting the ship's movement to a circle with constant radius $\rho = \rho_d$ and at the same time force the ship's bow to point towards the center of the circle until the weather optimal heading angle $\psi = \psi_{\text{opt}}$ is reached; see Figure 11.15. An analogy to this is a pendulum in gravity field; see Figure 11.13. The position $(n, e) = (n_0 + \rho\cos\gamma, e_0 + \rho\sin\gamma)$ will vary until the weather optimal heading angle is reached. This is obtained by specifying the control objective in polar coordinates according to:

$$\rho_d = \text{constant}, \quad \dot{\gamma}_d = 0, \quad \psi_d = \pi + \gamma. \tag{11.172}$$

Discussion: The requirement ρ_d = constant implies that the ship moves on a circle with constant radius. The second requirement $\dot{\gamma}_d = 0$ implies that the tangential speed $\rho\dot{\gamma}$ is kept small while the last requirement $\psi_d = \pi + \gamma$ ensures that the ship's bow points towards the center of the circle.

O2: **Weather Optimal Positioning Control (WOPC):** In order to maintain a fixed Earth-fixed position $(n, e) = (n_d, e_d)$, the circle center $\mathbf{p}_0 = [n_0, e_0]^\top$ must be moved simultaneously as Control Objective O1 is satisfied. The is referred to as *translatory circle center control.*

11.4.3 Nonlinear and Adaptive Control Design

The WOPC positioning controller is derived by using the polar coordinate representation. The backstepping design methodology (Krstic *et al.* 1995) with extension to integral control (Fossen *et al.* 2001) is used to derive the feedback controller (see Section 7.4). It is assumed that all states can be measured by using conventional sensor technology and a satellite navigation system; see Chapter 6.

The WOPC controller will be derived in 3 successive steps:

1. *Nonlinear backstepping (PD-control):* the ship is forced to move on a circle arc with desired radius ρ_d, with minimum tangential velocity $\rho\dot{\gamma}$ and desired heading ψ_d.
2. *Adaptive backstepping (PID-control):* this is necessary to compensate for the unknown environmental force F_e.
3. *Translational control of the circle center:* the circle center (n_0, e_0) is translated such that the ship maintains a constant position (n_d, e_d) even though it is moving on a virtual circle arc. Hence, the Captain of the ship will only notice that the ship is rotating a yaw angle ψ about a constant position (n_d, e_d) until the weather optimal heading ψ_{opt} is reached.

Nonlinear Backstepping (PD-Control)

A general positioning controller is derived by using vectorial backstepping (Fossen and Grøvlen 1998). The tracking objective is specified in polar coordinates using a smooth reference trajectory $\mathbf{x}_d = [\rho_d, \gamma_d, \psi_d]^\top \in C^3$ where:

$$\mathbf{x}_d, \dot{\mathbf{x}}_d, \ddot{\mathbf{x}}_d \in \mathcal{L}_\infty.$$

Since the transformed system (11.167) is of order 2, backstepping is performed in two *vectorial steps* resulting in a nonlinear PD-control law. First, a *virtual reference trajectory* is defined as:

$$\dot{\mathbf{x}}_r := \dot{\mathbf{x}}_d - \mathbf{\Lambda}\mathbf{z}_1 \tag{11.173}$$

where $\mathbf{z}_1 = \mathbf{x} - \mathbf{x}_d$ is the NED tracking error and $\mathbf{\Lambda} > 0$ is a diagonal design matrix. Furthermore, let $\mathbf{z}_2$ denote a measure of tracking defined according to:

$$\mathbf{z}_2 \triangleq \dot{\mathbf{x}} - \dot{\mathbf{x}}_r = \dot{\mathbf{z}}_1 + \mathbf{\Lambda}\mathbf{z}_1 \tag{11.174}$$

From (11.174), the following expressions are obtained:

$$\dot{\mathbf{x}} = \mathbf{z}_2 + \dot{\mathbf{x}}_r, \quad \ddot{\mathbf{x}} = \dot{\mathbf{z}}_2 + \ddot{\mathbf{x}}_r \tag{11.175}$$

This implies that the vessel model (11.167) can be expressed in terms of $\mathbf{z}_2$, $\dot{\mathbf{x}}_r$, and $\ddot{\mathbf{x}}_r$ as:

$$\mathbf{M}_x\dot{\mathbf{z}}_2+\mathbf{C}_x\mathbf{z}_2+\mathbf{D}_x\mathbf{z}_2=\mathbf{T}^{-\top}\boldsymbol{\tau}+\mathbf{T}^{-\top}\mathbf{q}(\cdot)-\mathbf{M}_x\ddot{\mathbf{x}}_r-\mathbf{C}_x\dot{\mathbf{x}}_r-\mathbf{D}_x\dot{\mathbf{x}}_r+\mathbf{T}^{-\top}\mathbf{w} \quad (11.176)$$

Step 1:
Let $\mathbf{z}_1$ be the first error variable, which from (11.174) has the dynamics:

$$\dot{\mathbf{z}}_1=-\boldsymbol{\Lambda}\mathbf{z}_1+\mathbf{z}_2 \quad (11.177)$$

A CLF for the first step is:

$$V_1=\frac{1}{2}\mathbf{z}_1^T\mathbf{K}_p\mathbf{z}_1 \quad (11.178)$$

$$\dot{V}_1=-\mathbf{z}_1^T\mathbf{K}_p\boldsymbol{\Lambda}\mathbf{z}_1+\mathbf{z}_1^T\mathbf{K}_p\mathbf{z}_2 \quad (11.179)$$

where $\mathbf{K}_p=\mathbf{K}_p^T>0$ is a constant design matrix.

Step 2:
In the second step the CLF is motivated by the "pseudo" kinetic energy, that is:

$$V_2=V_1+\frac{1}{2}\mathbf{z}_2^T\mathbf{M}_x\mathbf{z}_2,\quad \mathbf{M}_x=\mathbf{M}_x^T>0 \quad (11.180)$$

Time differentiation of V_2 along the trajectories of $\mathbf{z}_1$ and $\mathbf{z}_2$, gives:

$$\dot{V}_2=\dot{V}_1+\mathbf{z}_2^\top\mathbf{M}_x\dot{\mathbf{z}}_2+\frac{1}{2}\mathbf{z}_2^\top\dot{\mathbf{M}}_x\mathbf{z}_2 \quad (11.181)$$

which by substitution of (11.179) and (11.176), gives:

$$\begin{aligned}\dot{V}_2 &= -\mathbf{z}_1^\top\mathbf{K}_p\boldsymbol{\Lambda}\mathbf{z}_1+\frac{1}{2}\mathbf{z}_2^\top\left(\dot{\mathbf{M}}_x-2\mathbf{C}_x\right)\mathbf{z}_2-\mathbf{z}_2^T\mathbf{D}_x\mathbf{z}_2+\mathbf{z}_2^T\mathbf{T}^{-T}\mathbf{w}\\ &\quad+\mathbf{z}_2^T\left(\mathbf{K}_p\mathbf{z}_1+\mathbf{T}^{-\top}\boldsymbol{\tau}+\mathbf{T}^{-\top}\mathbf{q}(\cdot)-\mathbf{M}_x\ddot{\mathbf{x}}_r-\mathbf{C}_x\dot{\mathbf{x}}_r-\mathbf{D}_x\dot{\mathbf{x}}_r\right) \quad (11.182)\end{aligned}$$

By using the property (11.168) and choosing the nonlinear PD-control law as:

$$\mathbf{T}^{-\top}\boldsymbol{\tau}=\mathbf{M}_x\ddot{\mathbf{x}}_r+\mathbf{C}_x\dot{\mathbf{x}}_r+\mathbf{D}_x\dot{\mathbf{x}}_r-\mathbf{K}_p\mathbf{z}_1-\mathbf{K}_d\mathbf{z}_2-\mathbf{T}^{-\top}\mathbf{q}(\cdot) \quad (11.183)$$

where $\mathbf{K}_d>0$ is a strictly positive design matrix, we finally get:

$$\dot{V}_2=-\mathbf{z}_1^\top\mathbf{K}_p\boldsymbol{\Lambda}\mathbf{z}_1-\mathbf{z}_2^\top(\mathbf{K}_d+\mathbf{D}_x)\mathbf{z}_2+\mathbf{z}^\top\mathbf{T}^{-\top}\mathbf{w} \quad (11.184)$$

Notice that the dissipative term $\mathbf{z}_2^\top\mathbf{D}_x\mathbf{z}_2>0,\forall\mathbf{z}_2\neq\mathbf{0}$ is exploited in the design as it appears in the expression for $\dot{V}_2$. With the control law (11.183) the closed-loop dynamics becomes:

$$\mathbf{M}_x\dot{\mathbf{z}}_2+(\mathbf{C}_x+\mathbf{D}_x+\mathbf{K}_d)\mathbf{z}_2+\mathbf{K}_p\mathbf{z}_1=\mathbf{T}^{-\top}\mathbf{w} \quad (11.185)$$

Error dynamics:
The error dynamics of the resulting system becomes *nonautonomous* since:

$$\begin{aligned}\begin{bmatrix}\mathbf{K}_p & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{M}_x\end{bmatrix}\begin{bmatrix}\dot{\mathbf{z}}_1\\ \dot{\mathbf{z}}_2\end{bmatrix} &= -\begin{bmatrix}\mathbf{K}_p\boldsymbol{\Lambda} & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{C}_x+\mathbf{D}_x+\mathbf{K}_d\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}\\ &\quad+\begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{K}_p\\ -\mathbf{K}_p & \mathbf{0}_{3\times3}\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}+\begin{bmatrix}\mathbf{0}_{3\times1}\\ \mathbf{T}^{-\top}\end{bmatrix}\mathbf{w}\\ &\Updownarrow\\ \mathcal{M}(\mathbf{x})\dot{\mathbf{z}} &= -\mathcal{K}(\mathbf{x},\boldsymbol{\nu})\mathbf{z}+\mathcal{S}\mathbf{z}+\bar{\mathcal{B}}(\mathbf{x})\mathbf{w} \quad (11.186)\end{aligned}$$

where the different matrices are defined as:

$$\mathcal{M}(\mathbf{x}) = \mathcal{M}^T(\mathbf{x}) = \begin{bmatrix} \mathbf{K}_p & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{M}_x(\mathbf{x}) \end{bmatrix}$$

$$\mathcal{K}(\mathbf{x},\boldsymbol{\nu}) = \begin{bmatrix} \mathbf{K}_p\boldsymbol{\Lambda} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{C}_x(\mathbf{x},\boldsymbol{\nu}) + \mathbf{D}_x(\mathbf{x},\boldsymbol{\nu}) + \mathbf{K}_d \end{bmatrix} > 0$$

$$\mathcal{S} = -\mathcal{S}^T = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{K}_p \\ -\mathbf{K}_p & \mathbf{0}_{3\times3} \end{bmatrix}, \quad \bar{\mathcal{B}}(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{3\times1} \\ \mathbf{T}^{-\top}(\mathbf{x}) \end{bmatrix}$$

In the absence of disturbances, $\mathbf{w} \equiv \mathbf{0}$, the origin $\mathbf{z} = 0$ is uniformly locally exponentially stable (ULES) according to Lyapunov. Global results cannot be achieved due to the local diffeomorphism between the Cartesian and polar coordinates, that is the transformation matrix $\mathbf{T}(\mathbf{x})$ is singular for $\rho = 0$.

With disturbances $\mathbf{w} \neq \mathbf{0}$, the closed-loop system is input-to-state stable (ISS). In the next section, it is shown how adaptive backstepping (backstepping with integral action) can be used to obtain ULES for the case of a non-zero disturbance vector $\mathbf{w} \neq \mathbf{0}$.

Adaptive Backstepping (PID-Control)

Since the mean disturbance $\mathbf{w}$ is non-zero this will result in a steady-state offset when using the PD-controller of the previous section. The ship is, however, restricted to move on a circle arc with $\mathbf{w}$ as a force field. Therefore there will be a stable and an unstable equilibrium point on the circle arc (similar to a pendulum in the gravity field); see Figure 11.13. The stable equilibrium point is given by:

$$\mathbf{w} = \boldsymbol{\phi} F_e = [-1, 0, 0]^\top F_e \tag{11.187}$$

Since, the disturbance F_e is assumed to be slowly-varying, adaptive backstepping can be applied to obtain integral effect in the system. Thus, in the analysis it will be assumed that $\dot{F}_e$=0. Let the estimate of F_e be denoted as $\hat{F}_e$, and $\tilde{F} = \hat{F}_e - F_e$. An additional step in the derivation of the backstepping control law must be performed in order to obtain an adaptive update law for $\hat{F}_e$:

Step 3:
The adaptive update law is found by adding the square parameter estimation error to V_2. Consequently:

$$V_3 = V_2 + \frac{1}{2\sigma}\tilde{F}_e^2, \ \sigma > 0 \tag{11.188}$$

$$\dot{V}_3 = \dot{V}_2 + \frac{1}{\sigma}\dot{\tilde{F}}_e\tilde{F}_e \tag{11.189}$$

The nonlinear control law (11.183) is modified to:

$$\boldsymbol{\tau} = \mathbf{T}^\top(\mathbf{M}_x\ddot{\mathbf{x}}_r + \mathbf{C}_x\dot{\mathbf{x}}_r + \mathbf{D}_x\dot{\mathbf{x}}_r - \mathbf{K}_p\mathbf{z}_1 - \mathbf{K}_d\mathbf{z}_2) - \mathbf{q}(\cdot) - \boldsymbol{\phi}\hat{F}_e \tag{11.190}$$

where the last term $\boldsymbol{\phi}\hat{F}_e$ provides integral action. Hence, the $\mathbf{z}_2$-dynamics becomes:

$$\mathbf{M}_x\dot{\mathbf{z}}_2 + (\mathbf{C}_x + \mathbf{D}_x + \mathbf{K}_d)\mathbf{z}_2 + \mathbf{K}_p\mathbf{z}_1 = -\mathbf{T}^{-\top}\boldsymbol{\phi}\tilde{F}_e \tag{11.191}$$

This implies that:

$$\begin{aligned}\dot{V}_3 &= -\mathbf{z}_1^\top \mathbf{K}_p \mathbf{\Lambda} \mathbf{z}_1 - \mathbf{z}_2^\top (\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 - \mathbf{z}_2^\top \mathbf{T}^{-\top} \phi \tilde{F}_e + \frac{1}{\sigma} \dot{\tilde{F}}_e \tilde{F}_e \\ &= -\mathbf{z}_1^\top \mathbf{K}_p \mathbf{\Lambda} \mathbf{z}_1 - \mathbf{z}_2^\top (\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 + \tilde{F}_e(-\phi^\top \mathbf{T}^{-1} \mathbf{z}_2 + \frac{1}{\sigma} \dot{\tilde{F}}_e) \end{aligned} \quad (11.192)$$

The adaptive law $\dot{\hat{F}}_e = \dot{\tilde{F}}_e$ is chosen as:

$$\dot{\hat{F}}_e = \sigma \phi^\top \mathbf{T}^{-1} \mathbf{z}_2, \sigma > 0 \quad (11.193)$$

such that:

$$\dot{V}_3 = -\mathbf{z}_1^\top \mathbf{K}_p \mathbf{\Lambda} \mathbf{z}_1 - \mathbf{z}_2^\top (\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 \leq 0 \quad (11.194)$$

Error Dynamics

The nonautonomous error dynamics for the adaptive backstepping controller can be written:

$$\mathcal{M}(\mathbf{x})\dot{\mathbf{z}} = [-\mathcal{K}(\mathbf{x}, \boldsymbol{\nu}) + \mathcal{S}]\mathbf{z} + \mathcal{B}(\mathbf{x})\tilde{F}_e \quad (11.195)$$

$$\dot{\tilde{F}}_e = -\sigma \mathcal{B}^T(\mathbf{x})\mathbf{z} \quad (11.196)$$

where:

$$\mathcal{B}(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ -\mathbf{T}^{-\top}(\mathbf{x})\phi \end{bmatrix} \quad (11.197)$$

In order to satisfy *Control Objective O1*, the controller gains must be chosen according to:

$$\mathbf{K}_p = \begin{bmatrix} k_{p1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & k_{p3} \end{bmatrix}, \quad \mathbf{K}_d = \begin{bmatrix} k_{d1} & 0 & 0 \\ 0 & k_{d2} & 0 \\ 0 & 0 & k_{d3} \end{bmatrix}, \quad \mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \quad (11.198)$$

Notice that $k_{p2} = \lambda_2 = 0$. This implies that the ship is free to move on the circle arc with tangential velocity $\rho\dot{\gamma}$. The gain $k_{d2} > 0$ is used to increase the tangential damping (D-control) while the radius ρ and heading ψ are stabilized by using PID-control

Semi-Definite Matrices

Since the controller gains k_{p2} and λ_2 are chosen to be zero, the matrices:

$$\mathbf{K}_p \geq 0, \quad \mathbf{\Lambda} \geq 0 \quad (11.199)$$

are only positive semi-definite resulting in a positive semi-definite V_3. Uniform local asymptotic stability (ULAS) of the equilibrium $(\mathbf{z}, \tilde{F}_e) = (\mathbf{0}, 0)$ can, however, be proven since the error dynamics $(\mathbf{z}_1, \mathbf{z}_2)$ is ISS. Consider the reduced order system $(\mathbf{z}_{1r}, \mathbf{z}_2)$ given by:

$$\mathbf{z}_{1r} = \mathbf{E}\mathbf{z}_1, \quad \mathbf{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (11.200)$$

This implies that:

$$\begin{aligned}\dot{\mathbf{z}}_{1r} &= -\mathbf{E}\boldsymbol{\Lambda}\mathbf{z}_1 + \mathbf{E}\mathbf{z}_2 \\ &= -(\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top)\mathbf{z}_{1r} + \mathbf{E}\mathbf{z}_2. \end{aligned} \tag{11.201}$$

Notice that the last step is possible since the diagonal matrices $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, 0, \lambda_3\}$ satisfies:

$$\boldsymbol{\Lambda}\mathbf{E}^\top\mathbf{z}_{1r} = \boldsymbol{\Lambda}\mathbf{z}_1. \tag{11.202}$$

Hence, the error dynamics (11.195)–(11.196) can be transformed to:

$$\mathcal{M}_r(\mathbf{x})\dot{\mathbf{z}}_r = [-\mathcal{K}_r(\mathbf{x}, \boldsymbol{\nu}) + \mathcal{S}_r]\mathbf{z}_r + \mathcal{B}_r(\mathbf{x})\tilde{F}_e \tag{11.203}$$

$$\dot{\tilde{F}}_e = -\sigma\mathcal{B}_r^\top(\mathbf{x})\mathbf{z}_r, \tag{11.204}$$

where $\mathbf{z}_r = [\mathbf{z}_{1r}^\top, \mathbf{z}_2^\top]^\top$ and:

$$\begin{aligned}\mathcal{M}_r(\mathbf{x}) &= \mathcal{M}_r^\top(\mathbf{x}) = \begin{bmatrix} (\mathbf{E}\mathbf{K}_p\mathbf{E}^\top) & \mathbf{0}_{2\times 3} \\ \mathbf{0}_{3\times 2} & \mathbf{M}_x(\mathbf{x}) \end{bmatrix} \\ \mathcal{K}_r(\mathbf{x}, \boldsymbol{\nu}) &= \begin{bmatrix} (\mathbf{E}\mathbf{K}_p\mathbf{E}^\top)(\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top) & \mathbf{0}_{2\times 3} \\ \mathbf{0}_{3\times 2} & \mathbf{C}_x(\mathbf{x}, \boldsymbol{\nu}) + \mathbf{D}_x(\mathbf{x}, \boldsymbol{\nu}) + \mathbf{K}_d \end{bmatrix} > 0 \\ \mathcal{S}_r &= -\mathcal{S}_r^T = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{E}\mathbf{K}_p \\ -\mathbf{K}_p\mathbf{E}^\top & \mathbf{0}_{3\times 3} \end{bmatrix}, \quad \mathcal{B}_r(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{2\times 1} \\ \mathbf{T}^{-\top}(\mathbf{x})\boldsymbol{\phi} \end{bmatrix}. \end{aligned}$$

where the fact that $\mathbf{K}_p\mathbf{E}^\top\mathbf{z}_{1r} = \mathbf{K}_p\mathbf{z}_1$ for $\mathbf{K}_p = \text{diag}\{k_{p1}, 0, k_{p3}\}$ has been applied.

Non-Autonomous Lyapunov Analysis

Even though the Lyapunov function V_3 corresponding to the states $(\mathbf{z}_1, \mathbf{z}_2)$ is only positive semi-definite (since $\mathbf{K}_p$ is positive semi-definite) the Lyapunov function V_{3r} corresponding to the new output $(\mathbf{z}_{1r}, \mathbf{z}_2)$ is positive definite. Using the fact that the closed loop system governed by $(\mathbf{z}_1, \mathbf{z}_2)$ is ISS, asymptotic tracking is guaranteed by:

$$V_{3r} = \frac{1}{2}\left[\mathbf{z}_{1r}^\top(\mathbf{E}\mathbf{K}_p\mathbf{E}^\top)\mathbf{z}_{1r} + \mathbf{z}_2^\top\mathbf{M}_x\mathbf{z}_2 + \frac{1}{\sigma}\tilde{F}_e^2\right] > 0, \tag{11.205}$$

$$\dot{V}_{3r} = -\mathbf{z}_{1r}^\top(\mathbf{E}\mathbf{K}_p\mathbf{E}^\top)(\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top)\mathbf{z}_{1r} - \mathbf{z}_2^\top(\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 \leq 0. \tag{11.206}$$

where $\mathbf{E}\mathbf{K}_p\mathbf{E}^\top > 0$ and $\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^\top > 0$. Hence, $\mathbf{z}_{1r}, \mathbf{z}_2, \tilde{F}_e \in \mathcal{L}_\infty$. Notice that $\dot{V}_3$ is only negative semi-definite since a negative term proportional to $-\tilde{F}_e^2$ is missing in the expression for $\dot{V}_3$. ULES of the equilibrium point $(\mathbf{z}_{1r}, \mathbf{z}_2, \tilde{F}_e) = (\mathbf{0}, \mathbf{0}, 0)$ follows by using the stability theorem of Fossen *et al.* (2001) for nonlinear *non-autonomous* systems; see Appendix A.2.4. Since, the closed-loop system $(\mathbf{z}_1, \mathbf{z}_2)$ is ISS it is sufficient to consider the reduced order system $(\mathbf{z}_{1r}, \mathbf{z}_2)$ with output $\mathbf{z}_{1r} = \mathbf{E}\mathbf{z}_1$ in the stability analysis. According to Appendix A.2.4, we can choose $\mathbf{x}_1 = [\mathbf{z}_{1r}^\top, \mathbf{z}_2^\top]^\top$, $\mathbf{x}_2 = \tilde{F}_e$, $\mathbf{P} = \sigma$ and $W(\mathbf{x}_1, t) = \frac{1}{2}\mathbf{x}_1^\top\mathbf{x}_1$. Then the equilibrium point $(\mathbf{z}_{1r}, \mathbf{z}_2, \tilde{F}_e) = (\mathbf{0}, \mathbf{0}, 0)$ of the nonlinear error system (11.195)–(11.196) is ULES since:

$$\text{rank}\{(\mathcal{M}_r^{-1}(\mathbf{x})\mathcal{B}_r(\mathbf{x}))^T(\mathcal{M}_r^{-1}(\mathbf{x})\mathcal{B}_r(\mathbf{x}))\} = 1, \forall \mathbf{x}$$

and:

$$\begin{aligned}\max\{\|\mathbf{h}(\mathbf{x}_1, t)\|, \|\mathbf{x}_1\|\} &= \max\{\|\mathcal{M}_r^{-1}(\mathbf{x})[-\mathcal{K}_r(\mathbf{x}, \boldsymbol{\nu}) + \mathcal{S}_r]\mathbf{x}_1\|, \|\mathbf{x}_1\|\} \\ &\leq \rho_1(\|\mathbf{x}_1\|)\|\mathbf{x}_1\| \end{aligned}$$

$$\begin{aligned} \|\mathbf{B}(\mathbf{x},t)\| &= \left\|\mathcal{M}_r^{-1}(\mathbf{x})\mathcal{B}_r(\mathbf{x})\right\| \leq \rho_2(\|\mathbf{x}_1\|) \\ \max\left\{\left\|\frac{\partial \mathbf{B}(\mathbf{x},t)}{\partial t}\right\|, \left\|\frac{\partial \mathbf{B}(\mathbf{x},t)}{\partial \mathbf{x}_i}\right\|\right\} &= \max\left\{\left\|\frac{\partial \mathcal{M}_r^{-1}(\mathbf{x})\mathcal{B}_r(\mathbf{x})}{\partial \mathbf{x}_i}\right\|\right\} \leq \rho_3(\|\mathbf{x}_1\|) \end{aligned}$$

Translational Control of the Circle Center

The adaptive backstepping controller satisfies Control objective O1–i.e., weather optimal heading control. Weather optimal position control, control objective O2, can be satisfied by moving the circle center $\mathbf{p}_0 = [n_0, e_0]^\top$ on-line such that the ship maintains a constant position $\mathbf{p} = [n, e]^\top$.

In order to meet the fixed position control objective, an update law for the circle center $\mathbf{p}_0$ must be derived. The Cartesian Earth-fixed position of the ship is given by:

$$\mathbf{p} = \mathbf{L}^\top \boldsymbol{\eta} \tag{11.207}$$

where $\mathbf{L}$ is defined in (11.162). Let $\tilde{\mathbf{p}} = \mathbf{p} - \mathbf{p}_d$ denote the corresponding deviation from the desired position vector $\mathbf{p}_d \triangleq [e_d, e_d]^\top$. The desired position can either be constant (regulation) or a smooth time-varying reference trajectory. The control law for translation of the circle center is derived by considering the following CLF:

$$V_p = \frac{1}{2}\tilde{\mathbf{p}}^\top \tilde{\mathbf{p}} \tag{11.208}$$

$$\dot{V}_p = \tilde{\mathbf{p}}^\top(\dot{\mathbf{p}} - \dot{\mathbf{p}}_d) = \tilde{\mathbf{p}}^\top(\mathbf{L}^\top \dot{\boldsymbol{\eta}} - \dot{\mathbf{p}}_d) \tag{11.209}$$

By using (11.161), $\mathbf{L}^\top \mathbf{L} = \mathbf{I}_{2\times 2}$ and $\dot{\mathbf{x}} = \mathbf{z}_2 + \dot{\mathbf{x}}_r$ it is seen that:

$$\begin{aligned} \dot{V}_p &= \tilde{\mathbf{p}}^\top[\mathbf{L}^\top(\mathbf{R}(\gamma)\mathbf{H}(\rho)\dot{\mathbf{x}} + \mathbf{L}\dot{\mathbf{p}}_0) - \dot{\mathbf{p}}_d] \\ &= \tilde{\mathbf{p}}^\top(\dot{\mathbf{p}}_0 - \dot{\mathbf{p}}_d + \mathbf{L}^\top \mathbf{R}(\gamma)\mathbf{H}(\rho)\dot{\mathbf{x}}_r) + \tilde{\mathbf{p}}^\top \mathbf{L}^\top \mathbf{R}(\gamma)\mathbf{H}(\rho)\mathbf{z}_2 \end{aligned} \tag{11.210}$$

Now, by choosing the circle center update law as:

$$\dot{\mathbf{p}}_0 = \dot{\mathbf{p}}_d - \mathbf{L}^\top \mathbf{R}(\gamma)\mathbf{H}(\rho)\dot{\mathbf{x}}_r - k_0 \tilde{\mathbf{p}} \tag{11.211}$$

where $k_o > 0$, it is seen that:

$$\dot{V}_p = -k_o \tilde{\mathbf{p}}^\top \tilde{\mathbf{p}} + \tilde{\mathbf{p}}^\top \mathbf{L}^\top \mathbf{R}(\gamma)\mathbf{H}(\rho)\mathbf{z}_2 \tag{11.212}$$

In (11.212) a cross term in $\tilde{\mathbf{p}}$ and $\mathbf{z}_2$ is noticed. In order to guarantee that the time derivative of the total system $V_{\text{wopc}} = V_{3r} + V_p$ is negative semi-definite, the weather optimal controller (11.190) must be modified such that the cross term in (11.212) is cancelled.

Weather Optimal Position Control (WOPC)

The cross-terms involving $\tilde{\mathbf{p}}$ and $\mathbf{z}_2$ in $\dot{V}_p$ can be removed by modifying the nonlinear controller (11.190) to:

$$\begin{aligned}\tau &= \mathbf{T}^\top(\mathbf{M}_x\ddot{\mathbf{x}}_r + \mathbf{C}_x\dot{\mathbf{x}}_r + \mathbf{D}_x\dot{\mathbf{x}}_r - \mathbf{K}_p\mathbf{z}_1 - \mathbf{K}_d\mathbf{z}_2) - \mathbf{q}(\cdot) - \phi\hat{F}_e \\ &\quad -\mathbf{T}^\top\mathbf{E}^\top(\rho)\mathbf{R}^\top(\gamma)\mathbf{L}\tilde{\mathbf{p}} \end{aligned} \tag{11.213}$$

The last term in τ implies that:

$$\dot{V}_{3r} = -\mathbf{z}_{1r}^\top(\mathbf{E}\mathbf{K}_p\mathbf{E}^\top)(\mathbf{E}\Lambda\mathbf{E}^\top)\mathbf{z}_{1r} - \mathbf{z}_2^\top(\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 - \tilde{\mathbf{p}}^\top\mathbf{L}^\top\mathbf{R}(\gamma)\mathbf{H}(\rho)\mathbf{z}_2 \tag{11.214}$$

Consider:

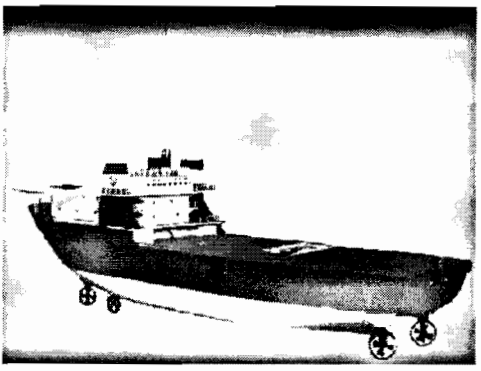
$$V_{\text{wopc}} = V_{3r} + V_p \tag{11.215}$$

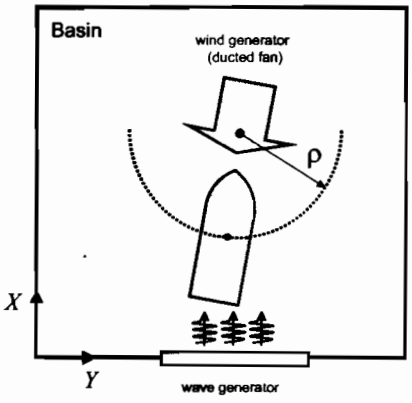
$$\dot{V}_{\text{wopc}} = -\mathbf{z}_{1r}^\top(\mathbf{E}\mathbf{K}_p\mathbf{E}^\top)(\mathbf{E}\Lambda\mathbf{E}^\top)\mathbf{z}_{1r} - \mathbf{z}_2^\top(\mathbf{K}_d + \mathbf{D}_x)\mathbf{z}_2 - k_o\tilde{\mathbf{p}}^\top\tilde{\mathbf{p}} \tag{11.216}$$

and therefore the equilibrium point $(\mathbf{z}_{1r}, \mathbf{z}_2, \tilde{F}_e, \tilde{\mathbf{p}}) = (\mathbf{0}, \mathbf{0}, 0, \mathbf{0})$ is ULES.

The term $\ddot{\mathbf{p}}_0$ is needed in the expression for $\mathbf{q}(\cdot)$. This term is computed from (11.211) as:

$$\begin{aligned}\ddot{\mathbf{p}}_0 &= \ddot{\mathbf{p}}_d - k_o(\dot{\mathbf{p}} - \dot{\mathbf{p}}_d) - \mathbf{L}^\top\mathbf{R}(\gamma)\mathbf{H}(\rho)\ddot{\mathbf{x}}_r \\ &\quad -\mathbf{L}^\top\dot{\mathbf{R}}(\gamma)\mathbf{H}(\rho)\dot{\mathbf{x}}_r - \mathbf{L}^\top\mathbf{R}(\gamma)\dot{\mathbf{H}}(\rho)\dot{\mathbf{x}}_r \end{aligned} \tag{11.217}$$





11.4.4 Experiments and Simulations

The proposed weather optimal positioning control system has been implemented and tested at the GNC Laboratory located at the Norwegian University of Science and Technology (NTNU). In the experiments Cybership I was used. This is a model ship of scale 1:70. A ducted fan is used to generate a slowly-varying or constant wind disturbance.

The length of the model ship is $L_m = 1.19$ (m) and the mass is $m_m = 17.6$ (kg). The experimental results are scaled to full scale by considering a supply vessel with mass $m_s = 4500$ (tons) using the Bis system; see Section 8.1.3.

Experiment 1: Weather Optimal Heading Control (WOHC)

In the first experiment the ship was allowed to move on the circle arc (the circle center controller (11.211) was turned off, that is $n_0 = \text{constant}$ and $e_0 = \text{constant}$). This is referred to as WOHC. The fixed origin and circle arc are shown in Figure 11.16. Notice that the initial heading is approximately 30 degrees, see Figure 11.17, while the position $(n, e) \approx (13, -43)$. These values are the one obtained when the fan was initially directed in 210 degrees (opposite direction of the ship heading).

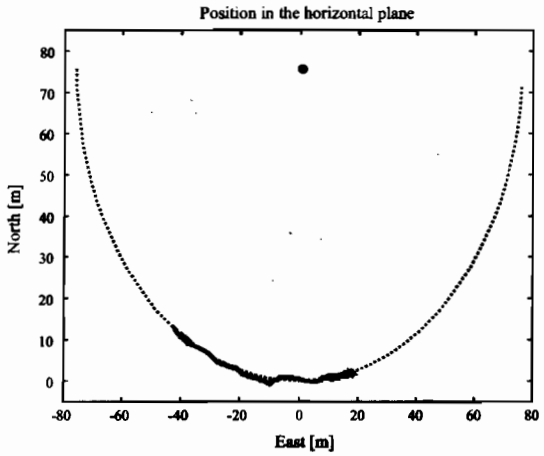


Figure 11.16: WOHC experiment showing the circular motion of the ship when the circle center controller is turned off (WOHC).

