NTNU – Trondheim
Norwegian University of
Science and Technology

Faculty of Information Technology
Mathematics and Electrical Engineering
Department of Engineering
Cybernetics

Nonlinear Observers for Parametric Roll

Gunnhild Konstanse Hoff Olsen

June 16, 2012

NTNU
Norwegian University of Science and Technology

Master Thesis

Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Engineering Cybernetics

Supervisor: Professor Thor I. Fossen, Dept. of Eng. Cybernetics, NTNU
Co-Supervisor: Dominik A. Breu, PhD Cand., Dept. of Eng. Cybernetics, NTNU

NTNU
Norwegian University of Science and Technology

Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Engineering Cybernetics

MSC THESIS DESCRIPTION SHEET

Name: Gunnhild Konstanse Hoff Olsen
Department: Engineering Cybernetics
Thesis title (Norwegian): Ulineære Estimatorer for Parametrisk Rull
Thesis title (English): Nonlinear Observers for Parametric Roll

Thesis Description: The purpose of the thesis is to investigate different observers for estimation of the frequency of encounter and the associated sea state parameter for a ship exhibiting parametric roll. This includes estimation of the wave spectrum frequency and the direction of the incoming waves.

The following items should be considered:

1) Use a standard model for parametric roll and express the model in state-space form.
2) Assume that forward speed, roll angle and roll rate are measured. Add wave-induced forces to the nonlinear state-space model using a sine wave and a wave spectrum. Write down the augmented state-space model including the measurement equation.
3) Derive conditions for nonlinear observability.
4) Implement an extended Kalman filter (EKF) for estimation of frequency of encounter and wave spectrum modal frequency.
5) Implement and unscented Kalman filter and compare the results with your findings under item #4.
6) Investigate if it is possible to also estimate the direction of the incoming waves.
7) Conclude your results in a report and include case studies verifying your theoretical derivations.

Start date: 2012-08-09
Due date: 2012-16-06

Thesis performed at: Department of Engineering Cybernetics, NTNU
Supervisor: Professor Thor I. Fossen, Dept. of Eng. Cybernetics, NTNU
Co-Supervisor: Dominik A. Breu, PhD Cand., Dept. of Eng. Cybernetics, NTNU

Til familien.

Abstract

Parametric roll resonance is a dangerous resonance phenomenon affecting several kinds of ships, such as cruise ships, fishing vessels and container ships. In a worst case scenario parametric roll resonance may lead to roll angles of up to 50 degrees, capsizing of the vessel, or damage of goods and ship for tens of millions of dollars, [22].

Accurate equations in the model for the ship motion are important because they are directly related to the observer design. There has been done allot of mathematical investigations on the problem. In this work, we use a standard model for parametric roll, and express it in state-space form. We establish an augmented state-space model by adding wave-induced forces to the nonlinear state-space model.

We revise Lie Derivatives, and use an observability check, for nonlinear systems, to derive conditions for nonlinear observability of our system.

A Matlab/Simulink model is implemented, and shows that the state space model is good for simulation of a ship exhibiting parametric roll.

The main goal of this thesis has been to investigate different observers for state estimation of a ship exhibiting parametric roll. The wave encounter frequency is estimated by implementing an extended Kalman filter, and in turn an unscented Kalman filter. By comparing the two filters, we conclude that the unscented Kalman filter has better empirical results for estimation of the states of a ship in parametric roll resonance condition.

Also attention has been paid to estimation of the direction of the incoming waves. In this thesis the wave direction has been estimated by implementing an extended Kalman filter, and consequently by implementing speed and heading controllers.

Norsk Sammendrag

Parametrisk rull resonans er et farlig resonansfenomen som påvirker flere typer skip. Cruiseskip, fiskebåter og containerskip er mest utsatt grunnet deres spesielle skrogform. I verste fall kan parametrisk rulling nå en vinkel på opp til 50 grader, kantring, eller det kan forårsake skader på skip og last til millioner av dollar, [22].

Det er viktig med en nøyaktig modell for skipet og dets bevegelser, ettersom den er direkte knyttet til designet av tilstandsestimatoren. Det er gjort mye matematisk forskning på problemet, og teoretisk er det derfor nøye beskrevet hvordan parametrisk resonans oppstår. I dette arbeidet, bruker vi en standard modell for parametrisk rull, og skriver den på tilstandsromform. Vi etablerer en utvidet tilstandsrommodell ved å legge krefter forårsaket av bølger til den ulineære modellen.

Vi må sjekke om det ulineære systemet er observerbart, og bruker Lie derivater for å utlede vilkår for ulineær observerbarhet for modellen vår.

En Matlab/Simulink modell er implementert, og viser at tilstandsrommodellen er god for simulering av et skip som opplever parametrisk rull.

Hovedmålet med denne masteroppgaven har vært å teste ut ulike ulineære tilstandsestimatorer for estimering av et skip som opplever parametrisk rull. Møtefrekvensen er estimert ved å implementere et extended Kalman filter, og deretter et unscented Kalman filter. Etter sammenligning av de to tilstandsestimatorene, konkluderer vi med at unscented Kalman filter gir bedre empiriske resultater for estimering av tilstander for et skip i parametrisk rull resonans tilstand.

Det har også blitt viet tid til estimering av retningen på innkommende bølger. I denne tesen er retningen på innkommende bølger estimert ved implementasjon av et extended Kalman filter, og følgelig ved å implementere kontrollere for fart og retning.

Preface

This report is submitted in partial fulfilment of the requirements for the master's programme, MSc, of Engineering Cybernetics at the Norwegian University of Science and Technology, NTNU. It represents the results of my master thesis carried out January 2012 - June 2012. My supervisor has been Professor Thor I. Fossen of the Department of Engineering Cybernetics, NTNU, and my Co-Supervisor has been Dominik A. Breu, PhD Candidate, CeSOS, NTNU.

Acknowledgements

There are several people I would like to thank. First of all my supervisor Thor I. Fossen of the Department of Engineering Cybernetics, NTNU, and my Co-Supervisor PhD. Candidate Dominik A. Breu, CeSOS, NTNU. Their help has made this report possible.

I send greetings and thanks to my co-students for enjoyable company and friendship.

Thanks also goes out to my friends at my student residence, Teknobyen Studentboliger. You guys make every day an adventure.

I would also like to thank my family, and close friends. Especially the family dog, Chica. When I'm home on visits, she always greets me with the outmost happiness.

To people who feel left out - my deep apologies.

Gunnhild Konstanse Hoff Olsen
Trondheim, June 2012

Contents

List of Figures

List of Tables

List of Algorithms

Part I

Preliminaries

Nomenclature

Here, all the acronym and variables used in the thesis are listed.

Acronyms & Abbrevations

ABS	American Bureau of Shipping
BODY	Body-fixed reference rame
DoF	Degrees of Freedom
ECEF	Earth Centered Earth Fixed reference frame
ECI	Earth Centered Inertial reference frame
EKF	Extended Kalman Filter
FFT	Fast Fourier Transform
IFAC	International Federation of Automatic Control
IMO	International Maritime Organization
IMU	Inertial Measurement Unit
ISA	Inertial Sensor Assembly
JONSWAP	Joint North Sea Wave Project