After 3000 seconds the fan was slowly rotated to 165 degrees corresponding to a weather optimal heading of -15 degrees, see Figure 11.17. During this process, the ship starts to move on the circle arc (with heading towards the circle center) until it is stabilized to its new heading, that is -15 degrees. The new position on the circle arc is $(n, e) \approx (3, 20)$. This clearly demonstrates that the ship heading converges to the optimal value (copies the dynamics of a pendulum in the gravity field). This is done without using any external wind sensor.

In the next experiment, we will show how the circle center can be translated on-line in order to obtain a constant position (n, e).

Experiment 2: Weather Optimal Position Control (WOPC)

In the second experiment the ship should maintain its position (the circle center controller (11.211) is activated). The performance during station-keeping (dynamic positioning) and translation of the circle is shown in Figure 11.18. The position controller works within an accuracy of ± 1 m which is the accuracy of the DGPS system.

Again the weather optimal heading is changed from approximately 23 degrees to 2 degrees but this time without changing the position (n, e) of the ship. The position deviations and the weather optimal heading are shown in Figure 11.19. These values are obtained by moving the fan from an initial angle of 203 degrees to 182 degrees.

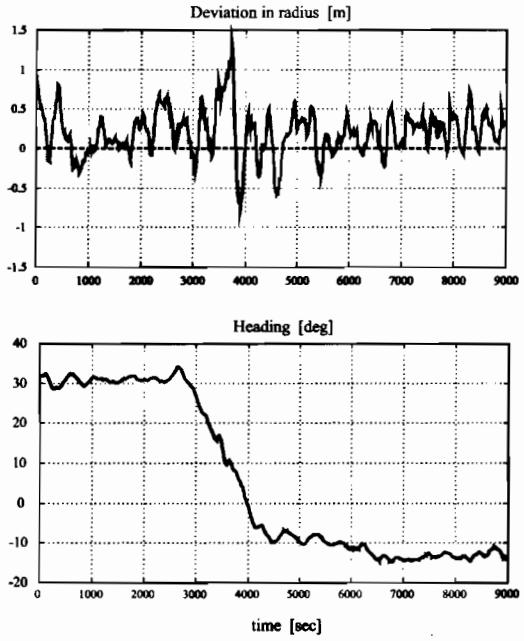


Figure 11.17: WOHC experiment showing the performance of the radius regulator (upper plot) and weather optimal heading (lower plot) versus time (s).

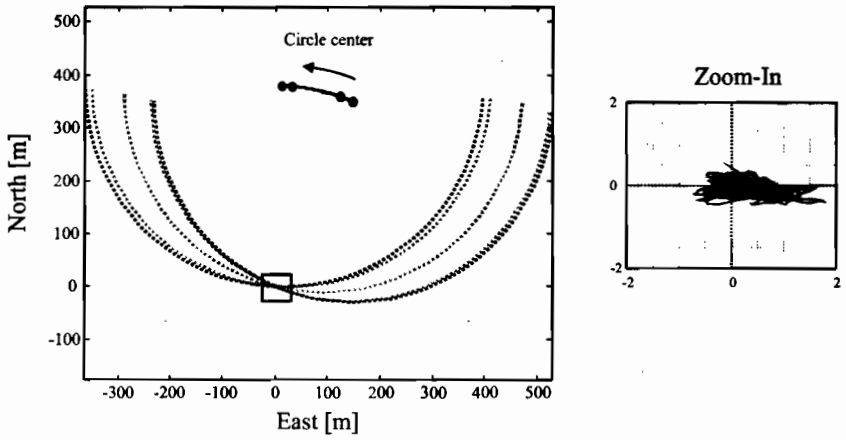


Figure 11.18: WOPC experiment showing how the circle center is moved to obtain station-keeping to $(n_d, e_d) = (0, 0)$.

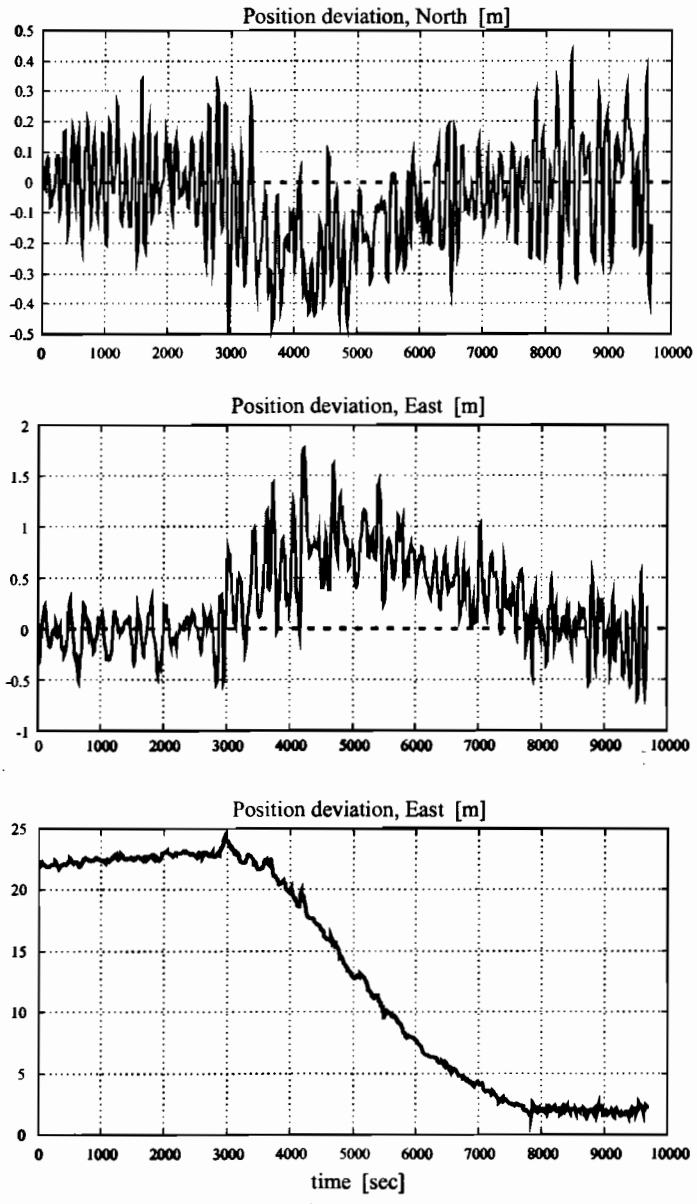


Figure 11.19: WOPC experiment showing the North and East position accuracies (upper plots) and weather optimal heading (lower plot) versus time (seconds). The position accuracy is within ± 1 m while the heading changes from 23 degrees to 2 degrees as the fan is rotated.

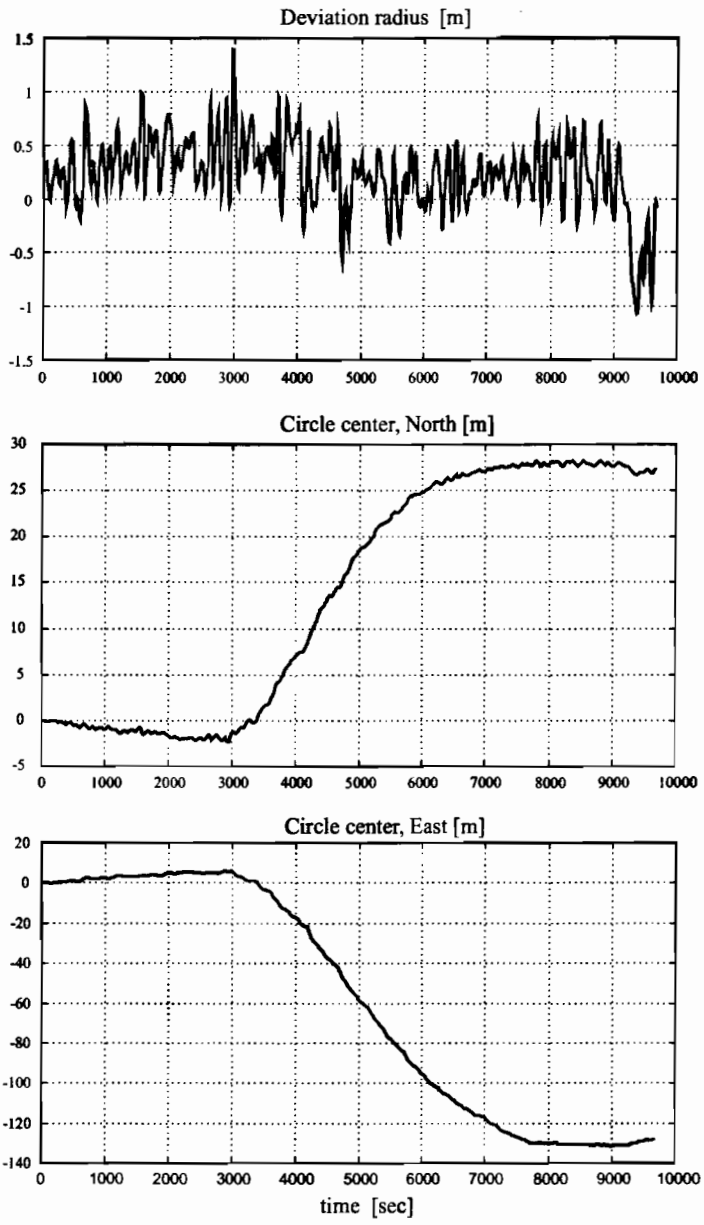


Figure 11.20: WOPC experiment showing the deviation for the radius regulator (upper plot) and the translation of the circle center (n_0, e_0) (lower plots) versus time in seconds. The radius deviation is within ± 1 m during the rotation of the fan.

11.5 Exercises

Exercise 11.1 *Compute the generalized inverse of the actuator configuration matrix* $\mathbf{T}(\boldsymbol{\alpha})$ *in the Examples 11.2 and 11.3 using* $l_1 = 30$ *(m),* $l_2 = 20$ *(m),* $l_3 = 15$ *(m),* $l_4 = 20$ *(m) and* $l_5 = l_6 = 5$ *(m) with* $\mathbf{W} = \mathbf{I}$*. Plot the determinant of* $\mathbf{T}(\boldsymbol{\alpha})\mathbf{T}^\top(\boldsymbol{\alpha})$ *as function of* α_1 *and* α_2 *and find the maximum value (energy optimal azimuth angles). Change the pricing on the actuators to* $\mathbf{W} = \text{diag}\{1, 1, 1, 1, 100, 100\}$ *and plot the determinants of* $\mathbf{T}(\boldsymbol{\alpha})\mathbf{T}^\top(\boldsymbol{\alpha})$ *and* $\mathbf{T}(\boldsymbol{\alpha})\mathbf{W}^{-1}\mathbf{T}^\top(\boldsymbol{\alpha})$ *together. Comment on the results.*

Exercise 11.2 *Use the Matlab GNC toolbox to simulate the nonlinear PID controller in Section 11.2.3 when applied to the supply vessel in the Simulink library. Tune the controller gains such that the vessel can handle both waves and wind disturbances in all directions. Present an xy-plot showing the performance of the DP system when exposed to different disturbances.*

Part IV

Underwater Vehicle Applications

Chapter 12

Propeller Control System Design

This chapter discusses model-based observer and control design techniques for underwater vehicle propellers. Lyapunov analysis is used as the main tool in deriving the structure of the output feedback controller.

12.1 Models for Propeller Shaft Speed and Motors

One-, two- and three-state dynamic models for propeller shaft speed are presented for use in voltage, armature current, and torque controlled DC-motors.

12.1.1 Propeller Shaft Speed Models

In Yoerger *et al.* (1991) a *one-state model* for propeller shaft speed n with propeller thrust T as output was proposed. This model can be written:

$$J_m \dot{n} + K_{n|n|} n\,|n| = \tau \tag{12.1}$$

$$T = T(n, u_p) \tag{12.2}$$

where n is the shaft speed, u_p is the axial flow velocity in the propeller disc, and τ is the control input (shaft torque); see Figure 12.1. It is common to assume $u_p = 0$ when computing T. However, u_p can be measured by using a laser-Doppler velocimeter (LDV) system, a particle image velocimeter (PIV) system or an acoustic Doppler velocimeter system for instance. In Section 12.3 a state observer for reconstruction of u_p is designed by treating u_p as an unmeasured state.

Healey *et al.* (1995) have modified the models (12.1)–(12.2) to describe overshoots in thrust, which are typical in experimental data. Based on the results of Cody (1992) and

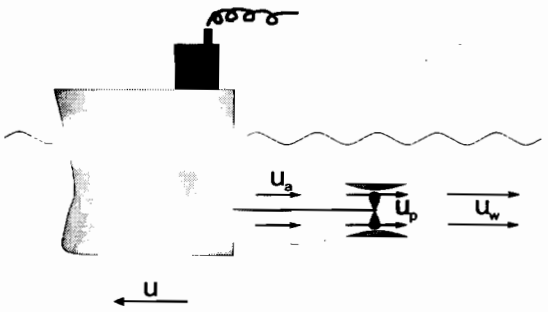


Figure 12.1: Definitions of axial flow velocity u_p, advance speed u_a and vehicle speed u.

McLean (1991), Healey and co-workers proposed a *two-state model*:

$$J_m\dot{n} + K_n n = \tau - Q \tag{12.3}$$

$$m_f\dot{u}_p + d_f(u_p - u)\,|u_p - u| = T \tag{12.4}$$

$$T = T(n, u_p) \tag{12.5}$$

$$Q = Q(n, u_p) \tag{12.6}$$

where u is the forward speed of the vehicle and Q is the propeller torque. This was done by modelling a control volume of water around the propeller as a mass-damper system. The mass-damper of the control volume interacts with the vehicle speed dynamics, which are also representd by a mass-damper system. Experimental verifications of the one- and two-state models are found in Whitcomb and Yoerger (1999*a*).

A more general model is the *three-state* propeller shaft speed model (Blanke *et al.* 2000*b*):

$$J_m\dot{n} + K_n n = \tau - Q \tag{12.7}$$

$$m_f\dot{u}_p + d_{f0}u_p + d_f\,|u_p|\,(u_p - u_a) = T \tag{12.8}$$

$$(m - X_{\dot{u}})\dot{u} - X_u u - X_{u|u|}u\,|u| = (1-t)T \tag{12.9}$$

$$T = T(n, u_p) \tag{12.10}$$

$$Q = Q(n, u_p) \tag{12.11}$$

where damping in surge is modelled as the sum of *linear laminar skin friction*, $-X_u u$ (Faltinsen and Sortland 1987) and *nonlinear quadratic drag*, $-X_{u|u|}u\,|u|$ (Faltinsen 1990). Similarly, linear damping, $d_{f0}u_p$, is included in the axial flow, model since quadratic damping, $d_f\,|u_p|\,u_p$, alone would give an unrealistic response at low speeds. Linear skin friction gives exponential convergence to zero at low speeds.

The ambient water velocity u_a in (12.8) is computed by using the steady-state condition:

$$u_a = (1-w)u \tag{12.12}$$

where $0 < w < 1$ is the *wake fraction number* (Lewis 1989).

Table 12.1: DC-motor control model.

	control input τ	Linear damping K_n
Voltage	$\frac{K_m}{R_a} V_m$	$\frac{K_m^2}{R_a}$
Current	$\frac{K_m K_p}{R_a + K_p} i_d$	$\frac{K_m^2}{R_a + K_p}$
Torque	$\frac{K_p}{R_a + K_p} Q_d$	$\frac{K_m^2}{R_a + K_p}$

12.1.2 Unified Representation of DC-Motor Controllers

The model (12.7) with parameter K_n and control input τ can be used to represent the following motor controllers:

- *Motor (armature) voltage control*
- *Motor (armature) current control*
- *Motor torque control*

The different DC-motor control strategies are obtained by choosing K_n and τ according to Table 12.1 (Fossen and Blanke 2000).

This result is obtained by considering a DC-motor:

$$L_a \frac{d}{dt} i_m = -R_a i_m - K_m \omega_m + V_m \tag{12.13}$$

$$J_m \dot{n} = K_m i_m - Q \tag{12.14}$$

where V_m is the armature voltage, i_m is the armature current, n is the propeller revolution and Q is the load from the propeller. In addition, L_a is the armature inductance, R_a is the armature resistance, K_m is the motor torque constant and J_m is the rotor moment of inertia.

Since the electrical time constant $T_a = L_a/R_a$ is small compared to the mechanical time constant, time scale separation suggests:

$$\frac{L_a}{R_a} \frac{d}{dt} i_m \approx 0 \tag{12.15}$$

Hence, the shaft speed dynamics is given by:

$$0 = -R_a i_m - K_m n + V_m \tag{12.16}$$

$$J_m \dot{n} = K_m i_m - Q \tag{12.17}$$

Motor Current Control

The motor current can be controlled by using a P-controller:

$$V_m = K_p (i_d - i_m), \quad K_p > 0 \tag{12.18}$$

where i_d is the desired motor current. From (12.16) we get:

$$(R_a + K_p)i_m = -K_m n + K_p i_d \tag{12.19}$$

The motor dynamics (12.17) for the current controlled motor therefore takes the form:

$$J_m \dot{n} + \frac{K_m^2}{R_a + K_p} n = \frac{K_m K_p}{R_a + K_p} i_d - Q \tag{12.20}$$

If a high gain controller $K_p \gg R_a > 0$ is used, this expression simplifies to (see Table 12.1):

$$J_m \dot{n} = K_m i_d - Q \tag{12.21}$$

Motor Torque Control

For a DC motor, the motor torque will be proportional to the motor current. Hence, the desired motor torque Q_d can be written as:

$$Q_d = K_m i_d \tag{12.22}$$

From (12.20) we see that this yields the following dynamics for a torque controlled motor:

$$J_m \dot{n} + \frac{K_m^2}{R_a + K_p} n = \frac{K_p}{R_a + K_p} Q_d - Q \tag{12.23}$$

which reduces to:

$$J_m \dot{n} = Q_d - Q \tag{12.24}$$

for $K_p \gg R_a > 0$ (see Table 12.1).

Motor Voltage Control

Finally, a motor voltage control is obtained by combining (12.16)–(12.17) to give:

$$J_m \dot{n} + \frac{K_m^2}{R_a} n = \frac{K_m}{R_a} V_m - Q \tag{12.25}$$

Unified Representation of DC-Motor Controllers

Based on the three models presented above, a unified control model for the DC-motor shaft speed dynamics can be written:

$$J_m \dot{n} + K_n n = \tau - Q \tag{12.26}$$

where motor *voltage*, *current* and *torque* control are obtained by choosing the control input τ and linear damping coefficient K_n according to Table 12.1:

12.1.3 Propeller Losses

When designing an UUV control system, commanded forces and moments must be realized by a propeller control system using a mapping from thrust demand to propeller revolution. This is a non-trivial task since a propeller in water suffers several phenomena that cause thrust losses. The primaries are:

Axial Water Inflow: Propeller losses caused by axial water inflow, that is the speed u_p of the water going into the propeller. The axial flow velocity will in general differ from the speed of the vehicle. The dynamics of the propeller axial flow is usually neglected when designing the propeller shaft speed controller. This leads to thrust degradation since the computed thruster force is a function of both the propeller shaft speed and axial flow. The magnitude of the axial flow velocity will strongly influence the thrust at high speed so it is crucial for the propeller performance.

Other effects that will reduce the propeller thrust are described in Sørensen *et al.* (1997) and references therein. Some of these effects are:

Cross-Coupling Drag: Water inflow perpendicular to the propeller axis caused by current, vessel speed or jets from other thrusters. This will introduce a force in the direction of the inflow due to deflection of the propeller race.

Air Suction: For heavily loaded propellers, ventilation (air suction) caused by decreasing pressure on the propeller blades may occur, especially when the submergence of the propeller becomes small due to the vessel's wave frequency motion.

In-and-out-of Water Effects: For extreme conditions with large vessel motions the *in-and-out-of water effects* will result in a sudden drop of thrust and torque following a hysteresis pattern.

Thruster Hull Interaction: Thrust reduction and change of thrust direction may occur due to thruster-hull interaction caused by frictional losses and pressure effects when the thruster race sweeps along the hull. The latter is the *Coanda effect;* see Faltinsen (1990), pp. 270–272.

Thruster-Thruster Interaction: Thruster-thruster interaction caused by influence from the propeller race from one thruster on neighboring thrusters may lead to significant thrust reduction.

12.2 Propeller Thrust and Torque Modelling

For a fixed pitch propeller the shaft torque Q and force (thrust) T depend on the forward speed u of the vessel, the advance speed u_a (ambient water speed) and the propeller rate n; see Figure 12.1. In addition, other dynamic effects due to *unsteady flows* will influence the propeller thrust and torque. According to Newman (1977), Breslin and Andersen (1994), and Carlton (1994) the following unsteady flow effects are significant:

- *air suction*
- *cavitation*
- *in-and-out-of-water effects (Wagner's effect)*
- *wave influenced boundary layer effect*
- *Kuessner effect (gust)*

For a deeply submerged vehicle, the first four effects above can be neglected. The *Kuessner effect,* which is caused by a propeller in gust, will appear as a rapid oscillating thrust component. These fluctuations are usually small compared to the total thrust in a dynamical situation. Under these assumptions, the thrust and torque models can be modelled using a *quasi-steady* representation.

Unsteady modelling is, however, an important topic for future research since unsteady flow effects are significant in many practical situations, particularly for surface vessels. A more detailed discussion on the accuracy of unsteady and quasi-steady modelling is found in Breslin and Andersen (1994), pp. 374–386.

12.2.1 Quasi-Steady Thrust and Torque

Quasi-steady modelling of thrust and torque is usually done in terms of *lift* and *drag* curves which are transformed to thrust and torque by using the angle of incidence. This approach has been used by Healey *et al.* (1995) and Whitcomb and Yoerger (1999*a*) for instance.

The lift and drag are usually represented as *non-dimensional* thrust and torque coefficients computed from self-propulsion tests; see Fossen (1994) or Lewis (1989). The non-dimensional thrust and torque coefficients K_T and K_Q are computed by measuring T, Q, and n. Consequently:

$$K_T(J_0) = \frac{T}{\rho D^4 n\,|n|}, \quad K_Q(J_0) = \frac{Q}{\rho D^5 n\,|n|} \tag{12.27}$$

where D is the propeller diameter, ρ is the water density, and:

$$J_0 = \frac{u_a}{nD} \tag{12.28}$$

is the *advance ratio.* The numerical expressions for K_T and K_Q are found by *open water* tests, usually performed in a cavitation tunnel or a towing tank. These tests neglect the unsteady flow effects, so steady-state values of T, Q, and n are used.

The non-dimensional thrust and torque coefficients can also be described by the following parameters; see Oosterveld and Oossanen (1975):

$$K_T = f_1\left(J_0, \frac{P}{D}, \frac{A_E}{A_o}, Z\right) \tag{12.29}$$

$$K_Q = f_2\left(J_0, \frac{P}{D}, \frac{A_E}{A_o}, Z, R_n, \frac{t}{c}\right) \tag{12.30}$$

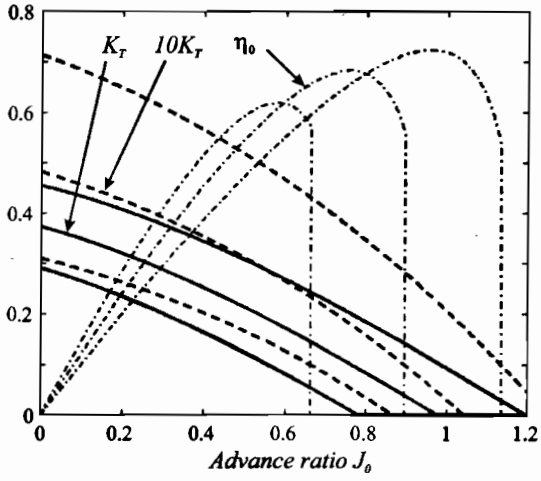


Figure 12.2: Open water K_T (solid), $10 \cdot K_Q$ (dash) and η_o (dash-dot) as a function of advance ratio J_0 for $P/D = 0.7$, 0.89 and 1.1. Reconstructed from data in [24].

where P/D is the pitch ratio, A_E/A_0 is the expanded-area ratio, Z is the number of blades, R_n is the *Reynolds number*, t is the maximum thickness of the blade section, and c is the chord length of the blade section.

From (12.27) the thrust T and torque Q can be written

$$T = \rho D^4 K_T(J_0) n |n| \tag{12.31}$$

$$Q = \rho D^5 K_Q(J_0) n |n| \tag{12.32}$$

The open water propeller efficiency in undisturbed water is given as the ratio of the work done by the propeller in producing a thrust force to the work required to overcome the shaft torque according to:

$$\eta_o = \frac{u_p T}{2\pi n Q} = \frac{J_0}{2\pi} \cdot \frac{K_T}{K_Q} \tag{12.33}$$

The K_T, K_Q, and η_o curves for different pitch ratios for a *Wageningen B-screw series* based on Table 5 in Oosterveld and Oossanen (1975), with $R_n = 2 \cdot 10^6$, $Z = 4$, $D = 3.1$ m, and $A_E/A_o = 0.52$ are shown in Figure 12.2.

For simplicity we will consider an underwater vehicle where K_T and K_Q show a linear behavior in J_0. Experimental data for K_T and K_Q are shown in Figure 12.3. The linear approximations are:

$$K_T = \alpha_1 J_0 + \alpha_2 \tag{12.34}$$

$$K_Q = \beta_1 J_0 + \beta_2 \tag{12.35}$$

where α_i and β_i $(i = 1, 2)$ are four non-dimensional constants. It should be noted that nonlinear functions for K_T and K_Q can also be used. This is equivalent to the *lint theory* result (Blanke 1981).

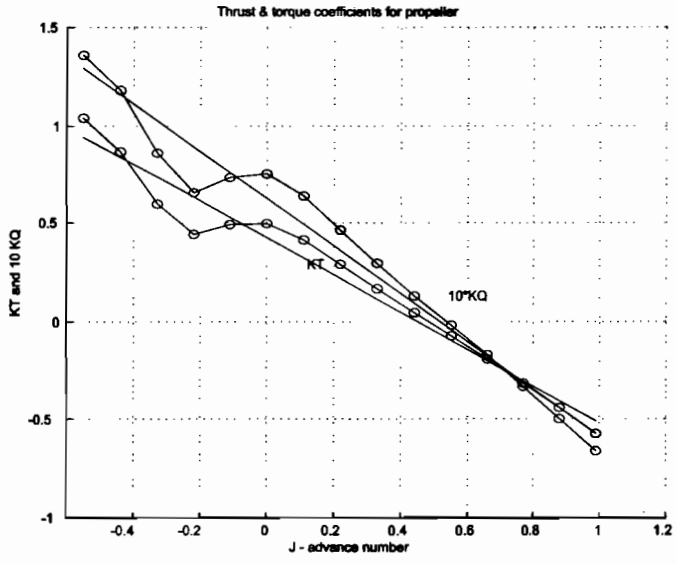


Figure 12.3: Experimental results for K_T and $10K_Q$ versus J_0 (circles) and least-squares fits to a straight line (solid lines), see Fossen and Blanke (2000).

Formulas (12.34)–(12.35) imply that the mathematical expressions for Q and T can be written as; see Fossen (1994), pp. 94–97:

$$T = T_{n|n|} n\,|n| - T_{|n|u_a}\,|n|\,u_a \tag{12.36}$$

$$Q = Q_{n|n|} n\,|n| - Q_{|n|u_a}\,|n|\,u_a \tag{12.37}$$

where

$$\begin{array}{ll} Q_{n|n|} = \rho D^5 \beta_2 & T_{n|n|} = \rho D^4 \alpha_2 \\ Q_{|n|u_a} = \rho D^4 \beta_1 & T_{|n|u_a} = \rho D^3 \alpha_1 \end{array} \tag{12.38}$$

are *positive* propeller coefficients given by the propeller characteristics. Notice that T and Q are defined for all n even though J_0 is undefined for $n = 0$. This is important since the observer-controller will be based on the expressions for T and Q.

The coefficient $T_{|n|u_a}$ is derived from steady state, where u_a has achieved its final value. To explicitly account for the *dynamic variation* in axial water speed, the thrust (12.36) is modified according to (Blanke *et al.* 2000*b*):

$$\begin{aligned} T &= T_{|n|n}|n|n - T^o_{|n|u_a}\,|n|\,u_a - T^o_{|n|u_a}\,|n|\,(u_p - u_a) \\ &= T_{|n|n}|n|n - T^o_{|n|u_a}\,|n|\,u_p \end{aligned} \tag{12.39}$$

where $T^o_{|n|u_a}$ is defined in terms of the steady-state ratio $\bar{u}_a/\bar{u}_p$, that is:

$$T^o_{|n|u_a} := \frac{\bar{u}_a}{\bar{u}_p} T_{|n|u_a} = \frac{1}{1+a} T_{|n|u_a} > 0 \tag{12.40}$$

Here $a > 0$ is an axial flow parameter; see Lewis (1989), pp.131–132. Similarly, the torque mapping (12.37) is modified to:

$$Q = Q_{|n|n}|n|n - Q^{\circ}_{|n|u_a}|n|\,u_p \tag{12.41}$$

12.3 Nonlinear Observer for Estimation of Propeller Axial Velocity

A nonlinear observer for shaft speed estimation and fault detection has been proposed by Blanke *et al.* (1998) under the assumption that the effect of the propeller axial inlet flow u_p can be neglected. They have proven semiglobal asymptotic and exponential stability for the case with quadratic damping.

Nonlinear observers for underwater vehicles can also be designed by using contraction analysis as described by Lohmiller (1999), and Lohmiller and Slotine (1996, 1998). This can be related to the work of Lewis (1951) where it is shown that the *Riemann* metric can be used as a tool for contraction analysis of nonautonomous nonlinear systems. Applications to underwater vehicles are found in Jouffroy and Lottin (2003).

In this chapter a Lyapunov-based output feedback controller is constructed (Fossen and Blanke 2000). Global exponential stability properties, which are important from a robust performance point of view, are also proven. For a more detailed discussion on nonlinear observer-controller design; see Nijmeijer and Fossen (1999) and references therein.

12.3.1 Vehicle Speed and Propeller Axial Flow Dynamics

Consider an UUV moving in surge (x-direction) equipped with one single propeller aft of the hull. Let u (positive forwards) denote the forward speed of the UUV. The surge dynamics is assumed to be coupled to the *axial flow velocity* u_p of the propeller (positive backwards) according to (Fossen and Blanke 2000):

$$(m - X_{\dot{u}})\dot{u} - X_u u - X_{u|u|}u\,|u| = (1-t)T \tag{12.42}$$

$$m_f \dot{u}_p + d_{f0}u_p + d_f\,|u_p|\,(u_p - u_a) = T \tag{12.43}$$

where $m_f > 0$, $d_{f0} > 0$, and $d_f > 0$.

For the vessel dynamics (12.42) in surge; $m - X_{\dot{u}} > 0$ is the mass of the vessel including hydrodynamic added mass, $-X_u u - X_{u|u|}u\,|u| \geq 0$ is damping due to *linear skin friction* and *quadratic drag*, and $t > 0$ is the thrust deduction number (typically 0.05–0.2) due to propeller-hull interactions. Notice that linear damping, $-X_u u$, ensures that u converges exponentially to zero for $T = 0$.

The dynamics of the water (12.43), represents a nonlinear *"hydrodynamic" mass-damper* for $u = 0$. For a vessel moving at positive cruise-speed in steady flow, $u =$ constant and $u_a =$ constant. The relationship between ambient water velocity and vehicle speed in steady-state is (Lewis 1989):

$$u_a = (1 - w)u \tag{12.44}$$

where $w > 0$ (typically 0.1–0.4) is denoted as the *wake fraction number.*

When designing the nonlinear observer in Section 12.3, the following assumption for the nonlinear term $|u_p|$ is needed:

Assumption A1: Under normal operation of the vehicle it is assumed that axial flow velocity u_p and propeller revolution n have the same signs. This implies that:

$$|u_p| = \text{sgn}(u_p)u_p \approx \text{sgn}(n)u_p \tag{12.45}$$

Under Assumption A1, (12.39) and (12.42)–(12.44) can be combined to give:

$$\begin{bmatrix} m - X_{\dot{u}} & 0 \\ 0 & m_f \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{u}_p \end{bmatrix} + \begin{bmatrix} -X_u - X_{u|u|}\,|u| & 0 \\ 0 & d_{f0} + d_f\,|u_p| - d_f(1-w)\text{sgn}(n)u \end{bmatrix} \begin{bmatrix} u \\ u_p \end{bmatrix} + |n| \begin{bmatrix} 0 & (1-t)T^o_{|n|u_a} \\ 0 & T^o_{|n|u_a} \end{bmatrix} \begin{bmatrix} u \\ u_p \end{bmatrix} = \begin{bmatrix} (1-t)T_{n|n|}n\,|n| \\ T_{n|n|}n\,|n| \end{bmatrix}$$

The measurement equation is:

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ u_p \end{bmatrix} \tag{12.46}$$

This can be written in state-space form according to:

$$\mathbf{H}\dot{\mathbf{x}} + \mathbf{D}_0\mathbf{x} + \mathbf{D}_1(\mathbf{x}, n, y)\mathbf{x} + |n|\,\mathbf{E}\mathbf{x} = \mathbf{f}(n) \tag{12.47}$$

$$y = \mathbf{h}^T\mathbf{x} \tag{12.48}$$

where $\mathbf{x} = [u, u_p]^\top$, $y = u$ and:

$$\mathbf{H} = \begin{bmatrix} m - X_{\dot{u}} & 0 \\ 0 & m_f \end{bmatrix}, \qquad \mathbf{f}(n) = \begin{bmatrix} (1-t)T_{n|n|}n\,|n| \\ T_{n|n|}n\,|n| \end{bmatrix}$$

$$\mathbf{D}_0 = \begin{bmatrix} -X_u & 0 \\ 0 & d_{f0} \end{bmatrix}, \quad \mathbf{D}_1(\mathbf{x}, n, y) = \begin{bmatrix} -X_{u|u|}\,|u| & 0 \\ 0 & d_f\,|u_p| - d_f(1-w)\text{sgn}(n)y \end{bmatrix}$$

$$\mathbf{E} = \begin{bmatrix} 0 & (1-t)T^o_{|n|u_a} \\ 0 & T^o_{|n|u_a} \end{bmatrix}, \qquad \mathbf{h}^\top = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

12.3.2 Observer Equations

In Fossen and Blanke (2000) a nonlinear state observer for the unmeasured state u_p is derived by copying the dynamics (12.47)–(12.48):

$$\mathbf{H}\dot{\hat{\mathbf{x}}} + \mathbf{D}_0\hat{\mathbf{x}} + \mathbf{D}_1(\hat{\mathbf{x}}, n, y)\hat{\mathbf{x}} + |n|\,\mathbf{E}\hat{\mathbf{x}} = \mathbf{f}(n) + \mathbf{k}(n)\tilde{y} \tag{12.49}$$

$$\hat{y} = \mathbf{h}^\top\hat{\mathbf{x}} \tag{12.50}$$

where $\tilde{y} = y - \hat{y}$. The observer gain vector is chosen as:

$$\mathbf{k}(n) = \begin{bmatrix} K_{10} \\ K_{20} \end{bmatrix} + |n| \begin{bmatrix} K_{11} \\ K_{21} \end{bmatrix} \tag{12.51}$$

The error dynamics corresponding to $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ becomes:

$$\mathbf{H}\dot{\tilde{\mathbf{x}}} = -\left(\mathbf{D}_0 + |n|\,\mathbf{E} + \mathbf{k}(n)\mathbf{h}^\top\right)\tilde{\mathbf{x}} - \boldsymbol{\delta} \tag{12.52}$$

where the nonlinear estimation error term $\boldsymbol{\delta}$ is:

$$\boldsymbol{\delta} := \mathbf{D}_1(\mathbf{x}, n, y)\mathbf{x} - \mathbf{D}_1(\hat{\mathbf{x}}, n, y)\hat{\mathbf{x}} = \begin{bmatrix} (-X_{u|u|})\,|u|\,u - (-X_{u|u|})\,|\hat{u}|\,\hat{u} \\ d_f\,|u_p|\,u_p - d_f\,|\hat{u}_p|\,\hat{u}_p \end{bmatrix} \tag{12.53}$$

Defining:

$$\mathbf{F} = \begin{bmatrix} -X_u + K_{10} & 0 \\ K_{20} & d_{f0} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} K_{11} & (1-t)T^o_{|n|u_a} \\ K_{21} & T^o_{|n|u_a} \end{bmatrix} \tag{12.54}$$

implies that (12.52) can be written:

$$\mathbf{H}\dot{\tilde{\mathbf{x}}} = -\mathbf{F}\tilde{\mathbf{x}} - |n|\,\mathbf{G}\tilde{\mathbf{x}} - \boldsymbol{\delta} \tag{12.55}$$

Lyapunov analyis will now be used to derive criteria on the observers in $\mathbf{k}(n)$ such that the equilibrium point $\tilde{\mathbf{x}} = \mathbf{0}$ is UGES. An alternative approach using contraction analysis is presented in Jouffroy and Lottin (2003).

12.3.3 Lyapunov Analysis

Consider the Lyapunov function candidate:

$$V_{\text{obs}}(\tilde{\mathbf{x}}, t) = \tilde{\mathbf{x}}^\top\mathbf{H}\tilde{\mathbf{x}} \tag{12.56}$$

$$\dot{V}_{\text{obs}}(\tilde{\mathbf{x}}, t) = -\tilde{\mathbf{x}}^\top\left(\mathbf{F} + \mathbf{F}^\top\right)\tilde{\mathbf{x}} - |n|\,\tilde{\mathbf{x}}^\top\left(\mathbf{G} + \mathbf{G}^\top\right)\tilde{\mathbf{x}} - 2\tilde{\mathbf{x}}^\top\boldsymbol{\delta} \tag{12.57}$$

where the design goal is to choose K_{10}, K_{20}, K_{11}, and K_{21} such that $\dot{V}_{\text{obs}} < 0$ for all $\tilde{\mathbf{x}} \neq \mathbf{0}$.

For a *nondecreasing function* $\mathbf{f}(\mathbf{x})$ it can be shown that:

$$(\mathbf{x} - \hat{\mathbf{x}})^\top\left(\mathbf{f}(\mathbf{x}) - \mathbf{f}(\hat{\mathbf{x}})\right) \geq 0 \tag{12.58}$$

From (12.58) it is seen that the nonlinear coupling term $\tilde{\mathbf{x}}^\top\boldsymbol{\delta} = \tilde{\mathbf{x}}_1\delta_1 + \tilde{\mathbf{x}}_2\delta_2$ in $\dot{V}_{\text{obs}}$ satisfies:

$$\begin{aligned}\tilde{\mathbf{x}}^\top\boldsymbol{\delta} &= (u - \hat{u})\,\delta_1 + (u_p - \hat{u}_p)\delta_2 \\ &= (u - \hat{u})\,(-X_{u|u|})\,(|u|\,u - |\hat{u}|\,\hat{u}) + (u_p - \hat{u}_p)d_f\,(|u_p|\,u_p - |\hat{u}_p|\,\hat{u}_p) \\ &\geq 0\end{aligned} \tag{12.59}$$

since $-X_{u|u|} > 0$ and $d_f > 0$. This is due to the fact that dissipative damping terms like quadratic drag, $u|u|$, and also higher order terms in $u|u|^n$ $(n = 1, 2, 3...)$ are all nondecreasing. Therefore:

$$\dot{V}_{\text{obs}}(\tilde{\mathbf{x}}, t) \leq -\tilde{\mathbf{x}}^\top \left(\mathbf{F} + \mathbf{F}^\top\right) \tilde{\mathbf{x}} - |n| \tilde{\mathbf{x}}^\top \left(\mathbf{G} + \mathbf{G}^\top\right) \tilde{\mathbf{x}} \tag{12.60}$$

Next we notice that the last term in (12.60) is zero if $n = 0$. For non-zero values of n we therefore require that:

$$\mathbf{G} + \mathbf{G}^\top = \begin{bmatrix} 2K_{11} & (1-t)T^o_{|n|u_a} + K_{21} \\ (1-t)T^o_{|n|u_a} + K_{21} & 2T^o_{|n|u_a} \end{bmatrix} > 0 \tag{12.61}$$

which is easy to satisfy since K_{11} and K_{21} can be chosen such that:

$$K_{11} > 0 \tag{12.62}$$

$$4K_{11}T^o_{|n|u_a} > \left((1-t)T^o_{|n|u_a} + K_{21}\right)^2 \tag{12.63}$$

Hence,

$$\dot{V}_{\text{obs}}(\tilde{\mathbf{x}}, t) \leq -\tilde{\mathbf{x}}^\top \left(\mathbf{F} + \mathbf{F}^\top\right) \tilde{\mathbf{x}} \tag{12.64}$$

The remaining two gains K_{10} and K_{20} can be chosen such that:

$$\begin{aligned} \dot{V}_{\text{obs}}(\tilde{\mathbf{x}}, t) &\leq -\tilde{\mathbf{x}}^\top \left(\mathbf{F} + \mathbf{F}^\top\right) \tilde{\mathbf{x}} \\ &\leq -q_1 \tilde{u}^2 - q_2 \tilde{u}_p^2 \\ &< 0, \forall \tilde{u} \neq 0, \quad \tilde{u}_p \neq 0 \end{aligned} \tag{12.65}$$

where $q_1 > 0$ and $q_2 > 0$. In order to prove this the following lemma will be applied:

Lemma 12.1 (Negative Quadratic Form)
The quadratic form:

$$\dot{V} = -\mathbf{x}^\top \mathbf{P} \mathbf{x} \tag{12.66}$$

with $\mathbf{P} = \{p_{ij}\}$ *is bounded by:*

$$\dot{V} \leq -q_1 x_1^2 - q_2 x_2^2 \tag{12.67}$$

with:

$$q_1 = p_{11} - \beta > 0 \tag{12.68}$$

$$q_2 = p_{22} - \frac{(p_{12} + p_{21})^2}{4\beta} > 0, \qquad \beta > 0 \tag{12.69}$$

if:

$$p_{11} > \beta > 0 \tag{12.70}$$

$$p_{22} > \frac{(p_{12} + p_{21})^2}{4\beta} > 0 \tag{12.71}$$

Proof. Expanding $\dot{V}$, yields:

$$\begin{aligned}\dot{V} &= -p_{11}x_1^2 - (p_{12}+p_{21})x_1x_2 - p_{22}x_2^2 \\ &= -(p_{11}-\beta)x_1^2 - \left(\sqrt{\beta}x_1 + \frac{(p_{12}+p_{21})}{2\sqrt{\beta}}x_2\right)^2 - \left(p_{22} - \frac{(p_{12}+p_{21})^2}{4\beta}\right)x_2^2 \\ &\leq -\underbrace{(p_{11}-\beta)}_{q_1}x_1^2 - \underbrace{\left(p_{22} - \frac{(p_{12}+p_{21})^2}{4\beta}\right)}_{q_2}x_2^2 \end{aligned} \tag{12.72}$$

From this it is seen that (12.70)–(12.71) implies that $q_1 > 0$ and $q_2 > 0$ and therefore that $\dot{V} < 0$ for all $x_1 \neq 0$ and $x_2 \neq 0$. ■

Theorem 12.1 (UGES Nonlinear Observer Error Dynamics)
The equilibrium point $\tilde{\mathbf{x}} = \mathbf{0}$ of the observer error dynamics (12.55) is UGES if K_{11} and K_{21} are chosen such that $\mathbf{G} + \mathbf{G}^\top > 0$, that is:

$$K_{11} > 0 \tag{12.73}$$

$$4K_{11}T^o_{|n|u_a} > \left((1-t)T^o_{|n|u_a} + K_{21}\right)^2 \tag{12.74}$$

while K_{10} and K_{20} must satisfy:

$$2d_0\beta > K_{20}^2 \tag{12.75}$$

$$K_{10} - X_u > \frac{1}{2}\beta \tag{12.76}$$

where $\beta > 0$.

Proof. Let

$$\mathbf{P} = \mathbf{F} + \mathbf{F}^\top = \begin{bmatrix} 2(-X_u + K_{10}) & K_{20} \\ K_{20} & 2d_{f0} \end{bmatrix} \tag{12.77}$$

in Lemma 12.1. Hence, $q_1 > 0$ and $q_2 > 0$ if:

$$p_{11} = 2(-X_u + K_{10}) > \beta > 0 \tag{12.78}$$

$$p_{22} = 2d_{f0} > \frac{K_{20}^2}{\beta} > 0 \tag{12.79}$$

It then follows that $\dot{V} \leq -q_1x_1^2 - q_2x_2^2$ with $q_1 > 0$ and $q_2 > 0$. Hence, it follows from Lyapunov stability theory that the equilibrium point $\tilde{\mathbf{x}} = \mathbf{0}$ is UGES if $\beta > 0$. ■

Remark 1
It should be noted that UGES is proven under Assumption A1, that is $|u_p| = sign(u_p)u_p \approx \text{sgn}(n)u_p$. If this assumption is relaxed by using the estimate $|\hat{u}_p|u$ instead of $\text{sgn}(n)\hat{u}_pu$ for the nonlinear coupling term $|u_p|u$, Lyapunov stability analysis can still be used to prove semiglobal exponential stability.

Remark 2
It should be noted that bias state estimation has not been considered when designing the observer. In a practical implementation it might be necessary to augment a constant bias term to the dynamic model in order to improve robustness to unmodelled dynamics and parametric uncertainties.

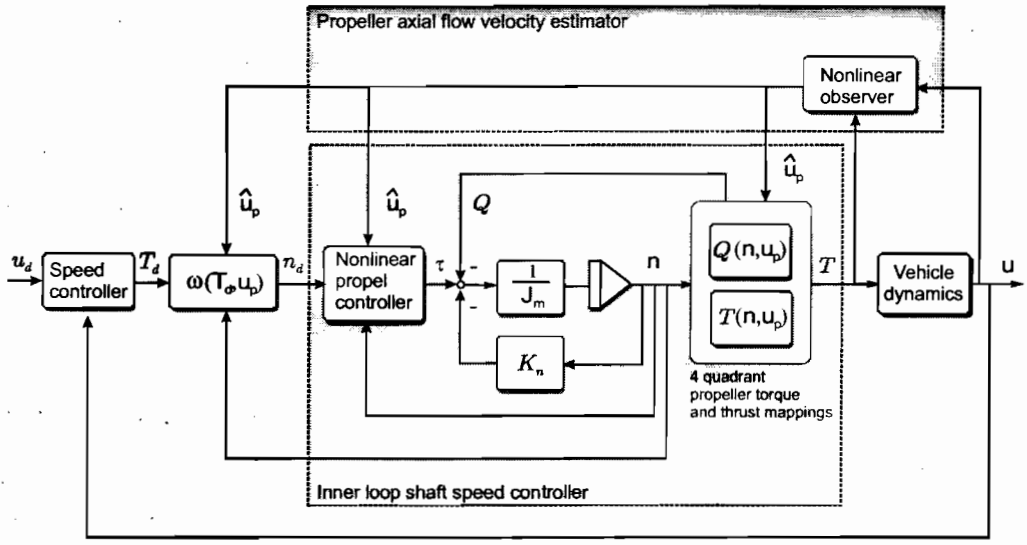


Figure 12.4: Block diagram showing the two control loops.

12.4 Nonlinear Output Feedback Control Design

The control objective is to design a propeller shaft speed controller which tracks the desired propeller revolution n_d (inner loop controller) by compensating the axial flow velocity u_p (Fossen and Blanke 2000). The desired propeller rate of revolution is generated by the UUV speed controller where u_d denotes the desired vessel speed (outer loop controller).

The dynamics of the two control loops can be summarized according to:

UUV Speed Control Loop: The surge dynamics of the UUV is:

$$\dot{x} = \boldsymbol{u} \tag{12.80}$$

$$(m - X_{\dot{u}})\dot{u} - X_u u - X_{u|u|}\boldsymbol{u}\,|u| = (1-t)T \tag{12.81}$$

where T is the control input (force) generated by a speed controller

$$T = T(\dot{u}_d, u_d, u) \tag{12.82}$$

designed such that $u \to u_d$.

Propeller Shaft Speed Control Loop: The desired shaft speed is found from (12.39) as:

$$n_d' = \frac{T^o_{|n|u_a} u_p + \mathrm{sgn}(T_d)\sqrt{\left|(T^o_{|n|u_a} u_p)^2 + 4T_{|n|n}T_d\right|}}{2T_{|n|n}} = \omega(T_d, u_p) \tag{12.83}$$

$$\ddot{n}_d + 2\omega_f \dot{n}_d + \omega_f^2 n_d = \omega_f^2 n_d', \quad \omega_f > 0 \tag{12.84}$$

where T_d is the desired thrust and (12.84) is a 2nd-order low-pass filter with natural frequency ω_f used to generate two smooth reference signal n_d and $\dot{n}_d$. These signals are again used as reference for the propeller controller:

$$\tau = \tau(\dot{n}_d, n_d, n, u_p) \tag{12.85}$$

corresponding to the *two-state* actuator dynamics:

$$\dot{n} = \phi_1(n, Q(n, u_p), \tau) \tag{12.86}$$

$$\dot{V}_a = \phi_2(n, u_p, u) \tag{12.87}$$

$$T = T(n, u_p) \tag{12.88}$$

The two control loops are shown in Figure 12.4 indicating how a nonlinear shaft speed propeller controller together with a conventional UUV speed controller should exploit the estimate of the propeller axial flow velocity $\hat{u}_p$. It is also seen that this is a nonlinear output feedback control problem. One solution to this control problem is to apply *observer backstepping* (Krstic *et al.* 1995). This is the topic for the next section.

Experiments with different shaft speed control strategies and position control of underwater vehicles are reported in Tsukamoto *et al.* (1997) and Whitcomb and Yoerger (1999*b*)..

In this section we will design a nonlinear output feedback propeller controller using only surge speed measurements u and propeller revolution measurements n. The axial flow velocity u_p will be estimated by the state estimator (12.49)–(12.50) which was proven to be UGES under Assumption A1 (Theorem 12.1). The design goal is to render the closed-loop error dynamics of the observer-controller UGES.

12.4.1 Nonlinear Model for Propeller Shaft Speed Control

Consider the unified DC-motor model, se Section 12.1.2:

$$J_m\dot{n} + K_n n = \tau - Q(n, u_p) \tag{12.89}$$

which can be used to describe motor *voltage, current* and *torque* controlled propellers. Substituting the expression for Q given by (12.41) into (12.89), yields the 3rd-order model:

$$J_m\dot{n} = -(K_n + Q_{n|n|}\,|n|)n + Q^o_{|n|u_a}\,|n|\,u_p + \tau \tag{12.90}$$

$$\mathbf{H}\dot{\mathbf{x}} + \mathbf{D}_0\mathbf{x} + \mathbf{D}_1(\mathbf{x}, n, y)\mathbf{x} + |n|\,\mathbf{E}\mathbf{x} = \mathbf{f}(n) \tag{12.91}$$

$$y = \mathbf{h}^\top\mathbf{x} \tag{12.92}$$

12.4.2 Lyapunov Analysis

The observer (12.49) is used to generate an estimate $\hat{u}_p$ of u_p. Consider the control Lyapunov function candidate:

$$V = V_{\text{obs}} + \frac{1}{2}J_m\tilde{n}^2 \tag{12.93}$$

$$\dot{V} \leq -q_1\tilde{u}^2 - q_2\tilde{u}_p^2 + J_m\tilde{n}\dot{\tilde{n}} \tag{12.94}$$

where $\tilde{n} = n - n_d$ is the tracking error. Substituting (12.90) into (12.94), yields:

$$\dot{V} \leq -q_1\tilde{u}^2 - q_2\tilde{u}_p^2 + \tilde{n}[\tau - J_m\dot{n}_d - (K_n + Q_{n|n|}\,|n|)n + Q^o_{|n|u_a}\,|n|\,u_p] \tag{12.95}$$

The expression for $\dot{V}$ suggests that the control law τ should be chosen to include three parts; 1) a nonlinear P-controller, $-(K_{p0} + K_{p1}n^2)\tilde{n}$, 2) a nonlinear feedforward term based on the measured propeller revolution n and the desired propeller revolution n_d, and 3) a nonlinear cross-term, $-Q^o_{|n|u_a}\,|n|\,\hat{u}_p$, compensating for the axial flow into the propeller.

Theorem 12.2 (UGES Observer-Controller Error Dynamics)
Consider the nonlinear shaft speed controller:

$$\tau = \underbrace{-(K_{p0} + K_{p1}n^2)\tilde{n}}_{\text{P-control}} + \underbrace{J_m\dot{n}_d + (K_n + Q_{n|n|}\,|n|)n_d}_{\text{Reference Feed Forward}} - \underbrace{Q^o_{|n|u_a}\,|n|\,\hat{u}_p}_{\text{Axial Flow Compensator}} \tag{12.96}$$

with

$$K_{p0} > 0, \quad K_{p1} > \frac{1}{4}(Q^o_{|n|u_a})^2 > 0 \tag{12.97}$$

Let the estimate $\hat{u}_p$ be generated by using the nonlinear observer (12.49)–(12.50) with $q_2 > 1$ in Lemma 12.1 implying that K_{20} must satisfy $2d_{f0} > 1 + K_{20}^2/\beta$. Hence, the equilibrium point $(\tilde{u}, \tilde{u}_p, \tilde{n}) = (0,0,0)$ of the observer-controller error dynamics:

$$\mathbf{M}\dot{\tilde{\boldsymbol{\nu}}} + \mathbf{D}(\boldsymbol{\nu},t)\tilde{\boldsymbol{\nu}} + \mathbf{d} = \mathbf{0} \tag{12.98}$$

where $\mathbf{v} = [u, u_p, n]^\top$ and:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m_f & 0 \\ 0 & 0 & J_m \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} (-X_{u|u|})\,[u\,|u| - |u - \tilde{u}|\,(u - \tilde{u})] \\ d_f\,[|u_p|\,u_p - |u_p - \tilde{u}_p|\,(u_p - \tilde{u}_p)] \\ 0 \end{bmatrix}$$

$$\mathbf{D}(\boldsymbol{\nu},t) = \begin{bmatrix} -K_{10} - K_{11}\,|n| - X_u & (1-t)T^o_{|n|u_a}\,|n| & 0 \\ -K_{20} - K_{21}\,|n| & d_{f0} + T^o_{|n|u_a}\,|n| & 0 \\ 0 & -Q^o_{|n|u_a}\,|n| & K_{p0} + K_{p1}n^2 + K_n + Q_{n|n|}\,|n| \end{bmatrix}$$

is UGES.

Proof. Substituting (12.96) into (12.95), yields:

$$\dot{V} \leq -q_1\tilde{u}^2 - q_2\tilde{u}_p^2 + Q^o_{|n|u_a}\,|n|\,\tilde{u}_p\tilde{n} - (K_{p0} + K_{p1}n^2 + K_n + Q_{n|n|}\,|n|)\tilde{n}^2 \tag{12.99}$$

Using the fact that

$$-\left(\frac{1}{2}Q^o_{|n|u_a}\,|n|\,\tilde{n} - \tilde{u}_p\right)^2 = -\frac{1}{4}(Q^o_{|n|u_a})^2 n^2\tilde{n}^2 + Q^o_{|n|u_a}\,|n|\,\tilde{n}\tilde{u}_p - \tilde{u}_p^2$$

Hence, the cross term in (12.99) can be replaced by:

$$Q^o_{|n|u_a}\,|n|\,\tilde{n}\tilde{u}_p = -\left(\frac{Q^o_{|n|u_a}}{2}\,|n|\,\tilde{n} - \tilde{u}_p\right)^2 + \frac{(Q^o_{|n|u_a})^2}{4}n^2\tilde{n}^2 + \tilde{u}_p^2 \tag{12.100}$$

implying that:

$$\begin{aligned}\dot{V} &\leq -q_1\tilde{u}^2 - (q_2-1)\,\tilde{u}_p^2 - \left(K_{p1} - \frac{1}{4}(Q^o_{|n|u_a})^2\right) n^2\tilde{n}^2 \\ &- \left(\frac{1}{2}Q^o_{|n|u_a}\,|n|\,\tilde{n} - \tilde{u}_p\right)^2 - \left(K_{po} + K_n + Q_{n|n|}\,|n|\right)^2\tilde{n}^2 \\ &< 0, \forall \tilde{u} \neq 0, \tilde{u}_p \neq 0, \tilde{n} \neq 0 \end{aligned} \tag{12.101}$$

Hence, according to Lyapunov stability theory the equilibrium point $(\tilde{u}, \tilde{u}_p, \tilde{n}) = (0,0,0)$ of the observer-controller error dynamics (12.98) is UGES if $q_1 > 0, q_2 > 1, K_{p0} > 0$ and $K_{p1} > (1/4)(Q^o_{|n|u_a})^2$. ■

12.4.3 Extensions to Integral Control

When implementing the shaft speed propeller controller, it is important to include integral action in order to compensate for non-zero, slowly-varying disturbances and unmodelled dynamics. This can be done by augmenting a constant bias term b to (12.89) according to:

$$J_m\dot{n} = -K_n n + \tau - Q(n,u_p) + b \tag{12.102}$$

$$\dot{b} = 0 \tag{12.103}$$

Choosing the nonlinear control law of PI-type with reference feedforward and axial flow compensation, that is:

$$\begin{aligned}\tau = &\underbrace{-(K_{p0} + K_{p1}n^2)\tilde{n} - \hat{b}}_{\text{PI-control}} + \underbrace{J_m\dot{n}_d + (K_n + Q_{n|n|}\,|n|)n_d}_{\text{Reference Feed Forward}} \\ &- \underbrace{Q^o_{|n|u_a}\,|n|\,\hat{u}_p}_{\text{Axial Flow Compensator}} \end{aligned} \tag{12.104}$$

$$\dot{\hat{b}} = K_i\tilde{n}, \quad K_i > 0 \tag{12.105}$$

implies that (12.98) takes the form:

$$\dot{\mathbf{x}}_1 = \mathbf{h}(\mathbf{x}_1, t) + \mathbf{g}x_2 + \mathbf{d}_1 \tag{12.106}$$

$$\dot{x}_2 = -K_i\mathbf{g}^\top\mathbf{x}_1 \tag{12.107}$$

with $\mathbf{x}_1 = \tilde{\mathbf{v}} \in \mathbb{R}^3$, $x_2 = \hat{b} - b \in \mathbb{R}$ and

$$\mathbf{h}(\mathbf{x}_1, t) = -\mathbf{M}^{-1}\mathbf{D}(\mathbf{x}_1 + \boldsymbol{\nu}_d(t))\mathbf{x}_1 \tag{12.108}$$

$$\mathbf{g} = \mathbf{M}^{-1}\begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1/J_m \end{bmatrix} \tag{12.109}$$

$$\mathbf{d}_1 = -\mathbf{M}^{-1}\mathbf{d} \tag{12.110}$$

where we have used $\boldsymbol{\nu} = \mathbf{x}_1 + \boldsymbol{\nu}_d$. The error dynamics (12.106)–(12.107) is a nonlinear nonautonomous system complicating the Lyapunov stability analysis since $\dot{V}$ is only negative

semi-definite. Hence, *LaSalle-Krazovskii's theorem* cannot be used to prove uniformly globally asymptotic stability (UGAS). However, UGAS and uniformly locally exponentially stability (ULES) of the equilibrium point $(\mathbf{x}_1^\top, x_2) = (0, 0, 0, 0)$ of the error dynamics (12.106)–(12.107) can be proven by applying the result of Appendix A.2.4 known as"backstepping with integral action".

A simulation study documenting the performance of the observer and controller is found in Fossen and Blanke (2000).

Chapter 13

Decoupled Autopilot Design

This chapter discusses autopilot design for underwater vehicles (submersibles). These are usually classified according to type of application, for instance:

- ROV: remotely operated vehicles
- AUV: autonomous underwater vehicles
- UUV: unmanned underwater vehicles
- URV: underwater robotic vehicles

The vehicles are usually designed for submerged operation. A naval vessel or warship that is capable of propelling itself beneath the water, as well as on its surface, is referred to as a submarine. The submarine hull is usually a torpedo shaped, slender body for low drag, while commercial underwater vehicles, used in offshore gas and oil production are often nonregular boxes equipped with robotic manipulators, cameras and other tools. Underwater vehicle speed and position control systems are subject to an increased focus with respect to performance and safety. This is due to an increased number of commercial and military applications of UUVs. Modeling and control of underwater vehicles are discussed by Allmendinger (1990), Fossen (1994), Yuh (1995), and Triantafyllou and Hover (2002).

The 6 DOF underwater vehicle equations of motion can be divided into three non-interacting or lightly interacting subsystems for *speed control, steering* and *diving*; see Section 3.5. Each systems consists of the state variables:

1) Speed system state: $u(t)$.

2) Steering system states: $v(t), r(t)$ and $\psi(t)$.

3) Diving system states: $w(t), q(t), \theta(t)$ and $z(t)$.

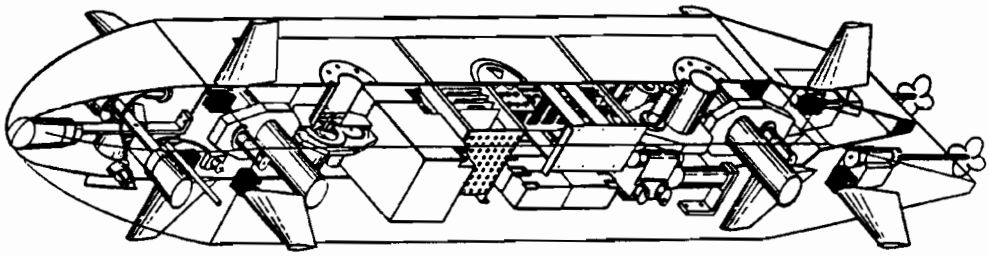


Figure 13.1: The NPS AUV II (Healey and Lienhard 1993).

The rolling mode, that is $p(t)$ and $\phi(t)$, is left passive in this approach. This decomposition has shown to describe the motion of slender formed vehicles like the the Naval Postgraduate School (NPS) AUV quite accurate; see Figure 13.1 (Healey and Marco 1992).

For this vehicle configuration, forward speed is controlled by means of propellers, steering/turning is obtained by using rudders, and bow and stern planes control the depth and pitch angle. This particular choice of actuators is inspired by those used in both flight and submarine control.

Guidance, Navigation and Control of Underwater Vehicles

Guidance, navigation and control of underwater vehicles have been addressed by a large number of authors Some useful references are; Fryxell *et al.* (1996), Pascoal *et al.* (1997), Leonard (1997), Conte and Serrani (1998), Yun *et al.* (1999), Caccia and Veruggio (2000), Liu *et al.* (2000), Lohmiller and Slotine (2000), Canudas de Wit *et al.* (2000), Grenon *et al.* (2001), Marco and Healey (2001), Leonard and Graver (2001), Encarnacao and Pascoal (2001*a*), and Encarnacao and Pascoal (2001*b*).

Adaptive Control of Underwater Vehicles

The highly nonlinear, time-varying dynamic behavior of underwater vehicles continually changes the parameters of the system model. The uncertainties in hydrodynamic coefficient make system identification difficult. These issues have motivated reserach on adaptive autopilots for underwater vehicles; see Yuh (1990), Fossen and Sagatun (1991*a*), Fjellstad *et al.* (1992), Tiano *et al.* (1997), Yuh *et al.* (1999), Antonelli *et al.* (2001), Zhao *et al.* (2001), and Yuh *et al.* (2001). Learning control for underwater vehicles have been discussed by Yuh and Lakshmi (1993), and Yuh (1994).

13.1 Course Autopilot

This section presents methods for robust control of underwater vehicles. In Section 3.5.3 the lateral underwater vehicle dynamics was written:

$$\begin{bmatrix} m - Y_{\dot{v}} & -mz_g - Y_{\dot{p}} & mx_g - Y_{\dot{r}} \\ -mz_g - Y_{\dot{p}} & I_x - K_{\dot{p}} & -I_{xz} - K_{\dot{r}} \\ mx_g - Y_{\dot{r}} & -I_{xz} - K_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \end{bmatrix}$$

$$+ \begin{bmatrix} -Y_v & -Y_p & -Y_r \\ -M_v & -M_p & -M_r \\ -N_v & -N_p & -N_r \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} \tag{13.1}$$

$$+ \begin{bmatrix} 0 & 0 & (m - X_{\dot{u}})u \\ 0 & 0 & 0 \\ (X_{\dot{u}} - Y_{\dot{v}})u & 0 & mx_g u \end{bmatrix} \begin{bmatrix} v \\ p \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ W\, BG_z \sin\phi \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_2 \\ \tau_4 \\ \tau_6 \end{bmatrix}$$

If the rolling mode is assumed to be negligible, this model reduces to the autopilot model:

$$\begin{bmatrix} m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} -Y_v & -Y_r \\ -N_v & -N_r \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & (m - X_{\dot{u}})u_o \\ (X_{\dot{u}} - Y_{\dot{v}})u_o & mx_g u_o \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} = \begin{bmatrix} \tau_2 \\ \tau_6 \end{bmatrix} \tag{13.2}$$

13.1.1 PID, Optimal Control and $\mathcal{H}_\infty$-Control

The course autopilot can be designed using the methods in Chapter 8 since the dynamics of both ships and underwater vehicles are quite accurate described by Nomoto's models. This includes:

- PID-control (Sections 8.4.2–8.4.3)
- LQ control (Section 8.4.5)

Nonlinear $\mathcal{H}_\infty$ optimal PID control of AUVs are discussed by Park *et al.* (2000) while scheduling $\mathcal{H}_\infty$ controllers have been proposed for ROVs by Kaminer *et al.* (1990) and Conte and Serrani (1998).

13.1.2 Nonlinear Control

Nonlinear control methods for autopilot control are described in Chapter 8. This includes:

- state feedback linearization (Section 8.4.6)
- backstepping (Section 8.4.8)
- SISO sliding mode control (Section 8.4.9)
- output feedback control (Section 8.4.10)

13.1.3 Sliding Mode Control using the Eigenvalue Decomposition

Healey and Lienard (1993) have applied the theory of *sliding mode control* to control the NPS AUV II. A related work discussing the problems of adaptive sliding mode control in the dive plane is found in Cristi *et al.* (1990). Sliding mode control for highly maneuverable underwater vehicles is discussed by Lyshevski (2001), who considers the 6 DOF underwater vehicle equations of motion.

The method presented in this section can be applied to the lateral model (13.1) which includes the roll mode, or to the reduced order model (13.2) which is recognized as the model of Davidson and Schiff (1946); see Section 8.1.2. The state-space representation is (Healey and Lienard 1993):

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{f}(\mathbf{x},t) \tag{13.3}$$

where $\mathbf{f}(\mathbf{x},t)$ is a a nonlinear function describing the deviation from linearity in terms of disturbances and unmodelled dynamics, $\mathbf{x} = [v, r, \psi]^\top$, and $u = \delta_R$ is the rudder angle. Consequently:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix} \tag{13.4}$$

The experiments of Healey and coauthors show that this model can be used to describe a large number of AUV maneuvers. The feedback control law is composed of two parts:

$$u = -\mathbf{k}^\top \mathbf{x} + u_o \tag{13.5}$$

where $\mathbf{k} \in \mathbb{R}^3$ is the feedback gain vector. Substituting (13.5) into (13.3) yields the closed-loop dynamics:

$$\dot{\mathbf{x}} = \underbrace{(\mathbf{A} - \mathbf{b}\mathbf{k}^\top)}_{\mathbf{A}_c}\mathbf{x} + \mathbf{b}u_o + \mathbf{f}(\mathbf{x},t) \tag{13.6}$$

where $\mathbf{k}$ is computed by means of pole-placement. In order to determine the nonlinear part u_o of the feedback control law, consider the output mapping:

$$\sigma(\tilde{\mathbf{x}}) = \mathbf{h}^\top \tilde{\mathbf{x}} \tag{13.7}$$

where $\mathbf{h} \in \mathbb{R}^3$ is a design vector to be chosen such that $\sigma(\tilde{\mathbf{x}}) \to 0$, implying convergence of the state tracking error $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d \to \mathbf{0}$. The output mapping $\sigma(\tilde{\mathbf{x}})$ is also referred to as a *sliding surface*. Pre-multiplication of (13.6) with $\mathbf{h}^\top$ and then subtraction of $\mathbf{h}^T\dot{\mathbf{x}}_d$ from both sides, gives:

$$\dot{\sigma}(\tilde{\mathbf{x}}) = \mathbf{h}^\top \mathbf{A}_c \mathbf{x} + \mathbf{h}^\top \mathbf{b}u_o + \mathbf{h}^\top \mathbf{f}(\mathbf{x},t) - \mathbf{h}^\top \dot{\mathbf{x}}_d \tag{13.8}$$

Assume that $\mathbf{h}^T\mathbf{b} \neq 0$ and let the nonlinear control law be chosen as:

$$u_o = (\mathbf{h}^\top \mathbf{b})^{-1}[\mathbf{h}^\top \dot{\mathbf{x}}_d - \mathbf{h}^\top \hat{\mathbf{f}}(\mathbf{x},t) - \eta \mathrm{sgn}(\sigma)], \qquad \eta > 0 \tag{13.9}$$

where $\hat{\mathbf{f}}(\mathbf{x},t)$ is an estimate of $\mathbf{f}(\mathbf{x},t)$. This gives the σ-dynamics:

$$\dot{\sigma}(\tilde{\mathbf{x}}) = \mathbf{h}^\top \mathbf{A}_c \mathbf{x} - \eta \text{sgn}(\sigma(\tilde{\mathbf{x}})) + \mathbf{h}^\top \Delta\mathbf{f}(\mathbf{x},t) \tag{13.10}$$

where $\Delta\mathbf{f}(\mathbf{x},t) = \mathbf{f}(\mathbf{x},t) - \hat{\mathbf{f}}(\mathbf{x},t)$. The first term in this equation can be rewritten as:

$$\mathbf{h}^\top \mathbf{A}_c \mathbf{x} = \mathbf{x}^\top \mathbf{A}_c^\top \mathbf{h} = \lambda \mathbf{x}^\top \mathbf{h} \tag{13.11}$$

by requiring that $\mathbf{h}$ is a *right eigenvector* of $\mathbf{A}_c^\top$ such that:

$$(\mathbf{A}_c^\top)\mathbf{h} = \lambda \mathbf{h} \tag{13.12}$$

where $\lambda = \lambda(\mathbf{A}_c^\top)$ is the *eigenvalue* corresponding to $\mathbf{h}$. Hence:

$$\dot{\sigma}(\tilde{\mathbf{x}}) = \lambda \mathbf{x}^\top \mathbf{h} - \eta \text{sgn}(\sigma(\tilde{\mathbf{x}})) + \mathbf{h}^\top \Delta\mathbf{f}(\mathbf{x},t) \tag{13.13}$$

Computation of $\mathbf{h}$ and $\mathbf{k}$

The eigenvalue λ in (13.13) can be made zero by noticing that (13.4) has one pure integrator. Let:

$$\mathbf{k} = [k_1 . k_2, 0]^\top \tag{13.14}$$

such that the linear part of the controller only stabilizes the sway velocity v and yaw rate r. The yaw angle ψ is left uncontrolled in the inner loop since this results in a closed system matrix:

$$\mathbf{A}_c = \begin{bmatrix} a_{11} - b_1 k_1 & a_{12} - b_1 k_2 & 0 \\ a_{21} - b_2 k_1 & a_{22} - b_2 k_2 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{13.15}$$

where one of the eigenvalues are zero. Consequently:

$$\lambda \mathbf{x}^\top \mathbf{h} = 0 \text{ if } \mathbf{h} \text{ is a right eigenvector of } \mathbf{A}_c^\top \text{ for } \lambda = 0 \tag{13.16}$$

With this choice of $\mathbf{h}$, the σ-dynamics (13.13) reduces to:

$$\dot{\sigma}(\tilde{\mathbf{x}}) = -\eta \text{sgn}(\sigma(\tilde{\mathbf{x}})) + \mathbf{h}^T \Delta\mathbf{f}(\mathbf{x},t) \tag{13.17}$$

and it follows from:

$$V(\sigma) = \frac{1}{2}\sigma^2 \tag{13.18}$$

that:

$$\begin{aligned} \dot{V}(\mathbf{x},t) &= \sigma\dot{\sigma} \\ &= -\eta\sigma \text{sgn}(\sigma) + \sigma \mathbf{h}^\top \Delta\mathbf{f}(\mathbf{x},t) \\ &= -\eta|\sigma| + \sigma \mathbf{h}^\top \Delta\mathbf{f}(\mathbf{x},t) \end{aligned} \tag{13.19}$$

Selecting η as:

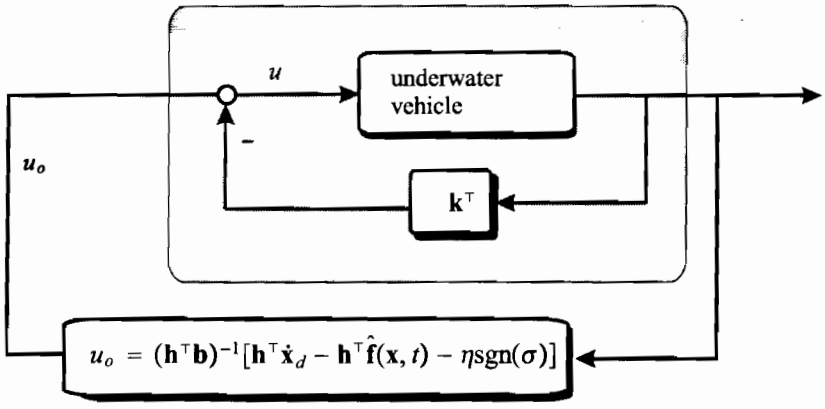


Figure 13.2: Nonlinear sliding mode controller.

$$\eta > \| \mathbf{h} \| \cdot \| \Delta\mathbf{f}(\mathbf{x},t) \| \tag{13.20}$$

finally yields:

$$\dot{V} \leq 0 \tag{13.21}$$

Hence, by application of Barbălat's lemma, σ converges to zero in finite time if η is chosen to be large enough to overcome the destabilizing effects of the unmodelled dynamics $\Delta\mathbf{f}(\mathbf{x},t)$. The magnitude of η will be a trade-off between robustness and performance.

Implementation Considerations

In practical implementations, chattering should be removed by replacing $\text{sgn}(\sigma)$ with:

$$\text{sat}(s) = \begin{cases} \text{sgn}(s) & \text{if } |s/\phi| > 1 \\ s/\phi & \text{otherwise} \end{cases} \tag{13.22}$$

where the design parameter ϕ is the sliding surface boundary layer thickness. Alternatively, the discontinuous function $\text{sat}(\sigma/\phi)$ could be replaced by the continuous function $\tanh(\sigma/\phi)$. It should be noted that the proposed feedback control with a predescribed η usually yields a conservative estimate of the necessary control action required to stabilize the plant. This suggests that η should be treated as a tunable parameter.

Example 13.1 (Course Autopilot)
Consider the ROV model:

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v - v_c \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix} \delta \tag{13.23}$$

where $|v_c| < v_c^{\max}$ *is the transverse current velocity. The reference trajectory is specified according to:*

$$\begin{bmatrix} \dot{\psi}_d \\ \dot{r}_d \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \begin{bmatrix} \psi_d \\ r_d \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \psi_{\text{ref}} \tag{13.24}$$

while $v_d = 0$ during turning. Let $\mathbf{x} = [v, r, \psi]^\top$ and $\mathbf{h} = [k_1, k_2, k_3]^\top$ such that:

$$\sigma = \mathbf{h}^\top(\mathbf{x} - \mathbf{x}_d) = h_1 v + h_2(r - r_d) + h_3(\psi - \psi_d) \tag{13.25}$$

Feedback from the sway velocity v and yaw rate r, that is $\mathbf{k} = [k_1, k_2, 0]^T$, implies that:

$$\mathbf{A}_c = \mathbf{A} - \mathbf{b}\mathbf{k}^T = \begin{bmatrix} a_{11} - b_1 k_1 & a_{12} - b_1 k_2 & 0 \\ a_{21} - b_2 k_1 & a_{22} - b_2 k_2 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{13.26}$$

where the yaw dynamics $\dot{\psi} = r$ is left unchanged. A pure integrator in yaw corresponding to the eigenvalue $\lambda = 0$ is necessary in order to satisfy:

$$\lambda \mathbf{x}^\top \mathbf{h} = 0 \tag{13.27}$$

The eigenvector $\mathbf{h}$ is computed in Matlab as

Matlab:

```
p = [-1 -1 0]          % desired poles for Ac
k = place(A,b,p)       % pole placement
Ac = A-b*k'
[V,D]=eig(Ac')         % eigenvalue decomposition
for i = 1:3            % extract the eigenvector h from V
   hi = V(:,i);
   if norm(hi.'*Ac) < 1e-10; h = hi; end
end
```

The resulting tracking controller is:

$$\delta = -k_1 v - k_2 r + \frac{1}{h_1 b_1 + h_2 b_2}\left[h_2 \dot{r}_d + h_3 r_d - \eta \mathrm{sat}(s)\right] \tag{13.28}$$

Since the disturbance v_c is unknown our best guess for:

$$\mathbf{f}(\mathbf{x}) = -\begin{bmatrix} a_{11} \\ a_{21} \\ 0 \end{bmatrix} v_c \tag{13.29}$$

is $\mathbf{f}(\hat{\mathbf{x}}) = \mathbf{0}$. Hence:

$$\eta > \| \mathbf{h} \| \cdot \| -[a_{11}, a_{12}, 0]^\top v_c^{\max} \| \tag{13.30}$$

13.2 Depth Autopilot

Pitch and depth control of underwater vehicles is usually done by using control surfaces, thrusters and ballast systems. For a neutrally buoyant vehicle, stern rudders are attractive for

diving and depth changing maneuvers, since they require relatively little control energy compared to thrusters. Consider, the longitudinal model in Section 3.5.3 which can be written:

$$\begin{bmatrix} m - X_{\dot{u}} & -X_{\dot{w}} & mz_g - X_{\dot{q}} \\ -X_{\dot{w}} & m - Z_{\dot{w}} & -mx_g - Z_{\dot{q}} \\ mz_g - X_{\dot{q}} & -mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \end{bmatrix}$$

$$+ \begin{bmatrix} -X_u & -X_w & -X_q \\ -Z_u & -Z_w & -Z_q \\ -M_u & -M_w & -M_q \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -(m - X_{\dot{u}})u \\ 0 & (Z_{\dot{w}} - X_{\dot{u}})u & mx_g u \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ W\, BG_z \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_3 \\ \tau_5 \end{bmatrix} \tag{13.31}$$

The speed dynamics can be removed from this model by assuming that the speed controller stabilizes the forward speed such that:

$$u = u_o = \text{constant} \tag{13.32}$$

Hence, (13.31) reduces to a combined pitch and diving model:

$$\begin{bmatrix} m - Z_{\dot{w}} & -mx_g - Z_{\dot{q}} \\ -mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -Z_w & -Z_q \\ -M_w & -M_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix}$$

$$+ \begin{bmatrix} 0 & -(m - X_{\dot{u}})u_o \\ (Z_{\dot{w}} - X_{\dot{u}})u_o & mx_g u_o \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ BG_z W \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_3 \\ \tau_5 \end{bmatrix}$$

A state-space representation of this model is:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u \tag{13.33}$$

$$\Updownarrow$$

$$\begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{d} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -u_0 & 0 \end{bmatrix} \begin{bmatrix} w \\ q \\ \theta \\ d \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ 0 \\ 0 \end{bmatrix} \delta_S \tag{13.34}$$

where δ_S is the stern rudder and the kinematic equations are based on the approximations (see Section 2.2.1):

$$\dot{\theta} = p\cos\phi - \sin\phi \approx q \tag{13.35}$$

$$\dot{d} = -u_0 \sin\theta + v\cos\theta\sin\psi + w\cos\theta\cos\psi \approx w - u_0\theta \tag{13.36}$$

for $v = p = 0$ and small values of θ and ϕ.

13.2.1 Optimal Control

It is straightforward to design an optimal diving autopilot based using the model (13.33). From Section 7.2.3 we have:

$$u = \underbrace{-\mathbf{R}^{-1}\mathbf{b}^\top\mathbf{P}_\infty}_{\mathbf{g}_1^\top}\mathbf{x} - \underbrace{\mathbf{R}^{-1}\mathbf{b}^\top(\mathbf{A}+\mathbf{b}\mathbf{g}_1^\top)^{-\top}\mathbf{C}^\top\mathbf{Q}}_{\mathbf{g}_2^\top} y_d \tag{13.37}$$

$$\mathbf{P}_\infty\mathbf{A} + \mathbf{A}^\top\mathbf{P}_\infty - \frac{1}{r}\mathbf{P}_\infty\mathbf{b}\mathbf{b}^\top\mathbf{P}_\infty + \tilde{\mathbf{Q}} = 0 \tag{13.38}$$

where $\mathbf{Q} = \mathbf{Q}^\top > 0, r > 0$, and $\mathbf{y}_d = d_d$ is the desired depth, e.g. generated using the reference model:

$$\dot{\mathbf{x}}_d = \mathbf{A}_d\mathbf{x}_d + \mathbf{b}_d r_{\text{com}} \tag{13.39}$$

$$y_d = \mathbf{c}^\top\mathbf{x}_d \tag{13.40}$$

where r_{com} is the operator input. This controller is based on the assumption that θ is small during depth changing maneuvers, and that the lateral states v, p, r, and ϕ are negligible. Since, this is a decoupled design large course changing maneuvers during diving will reduce the performance of the autopilot. However, experiments with low-speed underwater vehicles indicates that these are good assumptions.

13.2.2 Sliding Mode Control using the Eigenvalue Decomposition

The sliding mode controller of Section 13.1.3 can also be used to design a combined pitch and diving autopilot for the model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u + \mathbf{f}(\mathbf{x},t) \tag{13.41}$$

where $\mathbf{f}(\mathbf{x},t)$ is a a nonlinear function describing the deviation from linearity in terms of disturbances and unmodelled dynamics, $\mathbf{x} = [w, q, \theta, d]^\top$, and $u = \delta_S$ is the stern angle. This is illustrated in the following example.

Example 13.2 (Pitch and Diving Autopilot)
The sliding surface for pitch and diving control can be constructed as:

$$\sigma = \mathbf{h}^\top\tilde{\mathbf{x}} = h_1(w - w_d) + h_2(q - q_d) + h_3(\theta - \theta_d) + h_4(d - d_d) \tag{13.42}$$

where h_i for ($i = 1, \ldots, 4$) are the components of $\mathbf{h}$. Let $\mathbf{x}_d = [w_d, q_d, \theta_d, d_d]^\top$ be a desired state vector given by a reference model. From (13.5) and (13.9) it is seen that:

$$u = -\mathbf{k}^\top\mathbf{x} + u_o \tag{13.43}$$

$$u_o = (\mathbf{h}^\top\mathbf{b})^{-1}[\mathbf{h}^\top\dot{\mathbf{x}}_d - \mathbf{h}^\top\hat{\mathbf{f}}(\mathbf{x},t) - \eta\,\text{sgn}(\sigma)], \qquad \eta > 0 \tag{13.44}$$

where $\mathbf{k} = [k_1, k_2, 0, k_4]^T$. Notice that $k_3 = 0$ since there is one pure integration in the pitch channel. Hence, $\mathbf{h}$ is found by computing the eigenvalues $\lambda(\mathbf{A}_c) = \lambda(\mathbf{A} - \mathbf{b}\mathbf{k}^\top)$ where $\mathbf{A}_c^\top\mathbf{h} = 0$ for $\lambda_3 = 0$. Consequently:

$$\delta_S = -k_1 w - k_2 q - k_4 d + \frac{1}{h_1 b_1 + h_2 b_2}[h_1\dot{w}_d + h_2\dot{q}_d + h_3\theta_d + h_4 d_d - \eta\,\text{sgn}(\sigma)] \tag{13.45}$$

13.3 Speed Control System

The forward speed controller is designed using the surge equation in (13.31):

$$(m - X_{\dot{u}})\dot{u} - X_u u = \tau_1 + T_{\text{loss}} \tag{13.46}$$

where T_{loss} contains coupling terms and environmental disturbances. This is a 1st-order system which can be easily controlled using a PI-controller with reference feedforward:

$$\tau_1 = \underbrace{(m - X_{\dot{u}})\dot{u}_d - X_u u_d}_{\text{reference feedforward}} \underbrace{- K_p(u - u_d) - K_i \int_0^t (u - u_d)d\tau}_{\text{PI-controller}} \tag{13.47}$$

where u_d is the desired velocity. The error dynamics is:

$$(m - X_{\dot{u}})\ddot{z} + (K_p - X_u)\dot{z} + K_i z = T_{\text{loss}} \tag{13.48}$$

where:

$$z = \int_0^t (u - u_d)d\tau \tag{13.49}$$

Hence, it is straightforward to choose the controller gains $K_p > 0$ and $K_i > 0$ such that the closed loop system is GES. A pole placement algorithm for this is:

$$\frac{K_p - X_u}{m - X_{\dot{u}}} = 2\zeta\omega_n \tag{13.50}$$

$$\frac{K_i}{m - X_{\dot{u}}} = \omega_n^2 \tag{13.51}$$

where relative damping ratio ζ and natural frequency ω_n are the new design parameters. Hence:

$$\ddot{z} + 2\zeta\omega_n\dot{z} + \omega_n^2 z = \frac{1}{m - X_{\dot{u}}} T_{\text{loss}} \tag{13.52}$$

13.4 Exercises

Exercise 13.1 *Consider Example 13.1 and let:*

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} -0.250 & -0.870 & 0 \\ -0.012 & -0.230 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v - v_c \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} 0.220 \\ -0.043 \\ 0 \end{bmatrix} \delta_R$$

where δ_R is the rudder input and v_c is the transverse current velocity in body-fixed coordinates. Let:

$$v_c = V_c \sin(\psi_c - \psi)$$

where V_c and ψ_c are the NED current speed and direction, respectively. Let $\psi_c = 90^o$ while the maximum current velocity is denoted as α_0, that is $\sup_t |v_c(t)| \leq \alpha_0$. The current speed

V_c is defined below. Include in your model that the rudder angle is limited to $\delta_{\max} = 35°$.

a)Determine k_1 and k_2 such that the eigenvalues of $\mathbf{A}_c$ becomes: $\lambda_1 = -0.5$, $\lambda_2 = -0.32$, $\lambda_3 = 0$.

b) Derive a sliding mode controller for the ROV and find a formula for η such that $\dot{V} \leq 0$.

c) Simulate the closed-loop system in Matlab or Simulink for a step response $\psi_{\text{ref}} = 20°$ with initial conditions $(v(0), r(0), \psi(0)) = (0, 0, 0)$ using the following reference model:

$$\ddot{\psi}_d + 2\zeta\omega_o\dot{\psi}_d + \omega_o^2\psi_d = \omega_o^2\psi_{\text{ref}}$$

where $\zeta = 0.8$ and $\omega_o = 0.1$. In the simulation study the boundary layer thickness should be altered from $\phi = 1.0$, 0.1 and 0.01 for the following three cases:

- no disturbances: $V_c = 0$ m/s
- constant current speed $V_c = 0.5$ m/s
- time-varying current speed $V_c = 0.5\sin(0.2t)$ m/s

For each simulation present plots of all variables v, r, ψ and δ. Include comments to the results.

***Hint:** determine η by choosing $|\Delta\mathbf{f}(\mathbf{x})| = \max|disturbance|$.*

Exercise 13.2 *The crew onboard a navy submarine can in an emergency situation beed to be rescued by using a Deep Submergence Rescue Vehicle (DSRV). Consider the following model (from the lecture notes of Professor Anthony Healey, Naval Postgraduate School (NPS), Monterey, CA, 1992):*

$$\begin{aligned} \dot{n} &= u\cos\theta + w\sin\theta \\ \dot{d} &= -u\sin\theta + w\cos\theta \\ \dot{\theta} &= q \end{aligned}$$

where n and d are the North-Down positions, θ is the pitch angle and:

$$\begin{bmatrix} m - Z_{\dot{w}} & -Z_{\dot{q}} \\ -M_{\dot{w}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -Z_w & -Z_q \\ -M_w & -M_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -M_\theta \end{bmatrix} \begin{bmatrix} z \\ \theta \end{bmatrix} = \begin{bmatrix} Z_{\delta_S} \\ M_{\delta_S} \end{bmatrix} \delta_S$$

Here w and q are the heave velocity and pitch rate, respectively. This model describes the dynamics in heave and pitch at constant cruise speed $u = U_0 = 4.11$ (m/s). A Matlab m-file for computation of the DSRV dynamics is included the Matlab GNC toolbox (type help vesselmodels). The following analysis should be performed:

a) Find a formula for the restoring moment in pitch, M_θ, as a function of CG, CB, θ and mass (Hint: see the general expression for $\mathbf{g}(\boldsymbol{\eta})$ for a submerged vehicle)?

b) Compute the natural frequency ω_θ in pitch and the corresponding pitch period T_θ (use the numerical values in DSRV.m). Explain why there is no natural frequency ω_z in heave?

c) For a mass-damper-spring *system:*

$$\begin{aligned} m\ddot{x} + d\dot{x} + kx &= \tau \\ &\Updownarrow \\ \ddot{x} + 2\zeta\omega_o\dot{x} + \omega_o^2 x &= \tau \end{aligned}$$

it can be shown that:

$$2\zeta\omega_0 = \frac{d}{m}, \quad \omega_0^2 = \frac{k}{m}$$

where ω_o is the natural frequency ($d = 0$). Assume that $\omega = r\omega_o$ is the frequency in the damped system $d > 0$ with $0 < r < 1$ (indicates how much the frequency is reduced in the damped system). Show that:

$$d = 2\sqrt{1 - r^2}\sqrt{km}$$

This formula can be used to compute the damping coefficient d in heave, pitch *and* roll *for known k and m. For the other DOF this is impossible since $k = 0$. Compute the r factor in pitch (assume that there are no couplings between pitch and the other modes in the model). How many percent reduction in frequency is observed due to damping?*

d) Is the DSRV stable for small pitch angles θ during diving? (Hint: Compute the eigenvalues for the linearized model of the DSRV with steady-state value $\theta_{ss} = 0$).

Exercise 13.3 *Consider the DSRV in Example 13.2.*

a) Design a PID-based depth controller for the DSRV in Example 13.2 and simulate a depth changing maneuver from $d = 10$ (m) to $d = 100$ (m) for initial conditions $w(0) = q(0) = \theta(0) = 0$. Plot the trajectories $d(t)$ and $\theta(t)$ as functions of time. Also plot the depth profile. Check your design by plotting the stern rudder displacement $\delta_S(t)$ as function of time (the rudder should not saturate during the maneuver).

b) Assume that you only measure the states $n(t), d(t)$, and $\theta(t)$, and that $q(t)$ is unknown. Estimate $q(t)$ in a Kalman filter and plot $q(t)$ (true value) and the estimate $\hat{q}(t)$ in the same plot as a function of time. Does $\hat{q}(t)$ converge to $q(t)$? Does the PID-controller work satisfactoryily if all the states are replaced with the Kalman filter state estimates? (separation principle).

Chapter 14

6 DOF Position and Attitude Control

In Chapter 13 it was shown how three different control systems could be designed independently for *turning*, *diving* and *forward speed* control. This was based on the assumption that the longitudinal and lateral modes were decoupled. In addition, the speed equation was assumed to be independent of heave and pitch, while roll was left uncontrolled. In this chapter, 6 DOF nonlinear controllers for simultaneously control of:

- position (surge, sway, and yaw)
- attitude (roll, pitch, and yaw)

are presented. These control laws are designed for high-speed coupled maneuvers in 6 DOF while the decoupled control strategy should be used for noncoupled maneuvers. The presented methods are based on Fossen (1991, 1994).

14.1 Nonlinear PID Control

In this section 6 DOF nonlinear PID control is applied to underwater vehicles. It is assumed that the vehicles are fully actuated or overactuated such that the generalized control forces $\boldsymbol{\tau} = \mathbf{B}\mathbf{u}$ can be distributed to the actuators $\mathbf{u}$ using one of the methods in Section 7.5. In the unconstrained case:

$$\mathbf{u} = \mathbf{B}^{\dagger}\boldsymbol{\tau} \tag{14.1}$$

This formula is attractive due its simple real-time representation.

The 6 DOF equations of motion are written in accordance with the model representation used in Chapter 3. Let:

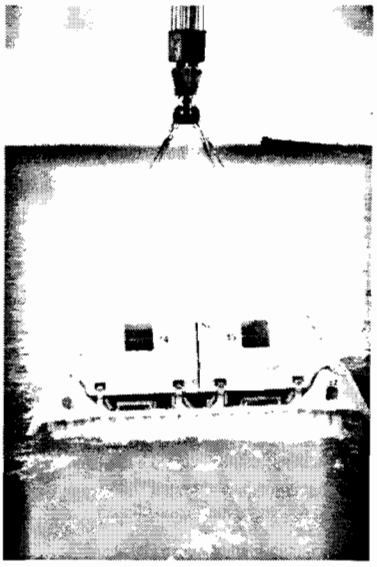


Figure 14.1: Remotely operated vehicle (ROV).

$$\dot{\boldsymbol{\eta}}_1 = \mathbf{J}_{\Theta}(\boldsymbol{\Theta})\boldsymbol{\nu} \tag{14.2}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}_1(\boldsymbol{\Theta}) = \boldsymbol{\tau} + \mathbf{w} \tag{14.3}$$

and:

$$\dot{\boldsymbol{\eta}}_2 = \mathbf{J}_q(\mathbf{q})\boldsymbol{\nu} \tag{14.4}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}_2(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{w} \tag{14.5}$$

where the attitude dynamics is represented by the *Euler angle* and *unit quaternion* representations (see Section 2.2):

$$\underbrace{\begin{bmatrix} \dot{\mathbf{p}}^n \\ \dot{\boldsymbol{\Theta}} \end{bmatrix}}_{\dot{\boldsymbol{\eta}}_1} = \underbrace{\begin{bmatrix} \mathbf{R}_b^n(\boldsymbol{\Theta}) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{T}_{\Theta}(\boldsymbol{\Theta}) \end{bmatrix}}_{\mathbf{J}_{\Theta}(\boldsymbol{\Theta})} \underbrace{\begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix}}_{\boldsymbol{\nu}} \tag{14.6}$$

and:

$$\underbrace{\begin{bmatrix} \dot{\mathbf{p}}^n \\ \dot{\mathbf{q}} \end{bmatrix}}_{\dot{\boldsymbol{\eta}}_2} = \underbrace{\begin{bmatrix} \mathbf{R}_b^n(\mathbf{q}) & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{4\times 3} & \mathbf{T}_q(\mathbf{q}) \end{bmatrix}}_{\mathbf{J}_q(\mathbf{q})} \underbrace{\begin{bmatrix} \mathbf{v}_o^b \\ \boldsymbol{\omega}_{nb}^b \end{bmatrix}}_{\boldsymbol{\nu}} \tag{14.7}$$

The Euler angles can be transformed to unit quaternions and *vice verse* using Algorithms 2.2 and 2.3 in Sections 2.2.3 and .2.2.4, respectively. Nonlinear control methods for both these representations are now presented.

14.1.1 Set-Point Regulation

Set-point regulations is obtained by specifying both the desired position $\mathbf{p}_d = [n_d, e_d, d_d]^\top$ and attitude $\boldsymbol{\Theta}_d = [\phi_d, \theta_d, \psi_d]^\top$, alternatively $\mathbf{q}_d = [\eta_d, \varepsilon_{1d}, \varepsilon_{2d}, \varepsilon_{3d}]^\top$, according to:

$$\boldsymbol{\eta_d} = \begin{bmatrix} \mathbf{p}_d \\ \boldsymbol{\Theta}_d \text{ (or } \mathbf{q}_d) \end{bmatrix} = \begin{bmatrix} \text{constant} \\ \text{constant} \end{bmatrix} \tag{14.8}$$

Position and Euler Angle Feedback

In Section 7.1.4 it was shown that:

$$V_1 = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{H}\boldsymbol{\nu} + \frac{1}{2}\tilde{\boldsymbol{\eta}}_1^\top \mathbf{K}_p \tilde{\boldsymbol{\eta}}_1, \qquad \tilde{\boldsymbol{\eta}}_1 = \begin{bmatrix} \mathbf{p} - \mathbf{p}_d \\ \boldsymbol{\Theta} - \boldsymbol{\Theta}_d \end{bmatrix} \tag{14.9}$$

where $\mathbf{H} = \mathbf{M} + \mathbf{K}_m$ resulted in:

$$\dot{V}_1 = \boldsymbol{\nu}^\top \mathbf{w} - \boldsymbol{\nu}^\top \left[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\Theta})\right] \boldsymbol{\nu}, \qquad \mathbf{K}_d^*(\boldsymbol{\Theta}) = \mathbf{J}_{\boldsymbol{\Theta}}^\top(\boldsymbol{\Theta}) \mathbf{K}_d \mathbf{J}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta}) \tag{14.10}$$

if the control law was chosen to be of PID type with acceleration feedback and gravity/buoyancy compensation:

$$\boldsymbol{\tau} = \mathbf{g}_1(\boldsymbol{\Theta}) - \mathbf{K}_m \dot{\boldsymbol{\nu}} - \mathbf{J}_{\boldsymbol{\Theta}}^\top(\boldsymbol{\Theta}) \boldsymbol{\tau}_{\text{PID}} \tag{14.11}$$

$$\boldsymbol{\tau}_{\text{PID}} = \mathbf{K}_p \tilde{\boldsymbol{\eta}}_1 + \mathbf{K}_d \dot{\boldsymbol{\eta}}_1 + \mathbf{K}_i \int_o^t \tilde{\boldsymbol{\eta}}_1(\tau) d\tau \tag{14.12}$$

If $\mathbf{w} = \mathbf{0}$ and $\mathbf{K}_i = \mathbf{0}$, it follows that $\dot{V}_1 = -\boldsymbol{\nu}^\top \left[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\boldsymbol{\Theta})\right] \boldsymbol{\nu} \leq 0$, and *Krasovskii–LaSalle's* Theorem guarantees that the equilibrium point $(\tilde{\boldsymbol{\eta}}, \boldsymbol{\nu}) = (\mathbf{0}, \mathbf{0})$ is GAS. If $\mathbf{w} \neq \mathbf{0}$ but bounded, the system trajectories will converge to ball about the origin. The radius of the ball depends on the magnitude of the disturbance vector $\mathbf{w}$. If $\mathbf{w}$ is constant and $\mathbf{K}_i > \mathbf{0}$ (PID-control) it can be shown that the system is locally asymptotically stable.

Position and Unit Quaternion Feedback

The attitude part of the nonlinear PID controller (14.12) can be modified to use unit quaternion feedback. Hence, the singular point $\theta = \pm 90\,^\circ$ can be avoided in a maneuvering situation (Fjellstad and Fossen 1994*b*). Quaternions also have better numerical properties than Euler angles. Let:

$$V_2 = \frac{1}{2}\boldsymbol{\nu}^\top \mathbf{H}\boldsymbol{\nu} + \frac{1}{2}\tilde{\boldsymbol{\eta}}_2^\top \begin{bmatrix} \mathbf{K}_p & \mathbf{0}_{3\times 4} \\ \mathbf{0}_{4\times 3} & c\mathbf{I}_{4\times 4} \end{bmatrix} \tilde{\boldsymbol{\eta}}_2, \qquad \tilde{\boldsymbol{\eta}}_2 = \begin{bmatrix} \mathbf{p} - \mathbf{p}_d \\ \mathbf{q} - \begin{bmatrix} 1 \\ \mathbf{0}_{3\times 1} \end{bmatrix} \end{bmatrix} \tag{14.13}$$

where $\mathbf{K}_p = \mathbf{K}_p^\top > 0$ and $c > 0$. It then follows that:

$$\dot{V}_2 = \boldsymbol{\nu}^\top \mathbf{w} - \boldsymbol{\nu}^\top \left[\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d^*(\mathbf{q})\right] \boldsymbol{\nu}, \qquad \mathbf{K}_d^*(\mathbf{q}) = \mathbf{J}_q^\top(\mathbf{q}) \mathbf{K}_d \mathbf{J}_q(\mathbf{q}) \tag{14.14}$$

if the controller is chosen as:

$$\tau = \mathbf{g}_2(\mathbf{q}) - \mathbf{K}_m\dot{\boldsymbol{\nu}} - \mathbf{J}_q^\top(\mathbf{q})\tau_{\text{PID}} \tag{14.15}$$

$$\tau_{\text{PID}} = \mathbf{K}_p\tilde{\boldsymbol{\eta}}_2 + \mathbf{K}_d\dot{\boldsymbol{\eta}}_2 + \mathbf{K}_i\int_o^t \tilde{\boldsymbol{\eta}}_2(\tau)d\tau \tag{14.16}$$

For this system there are two closed-loop equilibria corresponding to:

$$\tilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_d = [\tilde{\eta}, \tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \tilde{\varepsilon}_3]^\top = [\pm 1, 0, 0, 0]^\top \tag{14.17}$$

Both equilibria represent the desired attitude $\mathbf{q}_d$. If $\mathbf{q}$ represent one certain attitude, then $-\mathbf{q}$ is the same attitude after a $\pm 2\pi$ rotation about an axis. Physically these two points are indistinguishable, but mathematically they are distinct. In fact, $\tilde{\eta} = +1$ is a stable equilibrium point, whereas $\tilde{\eta} = -1$ is unstable. This can be seen from the following discussion: suppose $\tilde{\eta} = -1$ and $\tilde{\mathbf{p}} = \mathbf{0}_{3\times 1}$, then the steady-state value of the Lyapunov function V_2 is:

$$V_{2s} = 2c$$

If the system is perturbed to $\tilde{\eta} = -1 + \varepsilon$, where $\varepsilon > 0$, it can be shown that V_2 takes the value:

$$V_2 = 2c - \varepsilon < V_{2s}$$

Since V_2 decreases monotonically for $\tilde{\eta} \neq \pm 1$, the system can never return to the unstable equilibrium point. Consequently, application of Krasovskii–LaSalle's theorem, implies (almost) global asymptotic stability. From $\dot{V}_2$ it is observed that asymptotic stability is obtained even with a simple proportional feedback control law, that is $\mathbf{K}_d = \mathbf{0}$.

The unstable equilibrium point of the vector quaternion feedback control law is a well known attitude control phenomenon. A small perturbation in $\mathbf{q}$ will cause the vehicle to rotate an angle of 2π about an axis. This behavior is closely related to the mathematical properties of the Euler parameters, or unit quaternions in general.

In Fjellstad and Fossen (1994*b*) controllers using *Euler rotation* and *Rodriques parameter* feedback are also derived. These representations are singular so there is nothing to gain compared to Euler angles. A generalization to trajectory tracking control is presented in the next section.

14.1.2 Trajectory Tracking Control

In this section the assumption that $\boldsymbol{\eta}_d$ is constant is relaxed. Assume that $\boldsymbol{\eta}_d$ is a time-varying reference trajectory and that:

$$\boldsymbol{\eta}_d^{(3)}, \ddot{\boldsymbol{\eta}}_d, \dot{\boldsymbol{\eta}}_d, \boldsymbol{\eta}_d \text{ are smooth and bounded} \tag{14.18}$$

Position and Euler Angle Feedback

In Section 7.4.6 the trajectory tracking problem was solved for the model (14.2)–(14.3). For underwater vehicles the controller (7.387) takes the form:

$$\tau = \mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{g}_1(\boldsymbol{\Theta}) - \mathbf{J}_{\boldsymbol{\Theta}}^\top(\boldsymbol{\Theta})\mathbf{K}_p\tilde{\boldsymbol{\eta}}_1 - \mathbf{J}_{\boldsymbol{\Theta}}^\top(\boldsymbol{\Theta})\mathbf{K}_d\mathbf{s}_1 \tag{14.19}$$

where:

$$\mathbf{s}_1 = \dot{\tilde{\boldsymbol{\eta}}}_1 + \boldsymbol{\Lambda}_1 \tilde{\boldsymbol{\eta}}_1 \tag{14.20}$$

$$\dot{\boldsymbol{\eta}}_r = \dot{\boldsymbol{\eta}}_d - \boldsymbol{\Lambda}_1 \tilde{\boldsymbol{\eta}}_1, \quad \boldsymbol{\Lambda}_1 > 0 \tag{14.21}$$

$$\boldsymbol{\nu}_r = \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta}) \dot{\boldsymbol{\eta}}_r, \quad (\theta \neq \pm 90^o) \tag{14.22}$$

In Section 7.4.6 it was shown that thee equilibrium point $(\tilde{\boldsymbol{\eta}}_1, \mathbf{s}_1) = (\mathbf{0}, \mathbf{0})$ is GES such that convergence of $\mathbf{s}_1 \rightarrow \mathbf{0}$ and $\tilde{\boldsymbol{\eta}}_1 \rightarrow \mathbf{0}$ implies that $\tilde{\mathbf{v}} \rightarrow \mathbf{0}$.

Position and Unit Quaternion Feedback

The position and attitude controller (14.19) can be modified to use unit quaternion feedback by using the approach of Fjellstad and Fossen (1994*a*). The main problem in doing this is that $\mathbf{J}_q^{-1}(\mathbf{q})$ does not exists since $\mathbf{J}_q(\mathbf{q}) \in \mathbb{R}^{7\times 6}$ is a nonsquare matrix. Hence, the virtual reference signals (14.20)–(14.22) cannot be used in conjunction with unit quaternions. This problem is solved by defining the following signals (Fjellstad and Fossen 1994*a*):

$$\mathbf{s}_2 = \boldsymbol{\nu} - \boldsymbol{\nu}_r \tag{14.23}$$

$$\boldsymbol{\nu}_r = \boldsymbol{\nu}_d - \underbrace{\begin{bmatrix} \lambda \mathbf{R}_b^n(\mathbf{q}_d)^\top & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & -2c\,\mathrm{sgn}(\tilde{\eta})\mathbf{I}_{3\times 3} \end{bmatrix}}_{\boldsymbol{\Lambda}_2} \begin{bmatrix} \tilde{\mathbf{p}}^n \\ \tilde{\boldsymbol{\varepsilon}} \end{bmatrix} \tag{14.24}$$

$$\dot{\boldsymbol{\eta}}_d = \mathbf{J}_q(\mathbf{q})\boldsymbol{\nu}_d \tag{14.25}$$

where $\tilde{\boldsymbol{\varepsilon}} = [\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \tilde{\varepsilon}_3]^\top$. Let the position and unit quaternion feedback tracking controller for the system (14.4)–(14.5) be chosen as:

$$\boldsymbol{\tau} = \mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu}_r + \mathbf{g}_2(\mathbf{q}) - \mathbf{K}_d\mathbf{s}_2 \tag{14.26}$$

Substituting (14.26) into (14.4) gives the error dynamics:

$$\mathbf{M}\dot{\mathbf{s}}_2 + \mathbf{C}(\boldsymbol{\nu})\mathbf{s}_2 + \mathbf{D}(\boldsymbol{\nu})\mathbf{s}_2 + \mathbf{K}_d\mathbf{s}_2 = \mathbf{w} \tag{14.27}$$

Consequently:

$$V_2 = \frac{1}{2}\mathbf{s}_2^\top \mathbf{M}\mathbf{s}_2 \tag{14.28}$$

$$\dot{V}_2 = -\mathbf{s}_2^\top(\mathbf{D}(\boldsymbol{\nu}) + \mathbf{K}_d)\mathbf{s}_2 \tag{14.29}$$

implies convergence of $\mathbf{s}_2 \rightarrow \mathbf{0}$ if $\mathbf{w} = \mathbf{0}$.The dynamics associated with $\mathbf{s}_2 = \mathbf{0}$ is found from (14.23) and (14.24):

$$\begin{bmatrix} \tilde{\mathbf{v}}_o^b \\ \tilde{\boldsymbol{\omega}}_{nb}^b \end{bmatrix} = \begin{bmatrix} \lambda \mathbf{R}_b^n(\mathbf{q}_d)^\top & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & 2c\,\mathrm{sgn}(\tilde{\eta})\mathbf{I}_{3\times 3} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}^n \\ \tilde{\boldsymbol{\varepsilon}} \end{bmatrix} \tag{14.30}$$

It will now be shown that both $\tilde{\mathbf{p}}^n$ and $\tilde{\boldsymbol{\omega}}_{nb}^b$ converge to zero for $\lambda > 0$ and $c > 0$.

Since $\dot{\tilde{\mathbf{p}}}^n = \mathbf{R}_b^n(\mathbf{q})\tilde{\mathbf{v}}_o^b$ it follows that:

$$\dot{\tilde{\mathbf{p}}}^n = \mathbf{R}_b^n(\mathbf{q})\tilde{\mathbf{v}}_o^b = \lambda \mathbf{R}_b^n(\mathbf{q})\mathbf{R}_b^n(\mathbf{q}_d)^\top \tilde{\mathbf{p}}^n \tag{14.31}$$

Using (14.30) implies that Equation (14.31) can be written:

$$\dot{\tilde{\mathbf{p}}}^n = \lambda \mathbf{R}_b^n(\tilde{\mathbf{q}})\tilde{\mathbf{p}}^n, \qquad \tilde{\mathbf{R}}_b^n(\tilde{\mathbf{q}}) = \mathbf{R}_b^n(\mathbf{q})\mathbf{R}_b^n(\mathbf{q}_d)^\top \tag{14.32}$$

where $\mathbf{R}_b^n(\tilde{\mathbf{q}})$ is strictly positive for $\tilde{\eta}^2 > 1/2$. Consequently, $\tilde{\mathbf{p}}^n$ converges to zero for $\lambda > 0$. The quaternion error is defined as:

$$\tilde{\mathbf{q}} = \mathbf{q}^* \otimes \mathbf{q}_d \tag{14.33}$$

where $\mathbf{q}^* = [\eta, -\varepsilon_1, -\varepsilon_2, -\varepsilon_3]^\top$ is the conjugate of $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top$. The symbol $\otimes$ denotes the *quaternion product* which is defined as (Chou 1992):

$$\begin{aligned} \mathbf{q}_1 \otimes \mathbf{q}_2 &= \begin{bmatrix} \eta_1\eta_2 - \boldsymbol{\varepsilon}_1^\top \boldsymbol{\varepsilon}_2 \\ \eta_2\boldsymbol{\varepsilon}_1 + \eta_1\boldsymbol{\varepsilon}_2 + \boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2 \end{bmatrix} \\ &= \begin{bmatrix} \eta_1 & -\boldsymbol{\varepsilon}_1^\top \\ \boldsymbol{\varepsilon}_1 & \eta_1\mathbf{I} + \mathbf{S}(\boldsymbol{\varepsilon}_1) \end{bmatrix} \mathbf{q}_2 \end{aligned} \tag{14.34}$$

To prove convergence of $\tilde{\boldsymbol{\omega}}_{nb}^b$ to zero consider (see (2.66) in Section 2.2.2):

$$\dot{\tilde{\mathbf{q}}} = \frac{1}{2}\begin{bmatrix} -\tilde{\boldsymbol{\varepsilon}}^\top \\ \tilde{\eta}\mathbf{I} + \mathbf{S}(\tilde{\boldsymbol{\varepsilon}}) \end{bmatrix} \tilde{\boldsymbol{\omega}}_{nb}^b \quad \Longrightarrow \quad \dot{\tilde{\eta}} = -\frac{1}{2}\tilde{\boldsymbol{\varepsilon}}^\top \tilde{\boldsymbol{\omega}}_{nb}^b \tag{14.35}$$

Consider the Lyapunov function:

$$W = 1 - \mathrm{abs}(\tilde{\eta}) \tag{14.36}$$

and:

$$\begin{aligned} \dot{W} &= -\mathrm{sgn}(\tilde{\eta})\dot{\tilde{\eta}} \\ &= -\frac{1}{2}\mathrm{sgn}(\tilde{\eta})\tilde{\boldsymbol{\varepsilon}}^\top \tilde{\boldsymbol{\omega}}_{nb}^b \end{aligned} \tag{14.37}$$

An expression for $\tilde{\boldsymbol{\omega}}_{nb}^b$ is found in (14.30) such that:

$$\begin{aligned} \dot{W} &= -\frac{1}{2}\mathrm{sgn}(\tilde{\eta})\tilde{\boldsymbol{\varepsilon}}^\top [2c\,\mathrm{sgn}(\tilde{\eta})\tilde{\boldsymbol{\varepsilon}}] \\ &= -c\tilde{\boldsymbol{\varepsilon}}^\top \tilde{\boldsymbol{\varepsilon}} < 0 \end{aligned} \tag{14.38}$$

Consequently, $\tilde{\boldsymbol{\varepsilon}} \to \mathbf{0}$ and $\tilde{\eta} \to \pm 1$ (recall that $\tilde{\eta}^2 + \tilde{\boldsymbol{\varepsilon}}^\top \tilde{\boldsymbol{\varepsilon}} = 1$). It then follows from (14.30) that $\tilde{\boldsymbol{\omega}}_{nb}^b \to \mathbf{0}$.

A more general discussion on alternative kinematic representations is found in Fjellstad and Fossen (1994*c*). Extensions to adaptive control can be done by using the approach of Fossen and Sagatun (1991*b*).

14.2 State Feedback Linearization

In Section 7.3 feedback linearization was used as a tool for velocity and position/attitude control. These methods were referred to as decoupling in the *b*- and *n*-frame, respectively. It is straightforward to apply these schemes for trajectory tracking control of underwater vehicles in 6 DOF. The nonlinear model of an underwater vehicle in terms of *Euler angle* and *unit quaternion* representations is written:

$$\begin{aligned} \dot{\boldsymbol{\eta}}_1 &= \mathbf{J}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\nu} && (14.39)\\ \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}_1(\boldsymbol{\nu}, \boldsymbol{\Theta}) &= \boldsymbol{\tau} && (14.40)\\ \mathbf{n}_1(\boldsymbol{\nu}, \boldsymbol{\Theta}) &= \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}_1(\boldsymbol{\Theta}) && (14.41) \end{aligned}$$

and:

$$\begin{aligned} \dot{\boldsymbol{\eta}}_2 &= \mathbf{J}_q(\mathbf{q})\boldsymbol{\nu} && (14.42)\\ \mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}_2(\boldsymbol{\nu}, \mathbf{q}) &= \boldsymbol{\tau} && (14.43)\\ \mathbf{n}_2(\boldsymbol{\nu}, \mathbf{q}) &= \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}_2(\mathbf{q}) && (14.44) \end{aligned}$$

14.2.1 Trajectory Tracking Control

In this section trajectory tracking controllers using the reference trajectories:

$$\boldsymbol{\eta}_d = [n_d, e_d, d_d, \phi_d, \theta_d, \psi_d]^\top \quad (14.45)$$

$$\boldsymbol{\eta}_d = [n_d, e_d, d_d, \eta_d, \varepsilon_{1d}, \varepsilon_{2d}, \varepsilon_{3d}]^\top \quad (14.46)$$

where $\mathbf{p}_d = [n_d, e_d, d_d]^\top$ is the NED positions, while attitude is specified using Euler angles $\boldsymbol{\Theta}_d = [\phi_d, \theta_d, \psi_d]^\top$ or unit quaternions $\mathbf{q}_d = [\eta_d, \varepsilon_{1d}, \varepsilon_{2d}, \varepsilon_{3d}]^\top$, respectively.

Position and Euler Angle Feedback

As shown in Section 7.3, the nonlinearities can be canceled out by choosing the control law as:

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{M}\mathbf{a}_1^b + \mathbf{n}_1(\boldsymbol{\nu}, \boldsymbol{\Theta}) && (14.47)\\ \mathbf{a}_1^b &= \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta})[\mathbf{a}_1^n - \dot{\mathbf{J}}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\nu}] && (14.48) \end{aligned}$$

where $\mathbf{a}_1^b \in \mathbb{R}^6$ and $\mathbf{a}_1^n \in \mathbb{R}^6$ are the commanded accelerations in the *b*- and *n*-frame, respectively. This transformation results in the decoupled system:

$$\begin{aligned} \ddot{\boldsymbol{\eta}}_1 &= \mathbf{J}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\nu}\\ &= \mathbf{J}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\mathbf{a}_1^b + \dot{\mathbf{J}}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\nu}\\ &= \mathbf{a}_1^n && (14.49) \end{aligned}$$

The commanded acceleration $\mathbf{a}_1^n$ is chosen as a PID-control law with acceleration feedforward:

$$\mathbf{a}_1^n = \ddot{\boldsymbol{\eta}}_d - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}}_1 - \mathbf{K}_p\tilde{\boldsymbol{\eta}}_1 - \mathbf{K}_i\int_0^t \tilde{\boldsymbol{\eta}}_1(\tau)d\tau \tag{14.50}$$

where $\mathbf{K}_p \in \mathbb{R}^{6\times6}, \mathbf{K}_d \in \mathbb{R}^{6\times6}$, and $\mathbf{K}_i \in \mathbb{R}^{6\times6}$ are positive definite matrices. The linear error dynamics becomes:

$$\mathbf{z}_1^{(3)} + \mathbf{K}_d\ddot{\mathbf{z}}_1 + \mathbf{K}_p\dot{\mathbf{z}}_1 + \mathbf{K}_i\mathbf{z}_1 = \mathbf{0} \tag{14.51}$$

where:

$$\mathbf{z}_1 = \int_0^t \tilde{\boldsymbol{\eta}}_1(\tau)d\tau \tag{14.52}$$

Consequently $\mathbf{K}_p$, $\mathbf{K}_d$, and $\mathbf{K}_i$ can be found by using a pole placement algorithm.

Position and Unit Quaternion Feedback

When using unit quaternions the transformation matrix $\mathbf{J}_q(\mathbf{q}) \in \mathbb{R}^{7\times6}$ cannot be inverted. However, the 6×6 matrix:

$$\mathbf{I}_q = \mathbf{J}_q^\top(\mathbf{q})\mathbf{J}_q(\mathbf{q}) = \begin{bmatrix} \mathbf{R}_n^b(\mathbf{q})^\top\mathbf{R}_n^b(\mathbf{q}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{T}_q(\mathbf{q})^\top\mathbf{T}_q(\mathbf{q}) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \frac{1}{4}\mathbf{I}_{3\times3} \end{bmatrix} \tag{14.53}$$

has a nonsingular constant inverse $\mathbf{I}_q^{-1}$; see the properties of $\mathbf{R}_n^b(\mathbf{q})$ and $\mathbf{T}_q(\mathbf{q})$ in Section 2.2.2. Consider:

$$\begin{aligned} \ddot{\boldsymbol{\eta}}_2 &= \mathbf{J}_q(\mathbf{q})\dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_q(\mathbf{q})\boldsymbol{\nu} \\ &= \mathbf{J}_q(\mathbf{q})\mathbf{a}_2^b + \dot{\mathbf{J}}_q(\mathbf{q})\boldsymbol{\nu} \\ &= \mathbf{a}_2^n \end{aligned} \tag{14.54}$$

Pre-multiplication of this expression with $\mathbf{J}_q^\top(\mathbf{q})$ gives:

$$\mathbf{I}_q\mathbf{a}_2^b + \mathbf{J}_q^\top(\mathbf{q})\dot{\mathbf{J}}_q(\mathbf{q})\boldsymbol{\nu} = \mathbf{J}_q^\top(\mathbf{q})\mathbf{a}_2^n \tag{14.55}$$

where $\mathbf{a}_2^b \in \mathbb{R}^7$ can be computed since $\mathbf{I}_q^{-1}$ exists. Consequently, the feedback linearizing controller based on unit quaternions becomes:

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{a}_2^b + \mathbf{n}_2(\boldsymbol{\nu}, \mathbf{q}) \tag{14.56}$$

$$\mathbf{a}_2^b = \mathbf{I}_q^{-1}\left(\mathbf{J}_q^\top(\mathbf{q})\mathbf{a}_2^n - \mathbf{J}_q^\top(\mathbf{q})\dot{\mathbf{J}}_q(\mathbf{q})\boldsymbol{\nu}\right) \tag{14.57}$$

The commanded acceleration $\mathbf{a}^n \in \mathbb{R}^7$ is chosen as a PID-control law with acceleration feedforward:

$$\mathbf{a}^n = \ddot{\boldsymbol{\eta}}_d - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}}_2 - \mathbf{K}_p\tilde{\boldsymbol{\eta}}_2 - \mathbf{K}_i\int_0^t \tilde{\boldsymbol{\eta}}_2(\tau)d\tau \tag{14.58}$$

where $\mathbf{K}_p \in \mathbb{R}^{7\times7}, \mathbf{K}_d \in \mathbb{R}^{7\times7}$,and $\mathbf{K}_i \in \mathbb{R}^{7\times7}$ are positive definite matrices. The linear error dynamics becomes:

$$\mathbf{z}_2^{(3)} + \mathbf{K}_d\ddot{\mathbf{z}}_2 + \mathbf{K}_p\dot{\mathbf{z}}_2 + \mathbf{K}_i\mathbf{z}_2 = \mathbf{0} \tag{14.59}$$

where:

$$\mathbf{z}_2 = \int_0^t \tilde{\boldsymbol{\eta}}_2(\tau)d\tau \tag{14.60}$$

14.2.2 Adaptive Feedback Linearization

The 6 DOF feedback linearization controllers of the previous section can be modified to include a *parameter adaptation law* for estimation of the unknown model parameters. For the Euler angle feedback representation, the adaptive control law is (see Section 7.3.3):

$$\boldsymbol{\tau} = \boldsymbol{\Phi}(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta})\hat{\boldsymbol{\theta}} \tag{14.61}$$

$$\mathbf{a}_1^b = \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta})[\mathbf{a}_1^n - \dot{\mathbf{J}}_{\boldsymbol{\Theta}}(\boldsymbol{\Theta})\boldsymbol{\nu}] \tag{14.62}$$

$$\mathbf{a}_1^n = \ddot{\boldsymbol{\eta}}_d - \mathbf{K}_d\dot{\tilde{\boldsymbol{\eta}}}_1 - \mathbf{K}_p\tilde{\boldsymbol{\eta}}_1 \tag{14.63}$$

where $\hat{\boldsymbol{\theta}}$ is a vector of parameter estimate updates given by:

$$\dot{\hat{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma}\boldsymbol{\Phi}^\top(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta})\mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta})\mathbf{y}, \qquad \boldsymbol{\Gamma} = \boldsymbol{\Gamma}^\top > 0 \tag{14.64}$$

$$\mathbf{y} = c_0\tilde{\boldsymbol{\eta}} + c_1\dot{\tilde{\boldsymbol{\eta}}} \tag{14.65}$$

The constants c_0 and c_1 must satisfy the following inequalities:

$$\text{(i)} \quad (c_0\mathbf{K}_d + c_1\mathbf{K}_p)c_1 > c_0^2\mathbf{I}$$
$$\text{(ii)} \quad 2c_0\mathbf{K}_p > \beta\mathbf{I}$$
$$\text{(iii)} \quad 2(c_1\mathbf{K}_d - c_0\mathbf{I}) > \beta\mathbf{I}$$

The regressor matrix $\boldsymbol{\Phi}(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta})$ is defined by:

$$\mathbf{M}\mathbf{a}_1^b + \mathbf{n}_1(\boldsymbol{\nu}, \boldsymbol{\Theta}) = \boldsymbol{\Phi}(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta})\boldsymbol{\theta} \tag{14.66}$$

where $\boldsymbol{\theta}$ is a vector of unknown model parameters.

Example 14.1 (Adaptive Control of Underwater Vehicles)
Consider the longitudinal underwater vehicle dynamics:

$$\begin{bmatrix} m - X_{\dot{u}} & -X_{\dot{w}} & mz_g - X_{\dot{q}} \\ -X_{\dot{w}} & m - Z_{\dot{w}} & -mx_g - Z_{\dot{q}} \\ mz_g - X_{\dot{q}} & -mx_g - Z_{\dot{q}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -X_u & -X_w & -X_q \\ -Z_u & -Z_w & -Z_q \\ -M_u & -M_w & -M_q \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -(m - X_{\dot{u}})u \\ 0 & (Z_{\dot{w}} - X_{\dot{u}})u & mx_g u \end{bmatrix} \begin{bmatrix} u \\ w \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ W\, BG_z \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_3 \\ \tau_5 \end{bmatrix}$$

This model can be written in regressor form:

$$\mathbf{M}\mathbf{a}_1^b + \mathbf{n}_1(\boldsymbol{\nu}, \boldsymbol{\Theta}) = \boldsymbol{\Phi}(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta})\boldsymbol{\theta} \tag{14.67}$$

where $\mathbf{a}_1^b = [a_1^b, a_2^b, a_3^b]^\top$ *and* $\boldsymbol{\nu} = [u, w, q]^\top$. *In order to do this, we collect the mass, damping, and gravitational/buoyancy parameters into three vectors* $\boldsymbol{\theta}_\mathbf{m}^\top, \boldsymbol{\theta}_\mathbf{d}^\top$, *and* $\boldsymbol{\theta}_g^\top$ *such that:*

$$\boldsymbol{\theta} = [\boldsymbol{\theta}_\mathbf{m}^\top, \boldsymbol{\theta}_\mathbf{d}^\top, \boldsymbol{\theta}_g^\top]^\top \tag{14.68}$$

where:

$$\begin{aligned} \boldsymbol{\theta}_m &= [m - X_{\dot{u}}, -X_{\dot{w}}, mz_g - X_{\dot{q}}, m - Z_{\dot{w}}, -mx_g - Z_{\dot{q}}, I_y - M_{\dot{q}}, mx_g]^\top \\ \boldsymbol{\theta}_d &= [-X_u, -X_w, -X_q, -Z_u, -Z_w, -Z_q, -M_u, -M_w, -M_q]^\top \\ \boldsymbol{\theta}_g &= [W\, BG_z] \end{aligned}$$

Consequently,

$$\boldsymbol{\Phi}(\mathbf{a}_1^b, \boldsymbol{\nu}, \boldsymbol{\Theta}) = [\boldsymbol{\Phi}_m(\mathbf{a}_1^b, \boldsymbol{\nu}), \boldsymbol{\Phi}_d(\boldsymbol{\nu}), \boldsymbol{\Phi}_g(\boldsymbol{\Theta})] \tag{14.69}$$

where:

$$\begin{aligned} \boldsymbol{\Phi}_m(\mathbf{a}_1^b, \boldsymbol{\nu}) &= \begin{bmatrix} a_1^b & a_2^b & a_3^b & 0 & 0 & 0 & 0 \\ -u & a_1^b & 0 & a_2^b & a_3^b & 0 & 0 \\ u & 0 & a_1^b & -u & a_2^b & a_3^b & u \end{bmatrix} \\ \boldsymbol{\Phi}_d(\boldsymbol{\nu}) &= \begin{bmatrix} u & v & w & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & u & v & w & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u & v & w \end{bmatrix} \\ \boldsymbol{\Phi}_g(\boldsymbol{\Theta}) &= \begin{bmatrix} 0 \\ 0 \\ \sin\theta \end{bmatrix} \end{aligned}$$

The parameter update laws are:

$$\dot{\hat{\boldsymbol{\theta}}}_m = -\boldsymbol{\Gamma}_m \boldsymbol{\Phi}_m^\top(\mathbf{a}_1^b, \boldsymbol{\nu}) \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta}) \mathbf{y} \tag{14.70}$$

$$\dot{\hat{\boldsymbol{\theta}}}_d = -\boldsymbol{\Gamma}_d \boldsymbol{\Phi}_d^\top(\boldsymbol{\nu}) \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta}) \mathbf{y} \tag{14.71}$$

$$\dot{\hat{\boldsymbol{\theta}}}_g = -\boldsymbol{\Gamma}_g \boldsymbol{\Phi}_g^\top(\boldsymbol{\Theta}) \mathbf{J}_{\boldsymbol{\Theta}}^{-1}(\boldsymbol{\Theta}) \mathbf{y} \tag{14.72}$$

14.3 Exercises

Exercise 14.1 *Derive a feedback linearizing controller where the pitch and roll modes are left uncontrolled. Assume that* $\tau_4 = \tau_5 = 0$ *and let* $\mathbf{u} \in \mathbb{R}^4$ *such that:*

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{u}, \quad \textit{where} \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Hint: define an output mapping $\mathbf{y} = \mathbf{C}\boldsymbol{\eta}$ *where* $\mathbf{y} = [n, e, d, \psi]^\top$ *and let* $\mathbf{y}_d = [n_d, e_d, d_d, \psi_d]^\top$.

Exercise 14.2 *Consider the underwater vehicle model:*

$$\begin{aligned} \dot{n} &= u\cos\theta + w\sin\theta \\ \dot{d} &= -u\sin\theta + w\cos\theta \\ \dot{\theta} &= q \end{aligned}$$

where n *and* d *are the North-Down positions,* θ *is the pitch angle and:*

$$\begin{bmatrix} m - Z_{\dot{w}} & -Z_{\dot{q}} \\ -M_{\dot{w}} & I_y - M_{\dot{q}} \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -Z_w & -Z_q \\ -M_w & -M_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -M_\theta \end{bmatrix} \begin{bmatrix} z \\ \theta \end{bmatrix} = \begin{bmatrix} \tau_3 \\ \tau_5 \end{bmatrix}$$

Assume that the speed dynamics is:

$$(m - X_{\dot{u}})\dot{u} - X_u u - X_{u|u|} u\,|u| = \tau_1$$

Derive a feedback linearizing controller for trajectory tracking and let (n_d, e_d, θ_d) *represent the desired trajectory.*

Exercise 14.3 *Find the regressor and parameter vector for the nonlinear model in Example 14.2.*

Exercise 14.4 *Derive an adaptive feedback linearizing controller for position and attitude control using the unit quaternion representation.*

Part V

Appendices

Appendix A

Nonlinear Stability Theory

This chapter briefly review some useful results from nonlinear stability theory. This includes:

- Lyapunov stability of nonlinear *autonomous* systems $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, that is systems where $\mathbf{f}(\mathbf{x})$ does not explicitly depend on the time t.
- Lyapunov stability of nonlinear *nonautonomous* systems $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},t)$, that is systems where $\mathbf{f}(\mathbf{x},t)$ does depend on t explicitly.

A.1 Lyapunov Stability for Autonomous Systems

Before stating the main Lyapunov theorems for *autonomous* systems, the concepts of stability and convergence are briefly reviewed (Khalil 2002).

A.1.1 Stability and Convergence

Consider the nonlinear time-invariant system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{A.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is assumed to be *locally Lipschitz* in $\mathbf{x}$, that is for each point $\mathbf{x} \in D \subset \mathbb{R}^n$, there exists a neighborhood $D_0 \in D$ such that:

$$\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{D}_0 \tag{A.2}$$

where L is called the Lipschitz constant on D_0.

Let $\mathbf{x}_e$ denote the equilibrium point of (A.1) given by:

$$\mathbf{f}(\mathbf{x}_e) = \mathbf{0} \tag{A.3}$$

Table A.1: Classification of theorems for stability and convergence.

Autonomous systems	$V > 0, \dot{V} < 0$	Lyapunov's direct method	GAS/GES
	$V > 0, \dot{V} \leq 0$	Krasovskii-LaSalle's theorem	GAS
Non-autonomous systems	$V > 0, \dot{V} < 0$	LaSalle-Yoshizawa's theorem	UGAS
	$V > 0, \dot{V} \leq 0$	Matrosov's theorem	UGAS
	$V \geq 0, \dot{V} \leq 0$	Barbalat's lemma	convergence

The solutions $\mathbf{x}(t)$ of (A.1) are:

- *bounded*, if there exist a non-negative function $0 < \gamma(\mathbf{x}(0)) < \infty$, such that:

$$\|\mathbf{x}(t)\| \leq \gamma(\mathbf{x}(0)), \quad \forall t \geq 0 \tag{A.4}$$

In addition, the equilibrium point $\mathbf{x}_e$ of (A.1) is:

- *stable*, if, for each $\epsilon > 0$, there exists a $\delta(\epsilon) > 0$ such that:

$$\|\mathbf{x}(0)\| < \delta(\epsilon) \Rightarrow \|\mathbf{x}(t)\| < \epsilon, \quad \forall t \geq 0 \tag{A.5}$$

- *unstable*, if it is not stable.
- *attractive*, if there for each $r > 0, \epsilon > 0$ exists a $T(r, \epsilon) > 0$ such that:

$$\|\mathbf{x}(0)\| \leq r \Rightarrow \|\mathbf{x}(t)\| \leq \epsilon, \quad \forall t \geq T(r, \epsilon) \tag{A.6}$$

  Attractivity implies convergence, that is $\lim_{t \to \infty} \|\mathbf{x}(t)\| = \mathbf{0}$.
- *(locally) asymptotically stable (AS)*, if the equilibrium point $\mathbf{x}_e$ is stable and attractive.
- *globally stable (GS)*, if the equilibrium point $\mathbf{x}_e$ is stable and $\delta(\epsilon)$ can be chosen to satisfy $\lim_{\epsilon \to \infty} \delta(\epsilon) = \infty$.
- *global asymptotically stable (GAS)*, if the equilibrium point $\mathbf{x}_e$ is stable for all $\mathbf{x}(0)$ (region of attraction $\mathbb{R}^n$).
- *(locally) exponentially stable (ES)*, if there exist positive constants α, λ and r such that:

$$\|\mathbf{x}(0)\| < r \Rightarrow \|\mathbf{x}(t)\| < \alpha \exp(-\lambda t) \|\mathbf{x}(0)\|, \ \forall t \geq 0 \tag{A.7}$$

- *globally exponentially stable (GES)*, if there exist positive constants α, λ and r such that for all $\mathbf{x}(0)$ (region of attraction $\mathbb{R}^n$).

$$\|\mathbf{x}(t)\| < \alpha \exp(-\lambda t) \|\mathbf{x}(0)\|, \ \forall t \geq 0 \tag{A.8}$$

Different theorems for investigation of stability and convergence will now be presented. A guideline for which theorem that should be applied is given in Table A.1 whereas the different theorems are listed in the forthcoming sections.

Notice that for nonautonomous systems GAS is replaced by *uniform global asymptotic stability* (UGAS) since uniformity is a necessary requirement in the case of time-varying nonlinear systems.

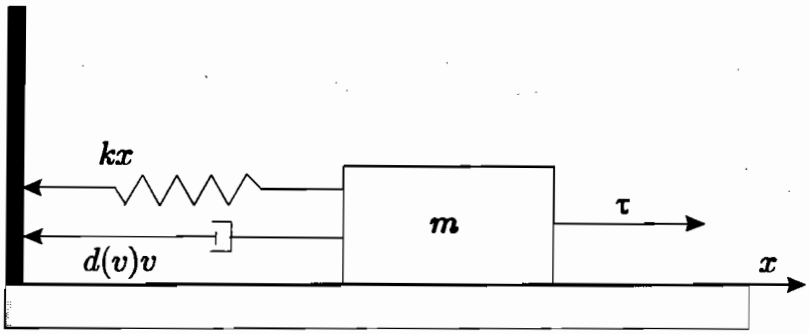


Figure A.1: Mass-damper-spring system.

A.1.2 Lyapunov's Direct Method

Theorem A.1 (Lyapunov's Direct Method)
Let $\mathbf{x}_e$ be the equilibrium point of (A.1) and assume that $\mathbf{f}(\mathbf{x})$ is locally Lipschitz in $\mathbf{x}$. Let $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ be a continuously differentiable function $V(\mathbf{x})$ satisfying:

$$\bullet V(\mathbf{x}) > 0 \text{ (positive definite) and } V(0) = 0 \tag{A.9}$$

$$\bullet \dot{V}(\mathbf{x}) = \frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) \leq -W(\mathbf{x}) \leq 0 \tag{A.10}$$

$$\bullet V(\mathbf{x}) \rightarrow \infty \text{ as } \|\mathbf{x}\| \rightarrow \infty \text{ (radially unbounded)} \tag{A.11}$$

then the equilibrium point $\mathbf{x}_e$ is GS if $W(\mathbf{x}) \geq 0$ (positive semi-definite) and GAS if $W(\mathbf{x}) > 0$ (positive definite) for all $\mathbf{x} \neq \mathbf{0}$.

Proof. See Khalil (2002) or Lyapunov (1907). ■

The requirement that $W(\mathbf{x}) > 0$ such that $\dot{V}(\mathbf{x}) < 0$ is in many cases difficult to satisfy. This is illustrated in the following example.

Example A.1 (Stability of a Mass-Damper-Spring System)
Consider the nonlinear mass-damper-spring system:

$$\dot{x} = v \tag{A.12}$$

$$m\dot{v} + d(v)v + kx^2 = 0 \tag{A.13}$$

where $m > 0, d(v) > 0, \forall v$ and $k > 0$, see Figure A.1. Let us choose $V(\mathbf{x})$ as the sum of kinetic energy $\frac{1}{2}mv^2$ and potential energy $\frac{1}{2}kx^2$ such that:

$$V(\mathbf{x}) = \frac{1}{2}\left(mv^2 + kx^2\right) = \frac{1}{2}\mathbf{x}^\top \begin{bmatrix} m & 0 \\ 0 & k \end{bmatrix} \mathbf{x} \tag{A.14}$$

where $\mathbf{x} = [v, x]^\top$, *results in*

$$\begin{aligned} \dot{V}(\mathbf{x}) &= mv\dot{v} + kx\dot{x} \\ &= v(m\dot{v} + kx) \\ &= -d(v)v^2 \\ &= -\mathbf{x}^\top \begin{bmatrix} d(v) & 0 \\ 0 & 0 \end{bmatrix} \mathbf{x} \end{aligned} \tag{A.15}$$

Hence, only stability can be concluded from Theorem A.1, since $\dot{V}(\mathbf{x}) = 0$ *for all* $v = 0$. *However, GAS can in many cases also be proven for systems with a negative semi-definite* $\dot{V}(\mathbf{x})$ *thanks to the invariant set theorem of* Krasovskii–LaSalle, *see LaSalle and Lefschetz (1961) and LaSalle (1966).*

A.1.3 Krasovskii–LaSalle's Theorem

The theorem of Krasovskii–LaSalle can be used to check a nonlinear *autonomous* system for GAS in the case of a negative semi-definite $\dot{V}(\mathbf{x})$.

Theorem A.2 (Krazovskii–LaSalle's Theorem)
Let $V : \mathbb{R}^n \to \mathbb{R}_+$ *be a continuously differentiable positive definite function such that:*

$$V(\mathbf{x}) \rightarrow \infty \text{ as } \|\mathbf{x}\| \rightarrow \infty \tag{A.16}$$

$$\dot{V}(\mathbf{x}) \leq 0, \forall \mathbf{x} \tag{A.17}$$

Let Ω *be the set of all points where* $\dot{V}(\mathbf{x}) = 0$, *that is:*

$$\Omega = \left\{ \mathbf{x} \in \mathbb{R}^n | \dot{V}(\mathbf{x}) = 0 \right\} \tag{A.18}$$

and M *be the largest invariant set in* Ω, *then all solutions* $\mathbf{x}(t)$ *converge to* M. *If* $M = \{\mathbf{x}_e\}$ *then the equilibrium point* $\mathbf{x}_e$ *of (A.1) is GAS.*

Proof. See LaSalle (1966). ■

Example A.2 (Cont. Example A.1: Stability of a Mass-Damper-Spring System)
Again consider the mass-damper-spring system of Example A.1. The set Ω *is found by requiring that*

$$\dot{V}(\mathbf{x}) = -d(v)v^2 \equiv 0 \tag{A.19}$$

which is true for $v = 0$. *Therefore:*

$$\Omega = \{(x \in \mathbb{R}, v = 0)\} \tag{A.20}$$

Now, $v = 0$, *implies that* $m\dot{v} = -kx$, *which is non-zero when* $x \neq 0$. *Hence, the system cannot get "stuck" at a point other than* $x = 0$. *Since the equilibrium point of the mass-damper-spring system is* $(x, v) = (0, 0)$, *the largest invariant set* M *in* Ω *contains only one point, namely* $(x, v) = (0, 0)$. *Hence, the equilibrium point of (A.1) is GAS according to Theorem A.2.*

A.1.4 Global Exponential Stability

The following theorem is useful to guarantee exponential convergence.

Theorem A.3 (Global Exponential Stability)
Let $\mathbf{x}_e$ be the equilibrium point of (A.1) and assume that $\mathbf{f}(\mathbf{x})$ is locally Lipschitz in $\mathbf{x}$. Let $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ be a continuously differentiable and radially unbounded function satisfying:

$$V(\mathbf{x}) = \mathbf{x}^\top \mathbf{P}\mathbf{x} > 0, \quad \forall \mathbf{x} \neq \mathbf{0} \tag{A.21}$$

$$\dot{V}(\mathbf{x}) \leq -\mathbf{x}^\top \mathbf{Q}\mathbf{x} < 0, \quad \forall \mathbf{x} \neq \mathbf{0} \tag{A.22}$$

with constant matrices $\mathbf{P} = \mathbf{P}^\top > 0$ and $\mathbf{Q} = \mathbf{Q}^\top > 0$, then the equilibrium point $\mathbf{x}_e$ is GES and the state vector satisfies:

$$\|\mathbf{x}(t)\|_2 \leq \sqrt{\frac{\lambda_{\max}(\mathbf{P})}{\lambda_{\min}(\mathbf{P})}} \exp(-\alpha t) \|\mathbf{x}(0)\|_2 \tag{A.23}$$

where

$$\alpha = \frac{\lambda_{\min}(\mathbf{Q})}{2\lambda_{\max}(\mathbf{P})} > 0 \tag{A.24}$$

is a bound on the convergence rate.

Proof. Since $V(\mathbf{x})$ is bounded by:

$$0 < \lambda_{\min}(\mathbf{P}) \|\mathbf{x}(t)\|_2^2 \leq V(\mathbf{x}) \leq \lambda_{\max}(\mathbf{P}) \|\mathbf{x}(t)\|_2^2, \quad \forall \mathbf{x} \neq \mathbf{0} \tag{A.25}$$

it is seen that:

$$-\|\mathbf{x}(t)\|_2^2 \leq -\frac{1}{\lambda_{\max}(\mathbf{P})} V(\mathbf{x}) \tag{A.26}$$

Hence, it follows from (A.22) that:

$$\begin{aligned} \dot{V}(\mathbf{x}) &\leq -\mathbf{x}^\top \mathbf{Q}\mathbf{x} \\ &\leq -\lambda_{\min}(\mathbf{Q}) \|\mathbf{x}(t)\|_2^2 \\ &\leq -\underbrace{\frac{\lambda_{\min}(\mathbf{Q})}{\lambda_{\max}(\mathbf{P})}}_{2\alpha} V(\mathbf{x}) \end{aligned} \tag{A.27}$$

Integration of $\dot{V}(\mathbf{x}(t))$ yields:

$$V(\mathbf{x}(t)) \leq \exp(-2\alpha t) V(\mathbf{x}(0)) \tag{A.28}$$

Finally, (A.25) implies:

$$\lambda_{\min}(\mathbf{P}) \|\mathbf{x}(t)\|_2^2 \leq \exp(-2\alpha t) \lambda_{\max}(\mathbf{P}) \|\mathbf{x}(0)\|_2^2 \tag{A.29}$$

$$\|\mathbf{x}(t)\|_2 \leq \sqrt{\frac{\lambda_{\max}(\mathbf{P})}{\lambda_{\min}(\mathbf{P})}} e^{-\alpha t} \|\mathbf{x}(0)\|_2 \tag{A.30}$$

This shows that $\|\mathbf{x}(t)\|_2$ will converge exponentially to zero with convergence rate α. ■

A.2 Lyapunov Stability of Nonautonomous Systems

In this section several useful theorems for convergence and stability of time-varying nonlinear systems:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},t),\ \mathbf{x}(0) = \mathbf{x}_0 \tag{A.31}$$

where $\mathbf{x} \in \mathbb{R}^n$, $t \in \mathbb{R}_+$ and $\mathbf{f}: \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$ is assumed to be *locally Lipschitz* in $\mathbf{x}$ and uniformly in t are briefly reviewed.

A.2.1 Barbalat's Lemma

Lemma A.1 (Barbălat's Lemma)
Let $\phi: \mathbb{R}_+ \to \mathbb{R}$ be a uniformly continuous function and suppose that $\lim_{t\to\infty} \int_0^t \phi(\tau)d\tau$ exists and is finite, then:

$$\lim_{t\to\infty} \phi(t) = 0 \tag{A.32}$$

Proof. See Barbălat (1959). ■

Notice that *Barbălat's lemma* only guarantees *global convergence*. This result is particular useful if there exists a uniformly continuous function $V: \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ satisfying:

i) $V(\mathbf{x},t) \geq 0$
ii) $\dot{V}(\mathbf{x},t) \leq 0$
iii) $\dot{V}(\mathbf{x},t)$ is uniformly continuous

Hence, according to Barbălat's lemma $\lim_{t\to\infty} \dot{V}(\mathbf{x},t) = 0$. The requirement that $\dot{V}$ should be uniformly continuous can easily be checked by using:

$$\ddot{V}(\mathbf{x},t) \text{ is bounded} \implies \dot{V}(\mathbf{x},t) \text{ is uniformly continuous}$$

A.2.2 LaSalle-Yoshizawa's Theorem

For nonautonomous systems the following theorem of LaSalle (1966) and Yoshizawa (1968) is quite useful:

Theorem A.4 (LaSalle–Yoshizawa's Theorem)
Let $\mathbf{x}_e = \mathbf{0}$ be the equilibrium point of (A.31) and assume that $\mathbf{f}(\mathbf{x},t)$ is locally Lipschitz in $\mathbf{x}$. Let $V: \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}_+$ be a continuously differentiable function $V(\mathbf{x},t)$ satisfying:

$$\bullet V(\mathbf{x},t) > 0 \text{ (positive definite) and } V(0) = 0 \tag{A.33}$$

$$\bullet \dot{V}(\mathbf{x},t) = \frac{\partial V(\mathbf{x},t)}{\partial t} + \frac{\partial V(\mathbf{x},t)}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x},t) \leq -W(\mathbf{x}) \leq 0 \tag{A.34}$$

$$\bullet V(\mathbf{x},t) \to \infty \text{ as } \|\mathbf{x}\| \to \infty \quad \text{(radially unbounded)} \tag{A.35}$$

where $W(\mathbf{x})$ is a continuous function, then all solutions $\mathbf{x}(t)$ of (A.31) are uniformly globally bounded and:

$$\lim_{t\to\infty} W(\mathbf{x}(t)) = 0 \tag{A.36}$$

In addition, if $W(\mathbf{x}) > 0$ (positive definite), then the equilibrium point $\mathbf{x}_e = \mathbf{0}$ of (A.31) is UGAS.

Proof. See LaSalle (1966) and Yoshizawa (1968). ■

A.2.3 Matrosov's Theorem

Nonautonomous systems where $\dot{V}(\mathbf{x},t) \leq 0$ are UGAS if Matrosov's Theorem Matrosov (1962) is satisfied. The version presented here is taken from Teel (2002).

Definition A.1 (Class $\mathcal{K}$ Function)
A continuous function $\alpha : [0,a) \rightarrow [0,\infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. It is said to belong to class $\mathcal{K}_\infty$ if $a = \infty$ and $\alpha(r) \rightarrow \infty$ as $r \rightarrow \infty$.

Given two constants $0 \leq \delta \leq \Delta < \infty$, and $\mathcal{H}(\delta,\Delta) := \{\mathbf{x} \in \mathbb{R}^n : \delta \leq |\mathbf{x}| \leq \Delta\}$ then Matrosov's theorem can be stated according to:

Theorem 2 (Matrosov's Theorem)
Consider the system:

$$\dot{\mathbf{x}} = \mathbf{f}(t,\mathbf{x}), \quad \mathbf{x}(0) = \mathbf{x}_0,\ \mathbf{x} \in \mathbb{R}^n \tag{A.37}$$

If there for this system exist:
- a locally Lipschitz function V: $\mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}_+$
- a continuous positive semi-definite function $U : \mathbb{R}^n \rightarrow \mathbb{R}_+$
- functions $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$
such that

1. $\alpha_1(\|\mathbf{x}\|) \leq V(t,\mathbf{x}) \leq \alpha_2(\|\mathbf{x}\|) \quad \forall\,(t,x) \in \mathbb{R} \times \mathbb{R}^n$
2. $\dot{V}(t,\mathbf{x}) \leq -U(\mathbf{x})$ *for almost all* $(t,\mathbf{x}) \in \mathbb{R} \times \mathbb{R}^n$

and for each $0 < \delta \leq \Delta$ and $\mathcal{H}(0,\Delta) \subseteq \mathbb{R}^n$ there exists:
- a locally Lipschitz function $W : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$
- a continuous function $Y : \mathbb{R}^n \rightarrow \mathbb{R}$
- strictly positive numbers $\varepsilon_1, \varepsilon_2, \psi > 0$
such that:

3. $\max\{|W(t,\mathbf{x})|, |Y(\mathbf{x})|\} \leq \psi \ \ \forall\,(t,x) \in \mathbb{R} \times \mathcal{H}(0,\Delta)$
4. $\dot{W}(t,\mathbf{x}) \leq Y(\mathbf{x})$ *for all* $(t,\mathbf{x}) \in \mathbb{R} \times \mathbb{R}^n$.
5. $\mathbf{x} \in \mathcal{H}(\delta,\Delta) \cap \{\mathbf{x} : U(\mathbf{x}) \leq \varepsilon_1\} \Longrightarrow Y(\mathbf{x}) \leq -\varepsilon_2$.

Then, the origin of (A.37) is UGAS.

Matrosov's theorem has been successfully applied to prove UGAS of nonlinear observers for integration of GPS and INS by Vik and Fossen (2002).

Remark: If the system (A.37) is time-invariant, that is $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, then Condition 5 can be replaced by:

5. $\mathbf{x} \in \mathcal{H}(\delta,\Delta) \cap \{\mathbf{x} : U(\mathbf{x}) = 0\} \Longrightarrow Y(\mathbf{x}) < 0$

A.2.4 UGAS when Backstepping with Integral Action

When designing industrial control systems it is important to include integral action in the control law in order to compensate for slowly-varying and constant disturbances. This is necessary to avoid steady-state errors both in regulation and tracking. The integral part of the controller can be provided by using *adaptive backstepping* (Krstic *et al.* 1995) under the assumption of constant disturbances, see Section 7.4.4. Unfortunately, the resulting error dynamics in this case often becomes nonautonomous, which again implies that *Krasovskii–LaSalle's theorem* cannot be used. An alternative theorem for this case will be stated by considering the nonlinear system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, t) \tag{A.38}$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^n$, and $\boldsymbol{\theta} \in \mathbb{R}^p$ ($p \leq n$) is a constant *unknown* parameter vector. Furthermore, assume that there exists an adaptive control law:

$$\mathbf{u} = \mathbf{u}(\mathbf{x}, \mathbf{x}_d, \hat{\boldsymbol{\theta}}) \tag{A.39}$$

$$\dot{\hat{\boldsymbol{\theta}}} = \boldsymbol{\phi}(\mathbf{x}, \mathbf{x}_d) \tag{A.40}$$

where $\mathbf{x}_d \in C^r$ and $\hat{\boldsymbol{\theta}} \in \mathbb{R}^p$, such that the error dynamics can be written:

$$\dot{\mathbf{z}} = \mathbf{h}(\mathbf{z}, t) + \mathbf{B}(t)\tilde{\boldsymbol{\theta}} \tag{A.41}$$

$$\dot{\tilde{\boldsymbol{\theta}}} = -\mathbf{P}\mathbf{B}(t)^\top \left(\frac{\partial W(\mathbf{z}, t)}{\partial \mathbf{z}}\right)^\top, \quad \mathbf{P} = \mathbf{P}^\top > 0 \tag{A.42}$$

where $W(\mathbf{z}, t)$ is a suitable C^1 function and $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$ is the parameter estimation error. The parameter estimate $\hat{\boldsymbol{\theta}}$ can be used to compensate for a constant disturbance – i.e. integral action. Hence, the conditions in the following theorem can be used to establish UGAS when backstepping with integral action. The conditions are based on Loria *et al.* (1999) alternatively Fossen *et al.* (2001). This can also be proven by applying Matrosov's theorem. Bias estimation using Matrosov's theorem is discussed in detail by Vik and Fossen (2002).

Theorem A.5 (UGAS/LES when Backstepping with Integral Action)
The origin of the system (A.41)–(A.42) is UGAS if $\mathbf{B}^\top(t)\mathbf{B}(t)$ *is invertible for all* t, $\mathbf{P} = \mathbf{P}^\top > 0$, *there exist a continuous, non-decreasing function* $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ *such that*

$$\max\left\{\|\mathbf{h}(\mathbf{z}, t)\|, \left\|\frac{\partial W(\mathbf{z}, t)}{\partial \mathbf{z}}\right\|\right\} \leq \rho(\|\mathbf{z}\|)\,\|\mathbf{z}\| \tag{A.43}$$

and there exist class-$\mathcal{K}_\infty$ *functions* α_1 *and* α_2 *and a strictly positive real number* $c > 0$ *such that* $W(\mathbf{z}, t)$ *satisfy:*

$$\alpha_1(\|\mathbf{z}\|) \leq W(\mathbf{z}, t) \leq \alpha_2(\|\mathbf{z}\|) \tag{A.44}$$

$$\frac{\partial W(\mathbf{z}, t)}{\partial t} + \frac{\partial W(\mathbf{z}, t)}{\partial \mathbf{z}}\mathbf{h}(\mathbf{z}, t) \leq -c\,\|\mathbf{z}\|^2 . \tag{A.45}$$

If, in addition, $\alpha_2(s) \propto s^2$ *for sufficiently small* s *then the origin is LES.*

Proof. See Fossen *et al.* (2001). ■

Theorem A.5 implies that both $\mathbf{z} \to \mathbf{0}$ and $\tilde{\theta} \to 0$ when $t \to \infty$. The following example illustrates how a UGAS integral controller can be derived:

Example A.3 (UGAS Integral Controller)
Consider the non-autonomous system:

$$\dot{x} = -a(t)x + \theta + u \tag{A.46}$$

$$u = -K_p x - \hat{\theta} \tag{A.47}$$

$$\dot{\hat{\theta}} = px \tag{A.48}$$

where $0 < a(t) \leq a_{\max}, \theta =$ *constant,* $K_p > 0$, *and* $p > 0$. *This is a PI-controller since* $u = -K_p x - p\int_0^t x(\tau)d\tau$. *Choosing* $z = x$, *the error dynamics can be written:*

$$\dot{z} = -[a(t) + K_p]z - \tilde{\theta} \tag{A.49}$$

$$\dot{\tilde{\theta}} = pz \tag{A.50}$$

which is in the form (A.41)–(A.42) with $W(z) = \frac{1}{2}z^2$ *and* $\mathbf{B} = 1$. *Since* $\mathbf{B}^\top\mathbf{B} = 1 > 0$ *and;*

$$\max\{|a(t)z + K_p z|, |z|\} \leq \rho|z| \tag{A.51}$$

with $\rho = a_{\max} + K_p$, *the equilibrium point* $z = 0$ *is UGAS according to Theorem A.5. Notice that the LaSalle-Yoshizawa Theorem fails for this case since:*

$$V(z,t) = W(z) + \frac{1}{2p}\tilde{\theta}^2 \tag{A.52}$$

$$\begin{aligned} \dot{V}(z,t) &= z\dot{z} + \frac{1}{p}\tilde{\theta}\dot{\tilde{\theta}} \\ &= -[a(t) + K_p]z^2 \\ &\leq 0 \end{aligned} \tag{A.53}$$

which by LaSalle-Yoshizawa only shows UGS and $z(t) \to 0$, *but not* $\tilde{\theta} \to 0$.

Appendix B

Numerical Methods

From a physical point of view, vessel dynamics and kinematics are most naturally derived in the continuous-time domain using *Newtonian* or *Lagrangian* dynamics. In the implementation of a control law, it is desirable to represent the nonlinear dynamics and kinematics in discrete time. This chapter discusses methods for discretization of linear and nonlinear systems, numerical integration and differentiation.

B.1 Discretization of Continuous-Time Systems

This section discusses discretization of linear state-space models with extensions to nonlinear systems using the method of Smith (1977).

Forward Shift Operator

For notational simplicity, let $t_k = kt$ such that $\mathbf{x}(k) = \mathbf{x}(t_k)$ and $\mathbf{x}(k+1) = \mathbf{x}(t_k + h)$ where h is the sampling interval. The *forward shift operator* z is defined by:

$$\mathbf{x}(k+1) := z\mathbf{x}(k) \tag{B.1}$$

B.1.1 Linear State-Space Models

Consider the linear continuous-time model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{B.2}$$

Assume that $\mathbf{u}$ is piecewise constant over the sampling interval h and equal to $\mathbf{u}(k)$. Hence, the solution of (B.2) can be written:

$$\mathbf{x}(k+1) = \exp(\mathbf{A}h)\mathbf{x}(k) + \int_{kh}^{(k+1)h} \exp(\mathbf{A}[(k+1)h - \tau])\mathbf{B}u(k)\, d\tau \tag{B.3}$$

which after integration yields the linear discrete-time model:

$$\mathbf{x}(k+1) = \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Delta}\mathbf{u}(k) \tag{B.4}$$

where

$$\mathbf{\Phi} = \exp(\mathbf{A}h), \quad \mathbf{\Delta} = \mathbf{A}^{-1}(\mathbf{\Phi} - \mathbf{I})\mathbf{B} \tag{B.5}$$

Matlab:
The matrices $\mathbf{\Phi}$ and $\mathbf{\Delta}$ can be computed in Matlab™ as:

```
[PHI, DELTA] = c2d(A, B, h)
```

Example B.1 (Discretization of a 1st-Order Linear System)
Consider the SISO linear system:

$$\dot{x} = ax + bu \tag{B.6}$$
$$y = cx + du \tag{B.7}$$

Application of (B.3), yields:

$$x(k+1) = \exp(ah)x(k) + \frac{b}{a}(\exp(ah) - 1)u(k) \tag{B.8}$$
$$y(k) = cx(k) + du(k) \tag{B.9}$$

Computation of the Transition Matrix

The transition matrix $\mathbf{\Phi}$ can be computed numerically as:

$$\mathbf{\Phi} = \exp(\mathbf{A}h) = \mathbf{I} + \mathbf{A}h + \frac{1}{2!}\mathbf{A}^2h^2 + ... + \frac{1}{n!}\mathbf{A}^nh^n + ... \tag{B.10}$$

Hence:

$$\mathbf{\Delta} = \mathbf{A}^{-1}(\mathbf{\Phi} - \mathbf{I})\mathbf{B} = h + \frac{1}{2!}\mathbf{A}h^2 + ... + \frac{1}{n!}\mathbf{A}^{n-1}h^n + .. \tag{B.11}$$

Consequently, a 1st-order approximation (Euler discretization) is obtained by:

$$\mathbf{\Phi} \approx \mathbf{I} + \mathbf{A}h, \quad \mathbf{\Delta} \approx \mathbf{B}h \tag{B.12}$$

Alternately, $\mathbf{\Phi}$ can be computed by applying a *similarity transformation*:

$$\mathbf{\Phi} = \exp(\mathbf{A}h) = \mathbf{E}\exp(\mathbf{\Lambda}h)\mathbf{E}^{-1} \tag{B.13}$$

where

$$\exp(\mathbf{\Lambda}h) = \text{diag}\{\exp(\lambda_i h)\} \tag{B.14}$$

is a diagonal matrix containing the eigenvalues λ_i of $\mathbf{A}$ and $\mathbf{E}$ is the corresponding eigenvector matrix.

Matlab:
The transition matrix can be computed in Matlab™ as:

```
[L,E]=eig(A)
PHI=E*exp(L*h)*inv(E)
```

B.1.2 Nonlinear State-Space Models

Consider the nonlinear model:

$$\mathbf{M}\,\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\,\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\,\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = \mathbf{B}\mathbf{u} \tag{B.15}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{B.16}$$

which can be expressed as a nonlinear time-invariant system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{B.17}$$

where $\mathbf{x} = [\boldsymbol{\eta}^\top, \boldsymbol{\nu}^\top]^\top$ and:

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \\ \mathbf{M}^{-1}[\mathbf{B}\mathbf{u} - \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{g}(\boldsymbol{\eta})] \end{bmatrix} \tag{B.18}$$

Differentiating (B.17) with respect to time, yields:

$$\ddot{\mathbf{x}} = \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\dot{\mathbf{x}} + \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}}\dot{\mathbf{u}} \tag{B.19}$$

The effect of a zero-order-hold in the digital-to-analog converter makes $\dot{\mathbf{u}} = \mathbf{0}$ over the discrete-time interval. Furthermore, the definition of the Jacobian:

$$\mathcal{J}(\mathbf{x}) = \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \tag{B.20}$$

implies that the nonlinear continuous equation (B.19) is reduced to a homogeneous equation:

$$\ddot{\mathbf{x}} = \mathcal{J}(\mathbf{x})\dot{\mathbf{x}} \tag{B.21}$$

Let:

$$\mathcal{J}(\mathbf{x}(k)) = \left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}(k)} \tag{B.22}$$

Hence, the solution of the homogeneous differential equation is:

$$\dot{\mathbf{x}} = \exp(\mathcal{J}(\mathbf{x}(0))(t - t_0))\, \dot{\mathbf{x}}(0) \tag{B.23}$$

Integration of this expression over a sampling interval h, finally yields:

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \int_0^h \exp(\mathcal{J}(\mathbf{x}(k))\tau)\, \dot{\mathbf{x}}(k)\, d\tau \tag{B.24}$$

Example B.2 (Discretization of a 2nd-Order Nonlinear System)
Consider the SISO nonlinear system:

$$\dot{x}_1 = x_2 \tag{B.25}$$
$$\dot{x}_2 = f(x_2) + u \tag{B.26}$$

where $\mathbf{x} = [x_1, x_2]^\top$ *is the state vector and* u *is the input. The Jacobian is found as:*

$$\mathcal{J}(\mathbf{x}) = \begin{bmatrix} 0 & 1 \\ 0 & a(x_2) \end{bmatrix}; \qquad a(x_2) = \frac{\partial f(x_2)}{\partial x_2} \tag{B.27}$$

Hence, applying a similarity transformation:

$$\exp(\mathcal{J}(\mathbf{x}(k))t) = \mathbf{E}^{-1} \exp(\mathbf{\Lambda} t)\mathbf{E} \tag{B.28}$$

where $\mathbf{\Lambda}$ *is a diagonal matrix containing the eigenvectors of* $\mathcal{J}$ *and* $\mathbf{E}$ *is a matrix formed by the corresponding eigenvectors, yields:*

$$\exp(\mathbf{J}(\mathbf{x}(k))t) = \begin{bmatrix} 1 & \frac{1}{a_k}(1 - \exp(a_k t)) \\ 0 & \exp(a_k t) \end{bmatrix} \tag{B.29}$$

where $a_k = a(x_2(k))$. *Hence,*

$$\begin{bmatrix} \mathbf{x}_1(k+1) \\ \mathbf{x}_2(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1(k) \\ \mathbf{x}_2(k) \end{bmatrix} + \int_0^h \begin{bmatrix} 1 & \frac{1}{a_k}(1 - \exp(a_k\tau)) \\ 0 & \exp(a_k\tau) \end{bmatrix} \begin{bmatrix} x_2(k) \\ f(x_2(k)) + u(k) \end{bmatrix} d\tau \tag{B.30}$$

The discrete model (B.24) can be simplified by approximating the exponential function to the first order, that is:

$$\exp(\mathcal{J}(\mathbf{x}(k))h) = \mathbf{I} + \mathcal{J}(\mathbf{x}(k))h + O(h^2) \tag{B.31}$$

B.2 Numerical Integration Methods

In this section numerical solutions to the nonlinear time-varying system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{B.32}$$

where the control input $\mathbf{u}$ is assumed to be constant over the sampling interval h (zero-order hold) are discussed. Four different methods will be presented.

B.2.1 Euler's Method

Euler proposed the algorithm:

$$\mathbf{x}(k+1) = \mathbf{x}(k) + h\mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), t_k) \tag{B.33}$$

The global truncation error for Euler's Method is of order $O(h)$.

Applying Euler's method to a 2nd-order system:

$$\dot{x} = v \tag{B.34}$$

$$m\dot{v} + dv + kx = \tau \tag{B.35}$$

yields:

$$v(k+1) = v(k) + h\left[\frac{1}{m}\tau(k) - \frac{d}{m}v(k) - \frac{k}{m}x(k)\right] \tag{B.36}$$

$$x(k+1) = x(k) + hv(k) \tag{B.37}$$

It should be noted that Euler's method should only be applied to a well-damped 2nd-order system and not an undamped oscillator. In fact an undamped oscillator will yield an unstable solution as seen from Figure B.1 where the circle in the upper left plot represents the stable region. An undamped oscillator will have eigenvalues on the imaginary axis, which clearly lie outside of the circle.

Forward and Backward Euler Integration

A stable method for the undamped 2nd-order system can be obtained by combining the *forward* and *backward* methods of Euler (dotted line in the upper left plot in Figure B.1). This suggests:

$$\text{Forward Euler} \quad v(k+1) = v(k) + h\left[\frac{1}{m}\tau(k) - \frac{d}{m}v(k) - \frac{k}{m}x(k)\right] \tag{B.38}$$

$$\text{Backward Euler} \quad x(k+1) = x(k) + hv(k+1) \tag{B.39}$$

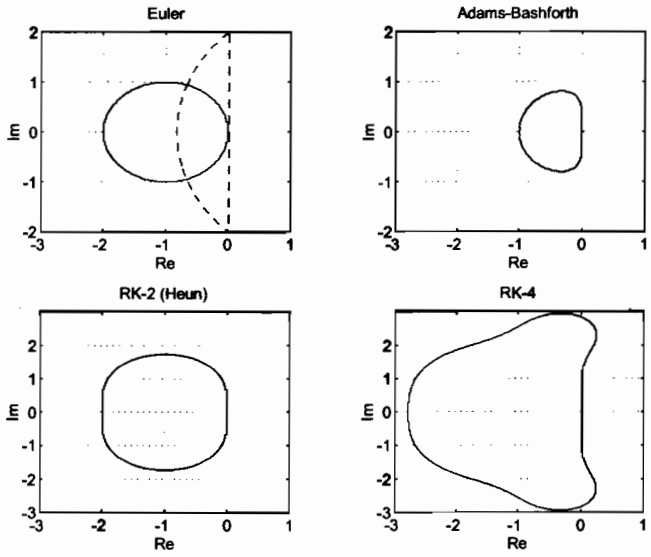


Figure B.1: Stability regions for the Euler, Adams-Bashford, RK-2 and RK-4 methods.

Extension to Nonlinear Systems

The methods of Euler can be extended to the more general nonlinear system:

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}\left[\mathbf{B}\mathbf{u} - \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} - \mathbf{g}(\boldsymbol{\eta})\right] \tag{B.40}$$

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{B.41}$$

by the following set of discrete-time equations:

$$\boldsymbol{\nu}(k+1) = \boldsymbol{\nu}(k) + h\mathbf{M}^{-1} \left[\mathbf{B}\mathbf{u}(k) - \mathbf{C}(\boldsymbol{\nu}(k))\boldsymbol{\nu}(k) - \mathbf{D}(\boldsymbol{\nu}(k))\boldsymbol{\nu}(k) - \mathbf{g}(\boldsymbol{\eta}(k))\right] \tag{B.42}$$

$$\boldsymbol{\eta}(k+1) = \boldsymbol{\eta}(k) + h\left[\mathbf{J}(\boldsymbol{\eta}(k))\boldsymbol{\nu}(k+1)\right] \tag{B.43}$$

B.2.2 Adams–Bashforth's 2nd-Order Method

Adams–Bashforth integration is more computationally intensive than the schemes of Euler. For instance, the two-step Adams–Bashforth integration:

$$\mathbf{x}(k+1) = \mathbf{x}(k) + h\left[\frac{3}{2}\mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), t_k) - \frac{1}{2}\mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k-1), t_{k-1})\right] \tag{B.44}$$

implies that the old value:

$$\dot{\mathbf{x}}(k-1) = \mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k-1), t_{k-1}) \tag{B.45}$$

must be stored. The global truncation error for this method is of order $O(h^2)$. The advantage with this method compared to Euler integration is seen from Figure B.1.

B.2.3 Runge–Kutta 2nd-Order Method (Heun's Method)

Heun's integration method can be written:

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), t_k) \\ \mathbf{k}_2 &= \mathbf{f}\left(\mathbf{x}(k) + h\mathbf{k}_1, \mathbf{u}(k), t_k + h\right) \\ \mathbf{x}(k+1) &= \mathbf{x}(k) + \tfrac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2) \end{aligned} \tag{B.46}$$

The global truncation error for Heun's Method is of order $O(h^2)$.

B.2.4 Runge–Kutta 4th-Order Method

An extension of Heun's integration method to 4th-order is:

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), t_k) \\ \mathbf{k}_2 &= h\mathbf{f}(\mathbf{x}(k) + \mathbf{k}_1/2, \mathbf{u}(k), t_k + h/2) \\ \mathbf{k}_3 &= h\mathbf{f}(\mathbf{x}(k) + \mathbf{k}_2/2, \mathbf{u}(k), t_k + h/2) \\ \mathbf{k}_4 &= h\mathbf{f}(\mathbf{x}(k) + \mathbf{k}_3/2, \mathbf{u}(k), t_k + h) \\ \mathbf{x}(k+1) &= \mathbf{x}(k) + \tfrac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \end{aligned} \tag{B.47}$$

The global truncation error for the RK-4 Method is of order $O(h^4)$.

B.3 Numerical Differentiation

Numerical differentiation is usually sensitive to noisy measurements. Nevertheless, a reasonable estimate $\dot{\eta}_f$ of the time derivative $\dot{\eta}$ of a signal η can be obtained by using a *filtered differentiation*. The simplest filter is obtained by the 1st-order low-pass structure:

$$\dot{\eta}_f(s) = \frac{Ts}{1+Ts}\,\eta(s) \tag{B.48}$$

corresponding to the continuous-time system:

$$\dot{x} = ax + bu \tag{B.49}$$

$$y = cx + du \tag{B.50}$$

with $u = \eta$, $y = \dot{\eta}_f$, $a = b = -1/T$ and $c = d = 1$. Using the results from Example B.1, the following discrete-time filter equations are obtained:

$$x(k+1) = \exp(-h/T)x(k) + (\exp(-h/T) - 1)u(k) \quad \text{(B.51)}$$

$$y(k) = x(k) + u(k) \quad \text{(B.52)}$$

Appendix C

Matlab GNC Toolbox

The examples and computer simulations in the book is based on the Matlab™ GNC Toolbox by Fossen (2001). In order to use the Matlab™ GNC Toolbox you need *Matlab 3.1, 6.0, 6.1* or *6.5* from:

MATHWORKS INC
http://www.mathworks.com/

The toolbox is free of charge and can be downloaded from:

MARINE CYBERNETICS
http://www.marinecybernetics.com/

In order to use the Matlab™ GNC toolbox, several directories under *.../toolbox/gnc/* must be added to your Matlab path. This is done automatically by typing:

```
>>gnc
```

each time you use the toolbox. Alternatively you can add the path permanently to the Matlab search list. The needed paths for a permanent configuration are found by opening the file *gnc.m* in an editor.

How to use the Matlab GNC Toolbox

A list of the GNC toolbox commands are obtained by typing:

```
>>help gnc
```

where more information on how to use the toolbox is given. You can also run several demo files by typing:

```
>>gncdemo
```

C.1 M-File Library

The toolbox directories are organized according to:

General Purpose Commands: *.../toolbox/gnc* contains general purpose m-file functions for kinematic computations, dynamics, modeling, navigation, control, simulation etc.

Vessel Models: *.../toolbox/gnc/vesselmodels/* contains stand-alone vessel *m-files* for feedback control design and computer simulation. These models are based on articles published in literature (see references in the help text of the files), and industrial models provided by several companies, international researchers and the author of this book. A list of the different models are presented by typing:

```
>>help vesselmodels
```

The format of the models is:

```
[xdot,U]=model(x,ui,...)
```

where `xdot` is the time derivative of the state vector `x`, `U` is the speed of the vessel (optionally), and `ui` is the input vector. Type

`>>help model` where `model` $\in$ {`mariner,tanker,container,npsauv,etc.`}

to see the definitions of the input and output arguments. In order to simulate the models you must include a numerical integration routine and store the time-series in an array. Type:

```
>>simdemo
```

to see how this can be done. The scripts `simdemo1.m, simdemo2.m,...` are user editable scripts for this purpose. Copy the files to your work directory and perform the necessary modifications needed for your case study.

Book Examples: *.../toolbox/gnc/examples* contains the example files used in this book. The examples files make use of the GNC Toolbox commands and demonstrates how several m-functions can be called. Type:

```
>>help gnc/examples
```

to see a list of the different example files.

The GNC toolbox is updated on a frequent basis as indicated by the version number. This includes new *m-file* functionality, bug fixes and other modifications. Please use:

http://www.marinecybernetics.com/

to download the latest version of the toolbox. Bugs and other problems can be reported to:

bugs@marinecybernetics.com

If you want to contribute new m-files or vessel models to be included in the toolbox, please send an e-mail, including the files and a short description to:

info@marinecybernetics.com

C.2 Simulink Library

The GNC Toolbox includes a Simulink library for simulation of marine vessels and control systems. The library includes blocks for:

- Guidance (reference models, path planners and guidance systems)
- Navigation (wave filters, sensor systems, GPS and IMU integration filters)
- Control (autopilots, DP control systems and tracking control systems)
- Library blocks (kinematics, equations of motion, transformations etc.)
- Models (tankers, cargo ship, rigs, underwater vehicles, propellers, rudders, wind, waves and currents)
- Examples (closed loop control systems)

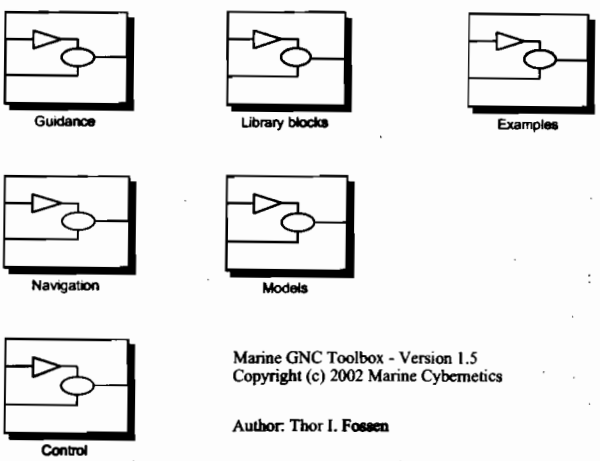


Figure C.1: The GNC Toolbox Simulink Library.

Matlab:
The GNC Simulink library is loaded from Matlab™ by the following command:

```
marineGNC
```

or by clicking on the Simulink library browser.

Bibliography

12th ITTC (1969). Report of the Seakeeping Committee. In: *Proceedings of the 12th International Towing Tank Conference.* Rome, September 1969. pp. 775–779.

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

15th ITTC (1978). Report of the Seakeeping Committee. In: *Proceedings of the 15th International Towing Tank Conference.* The Hague, September 1978. pp. 55–70.

17th ITTC (1984). Report of the Seakeeping Committee. In: *Proceedings of the 17th International Towing Tank Conference.* The Hague, September 1984. pp. 531–534.

2nd ISSC (1964). Report of the Seakeeping Committee. In: *Proceedings of the 2nd International Ship and Offshore Structures Congress.* Delft, August 1964. p. 19.

Aamo, O. M., M. Arcak, T. I. Fossen and P. V. Kokotovic (2000). Global Output Tracking Control of a Class of Euler-Lagrange Systems with Monotonic Nonlinearities in the Velocities. *International Journal of Nonlinear Control* **IJC-74**(7), 649–658.

Aarset, M. F., J. P. Strand and T. I. Fossen (1998). Nonlinear Vectorial Observer Backstepping With Integral Action and Wave Filtering for Ships. In: *Proc. of the IFAC Conf. on Control Applications in Marine Systems (CAMS'98).* pp. 83–89.

Abkowitz, M. A. (1964). Lectures on Ship Hydrodynamics - Steering and Maneuverability. Technical Report Hy-5. Hydro- and Aerodynamic's Laboratory. Lyngby, Denmark.

Allensworth, T. (1999). A Short History of Sperry Marine. Internet. <http://www.sperry-marine.com/ pages/history.html>[Accessed October 1, 2002].

Allmendinger, E. E., Ed.) (1990). *Submersible Vehicle Systems Design.* The Society of Naval Architects and Marine Engineers. 601 Pavonia Avenue, Jersey City, N. J. 07306.

Antonelli, G., S. Chiaverini, N. Sarkar and M. West (2001). Adaptive Control of an Autonomous Underwater Vehicle: Experimental Results on ODIN. *IEEE Transactions on Control Systems Technology* **TCST-9**(5), 756–765.

Aranda, J., J. M. de-la-Cruz, J. M. Diaz, B. de-Andres, P. Ruiperez, S. Esteban and J. M. Giron (2000). Modelling of a High Speed Craft by a Nonlinear Least Squares Method witb Constraints. In: *Proceedings of the 5th IFAC Conference on Maneuvering and Control of Marine Craft (MCMC'2000).* Aalborg, Denmark. pp. 227–232.

Arcak, M. and P. V. Kokotovic (1999*a*). Nonlinear Observers: A Circle Criterion Design. In: *Proceedings of the IEEE Conference on Decision and Control*. Phoenix, AZ. pp. 4872–4876.

Arcak, M. and P. V. Kokotovic (1999*b*). Observer Based Stabilization of Systems with Monotonic Nonlinearities. *Asian Journal of Control* **AJC-1**, 42–48.

Arimoto, S. and F. Miyazaki (1984). Stability and Robustness of PID Feedback Control for Robot Manipulators of Sensory Capability. In: *Proc. of the 1st Int. Symp. On Robotics Research* (M. Brady and R. Paul, Eds.). pp. 783–799. MIT Press.

Arstein, Z. (1983). Stabilization with Relaxed Controls. *Nonlinear Analysis* **TMA-7**, 1163–1173.

Asare, H. and D. Wilson (1986). Design of Computed Torque Model Rcference Adaptive Control for Space-Based Robotic Manipulators. ASME, Winter Annual Meeting (WAM), pp. 195–204.

Athans, M. and Falb, P. L., Eds.) (1966). *Optimal Control*. McGraw-Hill Book Company. New York.

Baitis, A. E. (1980). The Development and Evaluation of a Rudder Roll Stabilization System for the WHEC Hamiltonian Class. Technical Report DTNSRDC. Naval Ship Research and Development Center. Bethesda, Md.

Baitis, E., D. A. Woolaver and T. A. Beck (1983). Rudder Roll Stabilization for Coast Guards Cutters and Frigates. *Naval Engineers Journal*, pp. 267–282.

Baitis, E., D. A. Woolaver and T. A. Beck (1989). Ship Roll Stabilization in the U. S. Navy. *Naval Engineers Journal*, pp. 43–53.

Balchen, J. G., N. A. Jenssen and S. Sælid (1976). Dynamic Positioning Using Kalman Filtering and Optimal Control Theory. In: *Proc. of the IFAC/IFIP Symp. On Automation in Offshore Oil Field Operation*. Bergen, Norway. pp. 183–186.

Balchen, J. G., N. A. Jenssen and S. Sælid (1980*a*). Dynamic Positioning of Floating Vessels Based on Kalman Filtering and Optimal Control.. In: *Proceedings of the 19th IEEE Conference on Decision and Control*. New York, NY. pp. 852–864.

Balchen, J. G., N. A. Jenssen, E. Mathisen and S. Sælid (1980*b*). Dynamic Positioning System Based on Kalman Filtering and Optimal Control. *Modeling, Identification and Control* **MIC**-1(3), 135–163.

Barbier, C., P. Sen and M. Downie (1994). Parallel Dynamic Programming and Ship Voyage Management. *Concurrency: Practice and Experience* **CPE-6**(8), 673–696.

Barbour, N. and G. Schmidt (1998). Inertial Sensor Technology Trends. In: *Proc. Of the Workshop on Autonomous Underwater Vehicles*. pp. 52–55.

Barbălat (1959). Systèmes d'Équations Différentielles d'Oscillations Non Linéaires. *Revue de Mathématiques Pures et Appliquées* **Vol. 4**(2), 267–270. Académie de la République Populaire Roumaine (in Freneh).

Batlle, J., A. Casals, J. Freixenet and J. Martí (2000). A Review on Strategies for Recognizing Natural Objects in Colour Images of Outdoor Scenes. *Image and Vision Computing* **IVC-18**(6-7), 515–530.

Bech, M. I. (1968). The Reversed Spiral Test as Applied to Large Ships. In: *Shipping World and Shipbuilder*. pp. 1753–1754.

Bech, M. I. and L. Wagner Smith (1969). Analogue Simulation of Ship Maneuvers. Technical Report Hy-14. Hydro- and Aerodynamics Laboratory. Lyngby, Denmark.

Bennet, S. (1979). *A History of Control Engineering 1800-1930*. Peter Peregrinus. London.

Bennett, S. (1991). Ship Stabilization: History. In: *Concise Encyclopedia of Traffic and Transportation Systems* (Markos Papageorgiou, Ed.). pp. 454–459. Pergamon Press.

Berge, S. P. and T. I. Fossen (1997). Robust Control Allocation of Overactuated Ships: Experiments With a Model Ship. In: *Proc. of the 4th IFAC Conference on Manoeuvring and Control of Marine Craft*. Brijuni, Croatia. pp. 166–171.

Berge, S. P. and T. I. Fossen (2000). On the Properties of the Nonlinear Ship Equations of Motion. *Journal of Mathematical and Computer Modelling of Dynamical Systems* **JMCMDS-6**(4), 365–381.

Berge, S. P., K. Ohtsu and T. I. Fossen (1999). Nonlinear Control of Ships Minimizing the Position Tracking Errors. *Modeling, Identification and Control* **MIC-20**(3), 177–187.

Berghuis, H. (1993). Model-Based Robot Control: From Theory to Practice. PhD thesis. University of Twente, Enschede, The Netherlands.

Bhattacharyya, R. (1978). *Dynamics of Marine Vehicles*. John Wiley & Sons Ltd.. New York, NY.

Blanke, M. (1981). Ship Propulsion Losses Related to Automated Steering and Prime Mover Control. PhD thesis. The Technical University of Denmark, Lyngby.

Blanke, M. (1996). Uncertainty Models for Rudder-Roll Damping Control. In: *Proc. of the IFAC World Congress*. Vol. Q. Elesevier. pp. 285–290. San Francisco.

Blanke, M., Adrian, K-E. Larsen and J. Bentsen (2000*a*). Rudder Roll Damping in Coastal Region Sea Conditions. In: *Proc. 5th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC'00)*. pp. 39–44. Aalborg, Denmark.

Blanke, M. and A. Christensen (1993). Rudder-Roll Damping Autopilot Robustness due to Sway-Yaw-Roll Couplings. In: *Proc. of the 10th Int. Ship Control Systems Symposium (SCSS'93)*. Ottawa, Canada. pp. A.93–A.119.

Blanke, M. and A. G. Jensen (1997). Dynamic Properties of Container Vessel with Low Metacentric Height. *Transactions of the Institute of Measurement and Control* **TIMC-19**(2), 78–93.

Blanke, M. and A. Tiano (1997). Multivariable Identification of Ship Steering and Rol Motions. *Transactions of the Institute of Measurement and Control* **TIMC-19(2), 62–77.**

Blanke, M., K. P. Lindegaard and T. I. Fossen (2000*b*). Dynamic Model for Thrust Generation of Marine Propellers. In: *Proceedings of the IFAC Conference of Manoeuvreing of Marine Craft (MCMC'00)*. Aalborg, Denmark.

Blanke, M., P. Haals and K. K. Andreasen (1989). Rudder Roll Damping Experience in Denmark. In: *Proc. of the IFAC Workshop on Expert Systems and Signal Processing in Marine Automation*. Lyngby, Denmark. pp. 149–160.

Blanke, M., R. Izadi-Zamanabadi and T. F. Lootsma (1998). Fault Monitoring and Reconfigurable Control for a Ship Propulsion Plant. *Journal of Adaptive Control and Signal Processing* **JACSP-12**, 671–688.

Blendermann, W. (1986). Die Windkräfte am Schiff. Technical Report Bericht Nr. 467. Institut für Schiffbau der Universität Hamburg. in German.

Bordignon, K. A. and W. C. Durham (1995). Closed-Form Solutions to Constrained Control Allocation Problem. *Journal of Guidance, Control and Dynamics* **JGCD–18**(5), 1000–1007.

Breslin, J. P. and P. Andersen (1994). *Hydrodynamics of Ship Propellers*. Cambridge University Press. U.K.

Bretschneider, C. L. (1959). Wave Variability and Wave Spectra for Wind Generated Gravity Waves. Technical report. Beach Erosion Board, Corps. of Engineers. 118 (Technical Memo).

Bretschneider, C. L. (1969). *Wave and Wind Loads. Section 12 of Handbook of Ocean and Underwater Engineering*. McGraw-Hill. New York, NY.

Brian, D., O. Andersen and J. B. Moore (1989). *Optimal Control: Linear Quadratic Methods*. Prentice Hall. London.

Britting, K. R. (1971). *Inertial Navigation Systems Analysis*. Wiley Interscience.

Brown, R. G. and Y. C. Hwang (1998). *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley and Sons Ltd. New York.

Bullo, F., N. E. Leonard and A. D. Lewis (1999). Controllability and Motion Algorithms for Underactuated Lagrangian Systems on Lie Groups. *IEEE Transactions on Automatic Control*.

Burger, W. and A. G. Corbet (1960). *Ship Stabilizers. Their Design and Operation in Correcting the Rolling of Ships. A Handbook for Merchant Navy Officers*. Pergamon Press Ltd. London.

Byrnes, C. I., A. Isidori and J. C. Willems (1991). Passivity, Feedback Equivalence, and the Global Stabilization of Minimum Phases Nonlinear Systems. *IEEE Transactions on Automatic Control* **TAC-36**, 1228–1240.

Byrnes, C. I. and A. Isidori (1989). New Results and Examples in Nonlinear Feedback Stabilization. *Systems and Control Letters* **SCL-12**, 437–442.

Caccia, M. and G. Veruggio (2000). Guidance and Control of a Reconfigurable Unmanned Underwater Vehicle. *Control Engineering Practice* **CEP-8**(1), 21–37.

Calvert, S. (1989). Optimal Weather Routing Procedures for Vessels on Oceanic Voyages. PhD thesis. Institute of Marine Studies, Polytechnic South West, U.K.

Canudas de Wit, C., O. Olguin Diaz and M. Perrier (2000). Nonlinear Control of an Underwater Vehicle/Manipulator with Composite Dynamics. *IEEE Transactions on Control Systems Technology* **TCST-8**(6), 948–960.

Carley, J. B. (1975). Feasibility Study of Steering and Stabilizing by Rudder. In: *Proceedings of the 4th International Ship Control Systems Symposium (SCSS'75)*. The Hague, The Netherlands.

Carlton, J. S. (1994). *Marine Propellers and Propulsion*. Oxford: Butterworth-Heinemann.

Chislett, M. S. and J. Strøm-Tejsen (1965*a*). Planar Motion Mechanism Tests and Full-Scale Steering and Maneuvering Predictions for a Mariner Class Vessel. Technical Report Hy-5. Hydro- and Aerodynamics Laboratory. Lyngby, Denmark.

Chislett, M. S. and J. Strøm-Tejsen (1965*b*). Planar Motion Mechanism Tests and Full-Scale Steering and Maneuvering Predictions for a Mariner Class Vessel. Technical Report Hy-6. Hydro- and Aerodynamics Laboratory. Lyngby, Denmark.

Chou, J. C. K. (1992). Quaternion Kinematic and Dynamic Differential Equations. *IEEE Transactions on Robotics and Automation* **RA-8**(1), 53–64.

Christensen, A. and M. Blanke (1986). A Linearized State-Space Model in Steering and Roll of a High-Speed Container Ship. Technical Report 86-D-574. Servolaboratoriet, Technical University of Denmark. Denmark.

Cody, S. E. (1992). An experimental study of the response of small thrusters to step and triangular wave inputs. Master's thesis. Naval Postgraduate School, Monterey, CA.

Conte, G. and A. Serrani (1998). Robust Control of a Remotely Operated Underwater Vehicle. *Automatica* **AUT-34**(2), 193–198.

Coron, J.-M. (1995). On the Stabilization in Finite Time of Locally Controllable Systems by Means of Continuous Time-Varying Feedback Law. *SIAM Journal of Control and Optimization* **JCO-33**(3), 804–833.

Cowley, W. E. and T. H. Lambert (1972). The Use of Rudder as a Roll Stabilizer. In: *Proceedings of the 3rd International Ship Control Systems Symposium (SCSS'72)*. Bath, UK.

Cowley, W. E. and T. H. Lambert (1975). Sea Trials on a Roll Stabilizer Using the Ship's Rudder. In: *Proceedings of the 4th International Ship Control Systems Symposium (SCSS'75)*. The Hague, The Netherlands.

Craig, J. J. (1989). *Introduction to Robotics*. Addison-Wesley. Reading, Massachusetts.

Cristi, R., F. A. Papoulias and A. J. Healey (1990). Adaptive Sliding Mode Control of Autonomous Underwater Vehicles in the Dive Plane . *IEEE Journal of Oceanic Engineering* **OE-15**(3), 152–160.

Davidson, K. S. M. and L. I. Schiff (1946). Turning and Course Keeping Qualities. *Transactions of SNAME*.

De Kat, J. O. and J. E. W. Wichers (1991). Behavior of a Moored Ship in Unsteady Current, Wind and Waves. *Marine Technology* **28**(5), 251–264.

de-la-Cruz, J. M., J. Aranda, P. Ruiperez, J. M. Diaz and A. Maron (1998). Identification of the Vertical Plane Motion Model of a High Speed Craft by Model Testing in Irregular Waves. In: *Proceedings of the IFAC Conference on Control Applications in Marine Systems (CAMS'98)*. Fukuoaka, Japan. pp. 257–262.

Defant, A. (1961). *Physical Oceanography*. Pergamon Press. London.

Dieudonné, J. (1953). Collected French Papers on the Stability of Route of Ships at Sea, 1949-1950. (Translated by H. E. Saunders and E. N. Labouvie). Technical Report DTMB-246. Naval Ship Research and Development Center. Washington D.C.

DnV (1990). *Rules for Classification of Steel Ships: Dynamic Positioning Systems, Part 6, Chapter 7*. Det norske Vertitas, Veritasveien 1, N-1322 Høvik, Norway.

Do, K. D., Z. P. Jiang and J. Pan (2002*a*). Robust Global Stabilization of Underactuated Ships on a Linear Course: State and Output Feedback. *International Journal of Control*. to appear.

Do, K. D., Z. P. Jiang and J. Pan (2002*b*). Underactuated Ship Global Tracking under Relaxed Conditions. *IEEE Transactions on Automatic Control* **TAC-47**(9), 1529–1535.

Do, K. D., Z. P. Jiang and J. Pan (2002*c*). Universal Controllers for Stabilization and Tracking of Underactuated Ships. *Systems and Control Letters* **SCL-47**(4), 299–317. To appear.

Donha, D. C., D. S. Desanj, M. R. Katebi and M. J. Grimble (1998). H_∞ adaptive Controller for Autopilot Applications. *Spesial Issue on Marine Systems of the International Journal of Adaptive Control and Signal Processing* **IJACSP-12**(8), 623–648.

Duc-Hung, N. and K. Ohtsu (2000). An Adaptive Optimal Autopilot using the rRcursive Prediction Error Method. In: *Proc. of the IFAC Conference on Maneuvering and Control of Marine Craft (MCMC'00)*. Aalborg, Denmark. pp. 191–196.

Duc-Hung, N., L. Minh-Duc and K. Ohtsu (2000). Ship's Optimal Autopilot with a Multivariate Auto-Regressive Exogenous Model. In: *Proc. of the Control Applications of Optimization (CAO'00)*. Kidlington, UK. pp. 277–282.

Durham, W. C. (1993). Constrained Control Allocation. *Journal of Guidance, Control and Dynamics* **JGCD-16**(4), 717–725.

Durham, W. C. (1994*a*). Attainable Moments for the Constrained Control Allocation Problem. *Journal of Guidance, Control and Dynamics* **JGCD–17**(6), 1371–1373.

Durham, W. C. (1994*b*). Constrained Control Allocation: Three Moment Problem. *Journal of Guidance, Control and Dynamics* **JGCD-17**(2), 330–336.

Durham, W. C. (1999). Efficient, Near-Optimal Control Allocation. *Journal of Guidance, Control and Dynamics* **JGCD–22**(2), 369–372.

Dyne, G. and P. Trägårdh (1975). Simuleringsmodell för 350000 tdw tanker i fullast- och ballastkonditioner på djupt vatten. Technical Report Report 2075-1. Swedish State Shipbuilding Experimental Tank (SSPA). Gothenburg, Sweden.

Egeland, O. and J. T. Gravdahl (2002). *Modeling and Simulation for Automatic Control.* Marine Cybernetics. Trondheim, Norway.

Encarnacao, P. and A. Pascoal (2001*a*). Combined Trajectory Tracking and Path Following: An Application to the Coordinated Control of Autonomous Marine Craft. In: *Proc. of the IEEE Conference on Decision and Control.* Orlando, FL. pp. 964 –969.

Encarnacao, P. and A. Pascoal (2001*b*). Combined Trajectory Tracking and Path Following for Marine Craft. In: *Proc. of the 9th Mediterranean Conference on Control and Automation.* Dubbrovnik, Croatia.

Esteban, S., B. De Andres, J. M. Giron-Sierra, O. R. Polo and E. Moyano (2001). A Simulation Tool for a Fast Ferry Control Design. In: *Proc. of the IFAC Conference on Control Applications in Marine Systems (CAMS'01).* Glasgow, U.K.

Esteban, S., J. M. de-la-Cruz, J. M. Giron-Sierra, B. de-Andres, J. M. Diaz and J. Aranda (2000). Fast Ferry Vertical Accelerations Reduction with Active Flaps and T-foil. In: *Proc. of the 5th Conf. on Maneuvering and Control of Marine Craft (MCMC'2000).* Aalborg, Denmark. pp. 233–238.

Euler, L. (1776). Novi Commentarii Academiae Scientairum Imperialis Petropolitane. - **Vol. XX**, 189.

Ezal, K. (1998). Disturbance Attenuating Control of Nonlinear Systems with Local Optimality. PhD thesis. University of California Santa Barbara (UCSB).

Faltinsen, O. M. (1990). *Sea Loads on Ships and Offshore Structures.* Cambridge University Press.

Faltinsen, O. M. and B. Sortland (1987). Slow Drift Eddy Making Damping of a Ship. *Applied Ocean Research* **AOR-9**(1), 37–46.

Fantoni, I., R. Lozano, F. Mazenc and K. Y. Pettersen (2000). Stabilization of a Nonlinear Underactuated Hovercraft. *International Journal on Robust and Nonlinear Control.* To appear.

Farrell, J. A. and M. Barth (1998). *The Global Positioning System and Inertial Navigation.* McGraw-Hill. New York.

Feldman, J. (1979). DTMSRDC Revised Standard Submarine Equations of Motion. Technical Report DTNSRDC-SPD-0393-09. Naval Ship Research and Development Center. Washington D.C.

Fjellstad, O. E. and T. I. Fossen (1994*a*). Position and Attitude Tracking of AUVs: A Quaternion Feedback Approach. *IEEE Journal of Oceanic Engineering* **OE-19**(4), 857–862.

Fjellstad, O. E. and T. I. Fossen (1994*b*). Quaternion Feedback Regulation of Underwater Vehicles. In: *Proc. of the 3rd IEEE Conference on Control Applications (CCA'94)*. Glasgow. pp. 857–862.

Fjellstad, O. E. and T. I. Fossen (1994*c*). Singularity-Free Tracking of Unmanned Underwater Vehicles in 6 DOF. In: *Proc. of the 33rd IEEE Conference on Decision and Control (CDC'94)*. Orlando, Florida. pp. 1128–1133.

Fjellstad, O. E., T. I. Fossen and O. Egeland (1992). Adaptive Control of ROVs with Actuator Dynamics and Saturation. In: *Proceedings of the 2nd International Offshore and Polar Engineering Conference (ISOPE)*. San Francisco, CA.

Forssell, B. (1991). *Radio Navigation Systems*. Prentice Hall. Englewood Cliffs, NJ.

Fossen, T. I. (1991). Nonlinear Modeling and Control of Underwater Vehicles. PhD thesis. Department of Engineering Cybernetics, Norwegian University of Science and Technology. Trondheim.

Fossen, T. I. (1993*a*). Comments on "Hamiltonian Adaptive Control of Spacecraft". *IEEE Transactions on Automatic Control* **TAC-38**(4), 671–672.

Fossen, T. I. (1993*b*). High Performance Ship Autopilot With Wave Filter. In: *Proceedings of the 10th International Ship Control Systems Symposium (SCSS'93)*. Ottawa, Canada. pp. 2.271–2.285.

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

Fossen, T. I. (2000*a*). A Survey on Nonlinear Ship Control: From Theory to Practice. In: *Proc. of the IFAC Conf. on Maneuvering and Control of Marine Craft* (G. Roberts, Ed.). Elsevier Science. The Netherlands. pp. 1–16. Plenary Talk.

Fossen, T. I. (2000*b*). Recent Developments in Ship Control Systems Design. *World Superyacht Review, Sterling Publications Limited, London* pp. 115–116.

Fossen, T. I. (2001). Matlab GNC Toolbox, Marine Cybernetics. Internet. <http://www.marinecybernetics.com>[Accessed August 1, 2002].

Fossen, T. I., and J. P. Strand (2000). Position and Velocity Observer Design. In: *The Ocean Engineering Handbook* (F. El-Hawary, Ed.). Chap. 3, pp. 189–206. CRC Press.

Fossen, T. I., A. Loria and A. Teel (2001). A Theorem for UGAS and ULES of (Passive) Nonautonomous Systems: Robust Control of Mechanical Systems and Ships. *International Journal of Robust and Nonlinear Control* **JRNC-11**, 95–108.

Fossen, T. I. and Å. Grøvlen (1998). Nonlinear Output Feedback Control of Dynamically Positioned Ships Using Vectorial Observer Backstepping. *IEEE Transactions on Control Systems Technology* **TCST-6**(1), 121–128.

Fossen, T. I. and J. P. Strand (1998). Nonlinear Ship Control (Tutorial Paper). In: *Proc. of the IFAC CAMS'98*. Fukuoaka, Japan. pp. 1–75. <http://www.itk.ntnu.no/ansatte/Fossen_Thor/book/tutorial98.pdf>[Accessed October 1, 2002].

Fossen, T. I. and J. P. Strand (1999*a*). A Tutorial on Nonlinear Backstepping: Applications to Ship Control. *Modelling, Identification and Control* **MIC-20**(2), 83–135. Tutorial workshop presented at IFAC CAMS'98.

Fossen, T. I. and J. P. Strand (1999*b*). Passive Nonlinear Observer Design for Ships Using Lyapunov Methods: Experimental Results with a Supply Vessel. *Automatica* **AUT-35**(1), 3–16.

Fossen, T. I. and J. P. Strand (2001). Nonlinear Passive Weather Optimal Positioning Control (WOPC) System for Ships and Rigs: Experimental Results. *Automatica* **AUT-37**(5), 701–715.

Fossen, T. I. and M. Blanke (2000). Nonlinear Output Feedback Control of Underwater Vehicle Propellers Using Feedback from Estimated Axial Flow Velocity. *IEEE Journal of Oceanic Engineering* **JOE-25**(2), 241–255.

Fossen, T. I. and M. Paulsen (1992). Adaptive Feedback Linearization Applied to Steering of Ships. In: *Proceedings of the 1st IEEE Conference on Control Applications*. Dayton, Ohio. pp. 1088–1093.

Fossen, T. I. and O. E. Fjellstad (1995). Nonlinear Modelling of Marine Vehicles in 6 Degrees of Freedom. *International Journal of Mathematical Modelling of Systems* **JMMS-1**(1), 17–28.

Fossen, T. I. and S. I. Sagatun (1991*a*). Adaptive Control of Nonlinear Systems: A Case Study of Underwater Robotic Systems. *Journal of Robotic Systems* **JRS-8**(3), 393–412.

Fossen, T. I. and S. I. Sagatun (1991*b*). Adaptive Control of Nonlinear Underwater Robotic Systems. In: *Proceedings of the IEEE Conference on Robotics and Automation*. Sacramento, California. pp. 1687–1695.

Fossen, T. I. and S. P. Berge (1997). Nonlinear Vectorial Backstepping Design for Global Exponential Tracking of Marine Vessels in the Presence of Actuator Dynamics. In: *Proceedings of IEEE Conf. on Decission and Control (CDC'97)*. San Diego, CA. pp. 4237–4242.

Fossen, T. I. and T. Lauvdal (1994). Nonlinear Stability Analysis of Ship Autopilots in Sway, Roll and Yaw. Proceedings of the Conference on Marine Craft Maneuvering and Control (MCMC'94), Southampton, UK.

Fossen, T. I., J. M. Godhavn, S. Berge and K. P. Lindegaard (1998). Nonlinear Control of Underactuated Ships with Forward Speed Compensation. In: *Proceedings of the IFAC Symposium on Nonlinear Control Systems (NOLCOS'98)*. Enschede, The Netherlands. pp. 121–127.

Fossen, T. I., K. P. Lindegaard and R. Skjetne (2002). Inertia Shaping Techniques for Marine Vessels using Acceleration Feedback. In: *Proceedings of the IFAC World Congress*. Elsevier Science. Barcelona.

Fossen, T. I., R. Skjetne and M. Breivik (2003). LOS Maneuvering Control of Underactuated Marine Craft. *Transactions on Control Systems Technology*. submitted.

Fossen, T. I., S. I. Sagatun and A. J. Sørensen (1996). Identification of Dynamically Positioned Ships. *Journal of Control Engineering Practice* **CEP-4**(3), 369–376.

Fotakis, J., M. J. Grimble and B. Kouvaritakis (1982). A Comparison of Characteristic Locus and Optimal Designs for Dynamic Ship Positioning Systems. *IEEE Transactions on Automatic Control* **TAC-27**(6), 1143–1157.

Freund, E. (1973). Decoupling and Pole Assignment in Nonlinear Systems. *Electronics Letter*.

Fryxell, D., P. Oliveira, A. Pascoal, C. Silvestre and I. Kaminer (1996). Navigation, Guidance and Control of AUVs: An Application to the MARIUS Vehicle. *Control Engineering Practice* **CEP-4**(3), 401–409.

Fung, P. T-K. and M. J. Grimble (1981). Self-Tuning Control of Ship Positioning Systems. In: *Self-Tuning and Adaptive Control: Theory and Applications* (C. J. Harris and S. A. Billings, Eds.). pp. 308–331. Peter Peregrinus Ltd. on behalf of IEE.

Fung, P. T-K. and M. J. Grimble (1983). Dynamic Ship Positioning Using a Self Tuning Kalman Filter. *IEEE Transactions on Automatic Control* **TAC-28**(3), 339–349.

Gelb, A., J. F. Kasper, Jr., R. A. Nash, Jr., C. F. Price and A. A. Sutherland, Jr. (1988). *Applied Optimal Estimation*. MIT Press. Boston, Massacbusetts.

Gertler, M. and G. R. Hagen (1967). Standard Equations of Motion for Submarine Simulation. Technical Report DTMB-2510. Naval Ship Research and Development Center. Washington D.C.

Gertler, M. and S. C. Hagen (1960). Handling Criteria for Surface Ships. Technical Report DTMB-1461. Naval Ship Research and Development Center. Washington D.C.

Giron-Sierra, J. M., S. Esteban, B. De Andres, J. M. Diaz and J. M. Riola (2001). Experimental Study of Controlled Flaps and T-foil for Comfort Improvements of a Fast Ferry. In: *Proceedings of the IFAC Conference on Control Applications in Marine Systems (CAMS'01)*. Glasgow, U.K.

Goldstein, H. (1980). *Classical Mechanics*. Addison-Wesley. Reading, MA.

Grenon, G., P. E. An, S. M. Smith and A. J. Healey (2001). Enhancement of the Inertial Navigation System for the Morpheus Autonomous Underwater Vehicles. *IEEE Journal of Oceanic Engineering* **JOE-26**(4), 548–560.

Grewal, M. S., L. R. Weill and A. P. Andrews (2001). *Global Positioing Systems, Inertial Navigation and Integration*. John Wiley and Sons Ltd. New York.

Grimble, M. J. (1978). Relationship Between Kalman and Notch Filters Used in Dynamic Ship Positioning Systems. *Electronics Letters* **EL-14**(13), 399–400.

Grimble, M. J. and M. A. Johnson (1989). *Optimal Control and Stochastic Estimation. Theory and Applications*. John Wiley & Sons Ltd.

Grimble, M. J., R. J. Patton and D. A. Wise (1979). The Design of Dynamic Positioning Systems using Extended Kalman Filtering Techniques. In: *Proceedings of OCEANS'79*. pp. 488–497.

Grimble, M. J., R. J. Patton and D. A. Wise (1980*a*). The Design of Dynamic Positioning Control Systems Using Stochastic Optimal Control Theory. *Optimal Control Applications and Methods* **OCAM-1**, 167–202.

Grimble, M. J., R. J. Patton and D. A. Wise (1980*b*). Use of Kalman Filtering Techniques in Dynamic Ship Positioning Systems. In: *IEE Proceedings Vol. 127, Pt. D, No. 3*. pp. 93–102.

Grøvlen, Å. and T. I. Fossen (1996). Nonlinear Control of Dynamic Positioned Ships Using Only Position Feedback: An Observer Backstepping Approach. In: *Proc. of the 35th IEEE Conf. on Decision and Control (CDC'96)*. pp. 3388–3393.

Hagiwara, H. (1989). Weather Routing of Sail-Assisted Motor Vessels. PhD thesis. University of Delft, The Netherlands.

Hasselmann et al. (1973). Measurements of Wind-Wave Growth and Swell Decay during the Joint North Sea Wave Project (JONSWAP). *Deutschen Hydrografischen Zeitschrift*.

Hauser, J. and R. Hindmann (1995). Maneuvering Regulation from Trajectory Tracking: Feedback Linearizable Systems. In: *Proc. IFAC Symposium on Nonlinear Control Systems Design*. Lake Tahoe, CA. pp. 595–600.

Healey, A. J. and D. B. Marco (1992). Slow Speed Flight Control of Autonomous Underwater Vehicles: Experimental Results with the NPS AUV II. In: *Proceedings of the 2nd International Offshore and Polar Engineering Conference (ISOPE)*. San Francisco, CA. pp. 523–532.

Healey, A. J. and D. Lienard (1993). Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles. *IEEE Journal of Ocean Engineering* **OE-18**(3), 327–339.

Healey, A. J., S. M Rock, S. Cody, D. Miles and J. P. Brown (1995). Toward an improved understanding of thruster dynamics for underwater vehicles. *IEEE Journal of Oceanic Engineering* **JOE-29**(4), 354–361.

Hearns, E., R. Katebi and M. Grimble (2000). Robust Fin Roll Stabilizer Controller Design. In: *Proceedings of the 5th IFAC Conference on Maneuvering and Control of Marine Craft*. Aalborg, Denmark. pp. 81–86.

Hearns, G. and M. Blanke (1998). Quantitative Analysis and Design of a Rudder Roll Damping Controller. In: *Proc. IFAC Conference on Control Applications in Marine Systems (CAMS'98)*. pp. 115–120. Fukuoka, Japan.

Heiskanen, W. A. and H. Moritz (1967). *Physical Geodesy*. Freeman. London.

Hickey, N. A., M. A. Johnson, M. R. Katebi and M. J. Grimble (2000). PID Controller Optimization for Fin Roll Stabilization. In: *Proceedings of the IEEE Conference of Control Applications (CCA'2000)*. Hawaii, USA. pp. 1785–1790.

Hickey, N. A., M. J. Grimble, M. A. Johnson, M. R. Katebi and R. Melville (1997). Robust Fin Roll Stabilization of Surface Ships. In: *Proceedings of the IEEE Conference on Decision and Control (CDC'97)*. Vol. 5. pp. 4225–4230.

Hofmann-Wellenhof, B., H. Lichtenegger and J. Collins (1994). *Global Positioning System: Theory and Practice*. 3rd ed.. Springer Verlag. New York.

Holzhüter, T. (1990). A High Precision Track Controller for Ships. In: *Preprints of the 11th IFAC World Congress*. Tallinn, Estonia. pp. 118–123.

Holzhüter, T. (1992). On the Robustness of Course Keeping Autopilots. In: *Proceedings of IFAC Workshop on Control Applications in Marine Systems (CAMS'92)*. Genova, Italy. pp. 235–244.

Holzhüter, T. (1997). LQG Approach for the High-Precision Track Control of Ships. *IEE Proceedings on Control Theory and Applications* **144**(2), 121–127.

Holzhüter, T. and H. Strauch (1987). A Commercial Adaptive Autopilot for Ships: Design and Experimental Experience. In: *Proceedings of the 10th IFAC World Congress*. July 27-31, Munich, Germany. pp. 226–230.

Holzhüter, T. and R. Schultze (1996). On the Experience with a High-Precision Track Controller for Commercial Ships. *Control Engineering Practise* **CEP-4**(3), 343–350.

Hughes, P. C. (1986). *Spacecraft Attitude Dynamics*. John Wiley & Sons Ltd. New York.

Humphreys, D. E. and K. W. Watkinson (1978). Prediction of the Acceleration Hydrodynamic Coefficients for Underwater Vehicles from Geometric Parameters. Technical Report NCSL-TR-327-78. Naval Coastal System Center. Panama City, Florida.

Imlay, F. H. (1961). The Complete Expressions for Added Mass of a Rigid Body Moving in an Ideal Fluid. Technical Report DTMB 1528. David Taylor Model Basin. Washington D.C.

Ingram, M. J., R. C. Tyce and R. G. Allen (1996). Dynamic Testing of State of the Art Vertical Reference Units. In: *Proc. of the Oceans 96 MTS/IEEE*. IEEE. pp. 1533–1538.

Isherwood, R. M. (1972). Wind Resistance of Merchant Ships. In: *RINA Trans. Vol. 115*. pp. 327–338.

Isidori, A. (1989). *Nonlinear Control Systems*. Springer-Verlag. Berlin.

ISO 2631-1 (1997). Mechanical Vibration and Shock. Evaluation of Human Exposure to Whole-Body Vibration – Part 1: General Requirements.

ISO 2631-3 (1985). Evaluation of Human Exposure to Whole-Body Vibration – Part 3: Evaluation of Whole Body z-axis Vertical Vibration in the Frequency Range 0.1 to 0.63 Hz.

Jacobson, D. H. (1977). *Extensions to Linear-Quadratic Control, Optimization and Matrix Theory*. Academic Press. New York.

Jiang, Z. P. (2002). Global Tracking Control of Underactuated Ships by Lyapunov's Direct Method. *Automatica* **AUT-38**(2), 301–309.

Jiang, Z.-P., A. Teel and L. Praly (1994). Small-Gain Theorem for ISS Systems and Applications. *Math. Control Signals and Systems* **MCSS-7**(2), 95–120.

Jiang, Z.-P. and H. Nijmeijer (1999). A Recursive Technique for Tracking Control of Nonholonomic Systems in Chained Form. *IEEE Transactions on Automatic Control* **TAC-4**(2), 265–279.

Johansen, T. A., T. I. Fossen and P. Tøndel (2002). Efficient Optimal Constrained Control Allocation via Multi-Parametric Programming. *AIAA Journal of Guidance, Control and Dynamics*. submitted.

Johansen, T. A., T. I. Fossen and S. P. Berge (2003). Constraint Nonlinear Control Allocation with Singularity Avoidance using Sequential Quadratic Programming. *IEEE Transactions on Control Systems Technology*. submitted.

Jouffroy, J. and J. Lottin (2002*a*). Integrator Backstepping uisng Contraction Theory: A Brief Methodological Note. In: *Proc. of the IFAC World Congress*. Barcelona, Spain.

Jouffroy, J. and J. Lottin (2002*b*). On the use of Contraction Theory for the Design of Nonlinear Observers for Ocean Vehicles. In: *IEEE American Control Conference (ACC'02)*. Vol. 4. Danvers, MA. pp. 2647–2652.

Jouffroy, J. and J. Lottin (2003). Remarks on "Nonlinear Output Feedback Control of Underwater Vehicle Propellers using Feedback from Estimated Axial Flow Velocity". *IEEE Journal of Oceanic Engineering*. to appear.

Jurdjevic, V. and J. P. Quinn (1978). Controllability and Stability. *Journal of Differential Equations* **JDE-28**, 381–389.

Källström, C. G. (1981). Control of Yaw and Roll by Rudder/Fin Stabilization System. In: *Proceedings of the 6th International Ship Control Systems Symposium (SCSS'81), Vol 2, Paper F2 3-1*. Ottawa, Canada.

Källström, C. G. (1987). Improved Operational Effectiveness of Naval Ships by Rudder Roll Stabilization. In: *NAVAL'87, Asian Pacific Naval Exhibition and Conference*. Singapore.

Källström, C. G. and K. Theoren (1994). Rudder-Roll Stabilization an Improved Control Law. In: *Proc. of the IEEE Conference on Control Applications*. pp. 1099–1105. New York, NY.

Källström, C. G. and W. L. Schultz (1990). An Integrated Rudder Control System for Roll Damping and Maintenance. In: *Proc. of the 9th Int. Ship Control Systems Symposium (SCSS'90)*. Bethesda, MD. pp. 3.278–3.296.

Källström, C. G., P. Wessel and S. Sjölander (1988). Roll Reduction by Rudder Control. In: *Proc. of the SNAME Spring Meeting/STAR Symposium*. Pittsburg, Pennsylvania. pp. 67–76.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *ASME Transactions, Series D: Journal of Basic Engineering* **JBE-82**, 35–42.

Kalman, R. E. and R. S. Bucy (1961). New Results in Linear Filtering and Prediction Theory. *ASME Transactions, Series D: Journal of Basic Engineering* **JBE-83**, 95–108.

Kaminer, I., A. Pascoal, C. J. Silvestre and P. P. Khargonekar (1990). Control of an Underwater Vehicle using H-infinity Synthesis. In: *Proc. of the IEEE Conference on Decision and Control*. Brighton, UK. pp. 2350–2355.

Kane, T. R., P. W. Likins and D. A. Levinson (1983). *Spacecraft Dynamics*. McGraw-Hill. NY, NY.

Kanellakopoulos, I., P. V. Kokotovic and A. S. Morse (1992). A Toolkit for Nonlinear Feedback Design. *Systems and Control Letters* **SCL-18**, 83–92.

Katebi, M. R., D. K. K. Wong and M. J. Grimble (1987). LQG Autopilot and Rudder Roll Stabilization Control System Design. In: *Proceedings of the 8th International Ship Control Systems Symposium (SCSS'87)*. The Hague, The Netherlands. pp. 3.69–3.84.

Katebi, M. R., M. J. Grimble and Y. Zhang (1997*a*). H_∞ Robust Control Design for Dynamic Ship Positioning. In: *IEE Proceedings on Control Theory and Applications*. Vol. 144 2. pp. 110–120.

Katebi, M. R., N. A. Hickey and M. J. Grimble (2000). Evaluation of Fin Roll Stabilizer Controller Design. In: *Proccedings of the 5th IFAC Conference on Maneuvering and Control of Marine Craft*. Aalborg, Denmark. pp. 69–74.

Katebi, M. R., Y. Zhang and M. J. Grimble (1997*b*). Nonlinear Dynamic Ship Positioning. In: *Proceedings of the 13th IFAC World Cogress*. Vol. Q. pp. 303–308.

Kayton, M. and W. L. Fried (1997). *Avionics Navigation Systems*. John Wiley and Sons Ltd.. New York.

Kempf, G. (1932). Measurements of the Propulsive and Structural Characteristic of Ships. In: *Transactions of SNAME, vol. 40*. pp. 42–57.

Kempf, G. (1944). Manöveriemorm für Schiffe. In: *Hansa, Deutsche Schiffahrs-zeitschrift, Heft 27/28*. pp. 372–276. (in German).

Khalil, H. K. (2002). *Nonlinear Systems*. MacMillan. New York.

Kirchhoff, G. (1869). *Über die Bewegung eines Rotationskorpers in einer Flussigkeit*. Crelle's Journal, No. 71, pp. 237–273 (in German).

Kitamura, F., H. Sato, K. Shimada and T. Mikami (1997). Estimation of Wind Force Acting on Huge Floating Ocean Structures. In: *Proc. of the Oceans '97. MTS/IEEE Conference*. Vol. 1. pp. 197–202.

Koditschek, D. E. (1987). Adaptive Techniques for Mechanical Systems. In: *Proc. Of the 5th Yale Workshop on Adaptive Systems*. New Haven, CT. pp. 259–265.

Kokotovic, P. V. (1991). The Joy of Feedback: Nonlinear and Adaptive. *IEEE Control Systems Magazine* **ICSM-12**, 7–17. Bode Price Lecture.

Kokotovic, P. V. and H. J. Sussmann (1989). A Positive Real Condition for Global Stabilization of Nonlinear Systems. *Systems and Control Letters* **SCL-13**, 125–133.

Koyama, T. (1967). *On the Optimum Automatic Steering System of Ships at Sea*. J.S.N.A., Vol. 122.

Krstic, M. and H. Deng (1998). *Stabilization of Nonlinear Uncertain Systems*. Springer-Verlag. Berlin.

Krstic, M., I. Kanellakopoulos and P. V. Kokotovic (1995). *Nonlinear and Adaptive Control Design*. John Wiley and Sons Ltd. New York.

Kvam, K., K. Ohtsu and T. I. Fossen (2000). Optimal Ship Manoeuvring using Bryson and Ho's Time Varying LQ Controller. In: *Proc.of the IFAC Conference on Maneuvering and Control of Marine Craft (MCMC'00)*. Aalborg, Denmark. pp. 351–356.

Lamb, H. (1932). *Hydrodynamics*. Cambridge University Press. London.

Landau, Y. D. (1979). *Adaptive Control-The Model Reference Approach*. Marcel Dekker Inc. New York, NY.

LaSalle, J. and S. Lefschetz (1961). *Stability by Lyapunov's Direct Method*. Academic Press. Baltimore, MD.

LaSalle, J. P. (1966). Stability Theory for Ordinary Differential Equations. *Journal of Differential Equations* **JDE-4**, 57–65.

Lauvdal, T. and T. I. Fossen (1997). Nonlinear Rudder-Roll Damping of Non-Minimum Phase Ships Using Sliding Mode Control. In: *Proceedings of the European Control Conference*. Brussels, Belgium.

Lauvdal, T. and T. I. Fossen (1998*a*). Robust Adaptive Ship Autopilot with Wave Filter and Integral Action. *Special Issue on Marine Systems Control: International Journal of Adaptive Control and Signal Processing* **IJACSP-12**(8), 605–622.

Lauvdal, T. and T. I. Fossen (1998*b*). Rudder Roll Stabilization of Ships Subject to Input Rate Saturations Using a Gain Scheduled Control Law. In: *Proceedings of the IFAC Conference on Control Applications in Marine Systems (CAMS'98)*. pp. 121–127. Fukuoka, Japan.

Leick, A. (1995). *GPS: Satellite Surveying*. Chap. Appendix G, pp. 534–537. John Wiley and Sons Ltd.

Leonard, N. E. (1997). Stability of a Bottom-heavy Underwater Vehicle. *Automatica* **AUT-33**(3), 331–346.

Leonard, N. E. and J. G. Graver (2001). Model-Based Feedback Control of Autonomous Underwater Gliders. *IEEE Journal of Oceanic Engineering* **JOE-26**(4), 633–645.

Lewis, E. V., Ed.) (1989). *Principles of Naval Architecture*. 2nd ed.. Society of Naval Architeets and Marine Engineers (SNAME).

Lewis, F. M. (1967). The Motion of Ships in Waves. In: *Principles of Naval Architecture* (J. P. Comstock, Ed.). pp. 669–692. The Society of Naval Arehitects and Marine Engineers, 601 Pavonia Avenue, Jersey City, NJ 07306.

Lewis, P. C. (1951). Differential Equations Referred to a Variable Metric. *American Journal of Mathematics* **AJM-73**, 48–58.

Lin, Ching-Fang (1992). *Modern Navigation, Guidance and Control Processing*. Prentice-Hall Inc. Englewood Cliffs, New Jersey 07632.

Lindegaard, K.-P. (1997). Nonlinear Tracking Control and Station-Keeping of Underactuated Ships. Master's thesis. Norwegian University of Science and Technology, Dept. of Eng. Cybernetics.

Lindegaard, K.-P. (2003). Acceleration Feedback in Dynamic Positioning Systems. PhD thesis. Department of Engineering Cybernetics, Norwegian University of Science and Technology. Trondheim.

Lindegaard, K. P. and T. 1 . Fossen (2001*a*). A Model Based Wave Filter for Surface Vessels using Position, Velocity and Partial Acceleration Feedhack. In: *Proc. of the IEEE Conf. on Decision and Control (CDC'2001)*. lEEE. Orlando, FL. pp. 946–951.

Lindegaard, K. P. and T. 1. Fossen (2001*b*). On Global Model Based Observer Designs for Surface Vessels. In: *5th IFAC Conf. on Control Applications in Marine Systems (CAMS'2001)*. Elsevier Science.

Lindegaard, K.-P. and T. 1. Fossen (2002). Fuel Efficient Rudder and Propeller Control Allocation for Marine Craft: Experiments with a Model Ship. *IEEE Transactions on Control Systems Technology*. To appear.

Lindfors, 1. (1993). Thrust Allocation Method for the Dynamic Positioning System. In: *10th International Ship Control Systems Symposium (SCSS'93)*. Ottawa, Canada. pp. 3.93–3.106.

Liu, H., R. R. Costa, F. Lizarralde and J.P.V.S. Da Cunha (2000). Dynamic Positioning of Remotely Operated Underwater Vehicles. *IEEE Robotics and Automation Magazine* **RAM-7**(3), 21–31.

Lloyd, A. E. J. M. (1975). Roll Stabilization by Rudder. In: *Proceedings of the 4th International Ship Control Systems Symposium (SCSS'75)*. The Hague, The Netherlands.

Lloyd, A. R. J. M. (1989). Seakeeping; Ship Behavior in Rough Water. Ellis Horwood Ltd.

Lo, H. (1991). Dynamic Ship Routing through Stochastic, Spatially Dependent Ocean Currents. PhD thesis. The Ohio State University, Columbus, OH.

Lo, H. K. and M. R. McCord (1995). Routing through Dynamic Ocean Currents: General Heuristics and Empirical Results in the Gulf Stream Region. *Transportation Research Part B: Methodological* **29**(2), 109–124.

Lo, H. K. and M. R. McCord (1998). Adaptive Ship Routing through Stochastic Ocean Currents: General Formulations and Empirical Results. *Transportation Research Part A: Policy and Practice* **32**(7), 547–561.

Lohmiller, W. and J.-J. Slotine (1996). On Metric Controllers and Observers for Nonlinear Systems. In: *Proc. of the 35th Conf. on Decision and Control*. Kobe, Japan. pp. 1477–1482.

Lohmiller, W. and J.-J. Slotine (1998). On Contraction Analysis for Nonlinear Systems. *Automatica* **AUT-34(6)**, 683–696.

Lohmiller, W. and J.-J.E. Slotine (2000). Control System Design for Mechanical Systems using Contraction Theory. *IEEE Transactions on Automatic Control* **TAC-45**(5), 984–989.

Lohmiller, W. S. (1999). Contraction Analysis of Nonlinear Systems. PhD thesis. Department of Mechanical Engineering, MIT, MA.

Loria, A., T. I. Fossen and A. Teel (1999). UGAS and ULES of Non-Autonomous Systems: Applications to Integral Action Control of Ships and Manipulators. In: *Proc. Of the 5th European Control Conference (ECC'99)*. Karlsruhe, Germany,.

Loria, A., T. I. Fossen and E. Panteley (2000). A Separation Principle for Dynamic Positioning of Ships: Theoretical and Experimental Results. *IEEE Transactions on Control Systems Technology* **TCST-8**(2), 332–343.

Lozano, R., B. Brogliato and I. D. Landau (1992). Passivity and Global Stabilization of Cascaded Nonlinear Systems. *IEEE Transactions on Automatic Control* **TAC-37**, 1386–1388.

Lozano, R., B. Brogliato, O. Egeland and B. Maschke (2000). *Dissipative Systems Analysis and Control. Theory and Applications*. Springer Verlag.

Lyapunov, M. A. (1907). Probleme Général de la Stabilité de Mouvement. *Ann. Fac. Sci. Toulouse* **Vol. 9**, 203–474. (Translation of a paper published in Comm. Soc. math. Kharkow 1893, reprinted in Ann. math Studies, Vol. 17, Princeton 1949).

Lyshevski, S. E. (2001). Autopilot Design for Highly Maneuverable Multipurpose Underwater Vehicles. In: *Proc. of the American Control Conference*. Arlington, VA. pp. 131–136.

Marco, D. B. and A. J. Healey (2001). Command, Control, and Navigation Experimental Results with the NPS ARIES AUV. *IEEE Journal of Oceanic Engineering* **JOE-26**(4), 466–476.

Matrosov, V. M. (1962). On the Stability of Motion. *Prikl. Mat. Meh.* **PMM-26**, 885–895.

Maybeck, P. S. (1979). *Stochastic Models, Estimation and Control*. Academic Press. New York.

McCord, M. and S. Smith (1995). Beneficial Voyage Characteristics for Routing through Dynamic Currents. *Transportation Research Record* **1511**, 19–25.

McGookin, E. W., D. J. Murray-Smith, Y. Lin and T. I. Fossen (2000*a*). Experimental Results from Supply Ship Autopilot Optimisation Using Genetic Algorithms. *Transactions of the Institute of Measurement and Control* **TIMC-22**(2), 141–178.

McGookin, E. W., D. J. Murray-Smith, Y. Lin and T. I. Fossen (2000*b*). Ship Steering Control System Optimisation Using Genetic Algorithms. *Journal of Control Engineering Practise* **CEP-8**, 429–443.

McLean, M. B. (1991). Dynamic Performance of Small Diameter Tunnel Thrusters. Master's thesis. Naval Postgraduate School, Monterey, CA.

Meirovitch, L. (1990). *Dynamics and Control of Structures*. Wiley Interscience. NY, NY.

Meirovitch, L. and M. K. Kwak (1989). State Equations for a Spacecraft With Flexible Appendages in Terms of Quasi-Coordinates. *Applied Mechanics Reviews* **42**(11), 161–170.

Milne-Thomson, L. M. (1968). *Theoretical Hydrodynamics*. MacMillan Education Ltd. London.

Minorsky, N. (1922). Directional Stability of Automatic Steered Bodies. *J. Amer. Soc. of Naval engineers* **34**(2), 280–309.

Morgan, J. M., Ed.) (1978). *Dynamic Positioning of Offshore Vessels*. Petroleum, Tulsa, OK.

Munk, T. and M. Blanke (1987). Simple Command of a Complex Ship. In: *Proc. of the 8th Ship Control Systems Symposium*. The Hague.

Narendra, K. S. and A. M. Annaswamy (1989). *Stable Adaptive Systems*. Prentice Hall Inc.. Boston, MA.

Neumann, G. (1952). *On Wind-Generated Ocean Waves with Special Reference to the Problem of Wave Forecasting*. New York University, College of Eng. Res. Div., Dept. of Meteorology and Oceanography. Prepared for the Naval Res.

Newman, J. N. (1977). *Marine Hydrodynamics*. MIT Press. Cambridge, MA.

Nijmeijer, H. and Fossen, T. I., Eds.) (1999). *New Directions in Nonlinear Observer Design*. Springer-Verlag. London.

Nocedal, J. and S. J. Wright (1999). *Numerical Optimization*. Springer-Verlag, New York.

Nomoto, K., T. Taguchi, K. Honda and S. Hirano (1957). On the Steering Qualities of Ships. Technical report. International Shipbuilding Progress, Vol. 4.

Norrbin, N. H. (1963). On the Design and Analyses of the Zig-Zag Test on Base of Quasi Linear Frequency Response. Technical Report B 104-3. The Swedish State Shipbuilding Experimental Tank (SSPA). Gothenburg, Sweden.

Norrbin, N. H. (1965). Zig-Zag Provets Teknik och Analys. Technical Report no. 12. The Swedish State Shipbuilding Experimental Tank (SSPA). Gothenburg, Sweden. (in Swedish).

Norrbin, N. H. (1970). Theory and Observation on the use of a Mathematical Model for Ship Maneuvering in Deep and Confined Waters. In: *Proc. of the 8th Symposium on Naval Hydrodynamics*. Pasadena, California.

Norrbin, N. H. (1972). On the Added Resistance due to Steering on a Straight Course. In: *Proceedings of the 13th ITTC*. Berlin, Hamburg, Germany.

Norsok Standard (1999). Actions and Actions Effects Rev. 1.0. N-003.

OCIMF (1977). Prediction of Wind and Current Loads on VLCCs. Oil Companies International Marine Forum, London, pp. 1–77.

Oda et al., H. (1992). Rudder Roll Stabilization Control System through Multivariable Auto Regressive Model. In: *Proceedings of IFAC Workshop on Control Applications in Marine Systems (CAMS'92)*. Genova, Italy. pp. 113–127.

Oda, H., K. Igarashi and K. Ohtsu (1997). Simulation Study and Full-Scale Experiment of Rudder-Roll Stabilization Systems. In: *Proceedings of the 11th Ship Control Systems Symposium*. pp. 299–313. Southampton, U.K.

Oda, H., K. Ohtsu and T. Hotta (1996). Statistical Analysis and Design of a Rudder-Roll Stabilization System. *Control Engineering Practice* **CEP-4**(3), 351–358.

O'Hanlon, J. F. and M. E. McCauley (1974). Motion Sickness Incidence as a Function of Vertical Sinusoidal Motion. *Aerospace Medicine* **AM-45**(4), 366–369.

Oosterveld, M. W. C. and P. Van Oossanen (1975). Further Computer-Analyzed Data of the Wageningen B-Screw Series. *Int. Shipbuilding Progress* **ISP-22**, 251–262.

Ortega, R. (1991). Passivity Properties for Stabilization of Cascaded Nonlinear Systems. *Automatica* **AUT-27**, 423–424.

Ortega, R., A. Loria, P. J. Nicklasson and H. Sira-Ramirez (1998). *Passivity-Based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications*. Springer Verlag.

Padadakis, N. A. and A. N. Perakis (1990). Deterministic Minimal Time Vessel Routing. *Operations Research* **OR-38**(3), 416–438.

Panteley, E. and A. Loria (1998). On Global Uniform Asymtotic Stability of Nonlinear Time-Varying Noautonomous Systems in Cascade. *Syst. Contr. Letters.* **SCL-33**(2), 131–138.

Park, J., W Chung and J. Yuh (2000). Nonlinear H-infinity Optimal PID Control of Autonomous Underwater Vehicles. In: *Proc. of the 2000 International Symposium on Underwater Technology*. Tokyo, Japan. pp. 193–198.

Parkinson, B. W. and Spilker, J. J., Eds.) (1995). *Global Positioning System: Theory and Applications*. American Institute of Aeronautics and Astronautics Inc.. Washington DC, USA.

Parks, P. C. (1966). Lyapunov Redesign of Model Reference Adaptive Control Systems. *IEEE Transactions on Automatic Control* **TAC-11**, 362–367.

Pascoal, A., P. Oliveira, C. Silvestre, A. Bjerrum A. Ishoy, J.-P. Pignon, G. Ayela and C. Petzelt (1997). MARIUS: An Autonomous Underwater Vehicle for Coastal Oceanography. *IEEE Robotics and Automation Magazine* **RAM-4**(4), 46–59.

Pettersen, K. Y. and H. Nijmeijer (1999*a*). Global Practical Stabilization and Tracking for an Underactuated Ship - A Combined Averaging and Backstepping Approach. *Modeling, Identification and Control* **MIC-20**(4), 189–199.

Pettersen, K. Y. and H. Nijmeijer (1999*b*). Tracking Control of an Underactuated Surface Vessel. In: *Proceedings of the IEEE Conference on Decision and Control*. Phoenix, AZ. pp. 4561–4566.

Pettersen, K. Y. and H. Nijmeijer (2001). Underactuated Ship Tracking Control. *International Journal of Control* **IJC-74**, 1435–1446.

Pettersen, K. Y. and O. Egeland (1996). Exponential Stabilization of an Underactuated Surface Vessel. In: *Proceedings of the IEEE Conference on Decision and Control*. Kobe, Japan. pp. 967–971.

Pettersen, K. Y. and T. I. Fossen (2000). Underactuated Dynamic Positioning of a Ship - Experimental Results. *IEEE Transactions on Control Systems Technology* **TCST-8**(5), 856–863.

Pettersen, K.Y. and O. Egeland (1999). Time-Varying Exponential Stabilization of the Position and Attitude of an Underactuated Autonomous Underwater Vehicle. *IEEE Transactions on Automatic Control* **TAC-44**(1), 112–115.

Phillips, O. M. (1958). The Equilibrium Range in the Spectrum of Wind Generated Waves. *Journal of Fluid Mechanics* **JFM-4**(4), 426–434.

Pierson, W. J. and L. Moskowitz (1963). A Proposed Spectral Form for Fully Developed Wind Seas Based on the Similarity Theory of S. A. Kitaigorodsku. U. S. Naval Oceanograhic Office Contract 62306-1042.

Pinkster, J. A. and U. Nienhuis (1996). Dynamic Positioning of Large Tankers at Sea. In: *Proceedings of the Offshore Technology Conference (OTC'96)*. Houston, TX.

Price, W. G. and R. E. D. Bishop (1974). *Probalistic Theory of Ship Dynamics*. Chapman and Hall. London.

Reid, R. E., A. K. Tugcu and B. C. Mears (1984). The Use of Wave Filter Design in Kalman Filter State Estimation of the Automatic Steering Problem of a Tanker in a Seaway. *IEEE Transactions on Automatic Control* **TAC-29**(7), 577–584.

Reyhanoglu, M. (1996). Control and Stabilization of an Underactuated Surface Vessel. In: *Proceedings of the IEEE Conference on Decision and Control*. Kobe, Japan. pp. 2371–2376.

Roberts, G. N. (1992). Ship Roll Damping Using Rudder and Stabilizing Fins. In: *Proceedings of IFAC Workshop on Control Applications in Marine Systems (CAMS'92)*. Genova, Italy. pp. 129–138.

Marine Control Systems

Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles

Written by one of the experts in the field, this book provides the mathematical foundations and theory needed for the design of ship, rig, and underwater vehicle control systems. The work is highly relevant to marine and control engineers, computer scientists, and many others who work within the field of marine guidance, navigation and control systems. The book focuses on nonlinear kinematics and dynamics, modeling of marine craft with emphasis on control modeling, the exploitation of structural properties, and the effective representation of models in a computer language. For these applications the author has included a new *Guidance, Navigation and Control (GNC)* toolbox for use in Matlab™. The toolbox can be downloaded from http://www.marinecybernetics.com. Several vessel model m-files, Simulink™ models, example scripts, as well as GNC library functions, are included in the toolbox for computer simulations.

The mathematical models are used as the foundation for nonlinear model-based control system design. For this purpose nonlinear design techniques that have been implemented and tested onboard marine craft have been included within the toolbox. These include Lyapunov-based methods like feedback linearization, backstepping, sliding mode control, and energy- and passivity-based design methods. In addition state-of-the art design techniques like PID-control and linear optimal quadratic (optimal) control are discussed. Marine navigation systems are discussed using Kalman filters and nonlinear observers for integration of GPS and inertial sensors.

The book comes with a large number of Matlab™ examples, exercises and case studies, which utilize full-scale vessels and experimental results. Experiments with model ships are used to demonstrate advanced methods.

Thor I. Fossen is Professor of Guidance, Navigation and Control in the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway. He received the M.Sc. degree in Naval Architecture in 1987 and the Dr. Ing. degree in Eng. Cybernetics in 1991. Professor Fossen is the author of *"Guidance and Control of Ocean Vehicles"* (John Wiley & Sons Ltd., 1994) and co-editor of *"New Directions in Nonlinear Observer Design"* (Springer-Verlag, 1999).

27 8292356002 MARINE CONTROL SY kr 850,00
142 07.02.2008 222093
www.sit-tapir.no
9 788292 356005

Roberts, G. N. (1993). A Method to Determine the Applicability of Rudder Roll Stabilization for Ships. In: *Proceedings of the 12th IFAC World Congress*. pp. 1009–1012.

Roberts, G. N. and S. W. Braham (1990). Warship Roll Stabilization Using Integrated Rudder and Fins. In: *9th International Ship Control Systems Symposium (SCSS'90)*. Bethesda, MD. pp. 1.234–1.248.

Robertson, A. and R. Johansson (1998). Comments on "Nonlinear Output Feedback Control of Dynamically Positioned Ships Using Vectorial Observer Backstepping. *IEEE Transactions on Control Systems Technology* **TCST-6**(3), 439–441.

Routh, E. J. (1877). *A Treatise on The Stability on Motion*. Macmillan. London, U.K.

Saberi, A., P. V. Kokotovic and H. J. Sussmann (1990). Global Stabilization of Partially Linear Composite Systems. *SIAM J. Control Opt.* **SJCO-28**, 1491–1503.

Sælid, S. and N. A. Jenssen (1983). Adaptive Ship Autopilot with Wave Filter. *Modeling, Identification and Control* **MIC-4**(1), 33–46.

Sælid, S., N. A. Jenssen and J. G. Balchen (1983). Design and Analysis of a Dynamic Positioning System Based on Kalman Filtering and Optimal Control. *IEEE Transaction on Automatic Control* **TAC-28**(3), 331–339.

Sagatun, S. I. and T.I. Fossen (1991). Lagrangian Formulation of Underwater Vehicles' Dynamics. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. Charlottesville, VA. pp. 1029–1034.

Sagatun, S. I., T. I. Fossen and K.-P. Lindegaard (2001). Inertance Control of Underwater Installations. In: *Proc. of the 5th IFAC Conference on Control Applications in Marine Systems CAMS'2001*. Glasgow, U.K.

Salvesen, N., E. O. Tuck and O. M. Faltinsen (1970). *Ship Motions and Sea Loads*. Trans. SNAME, vol. 78, pp. 250–287.

Samson, C. (1991). Velocity and Torque Feedback Control of a Nonholonomic Cart. In: *Advanced Robot Control, Proc. of the Int. Workshop on Nonlinear and Adaptive Control: Issues in Robotics* (C. Canudas de Wit, Ed.). pp. 125–151. Springer-Verlag.

Sargent, J. S. and P. N. Cowgill (1976). Design Considerations for Dynamically Positioned Utility Vessels. In: *Proceedings of the 8th Offshore Technology Conference*. Dallas.

Sarpkaya, T. (1981). *Mechanics of Wave Forces on Offshore Structures*. Van Nostrand Reinhold Company. New York, NY.

Savage, P. G. (1990). Strapdown Inertial Navigation. Lecture Notes. Strapdown Associates Inc., Minnetonka, MN, USA.

Schuster, M. D. and S. D. Oh (1981). Three-Axis Attitude Determination from Vector Observations. *Journal of Guidance, Control and Dynamics* **JGCD-4**(1), 70–77.

Sciavicco, L. and B. Siciliano (1996). *Modeling and Control of Robot Manipulators*. McGraw-Hill Companies Inc.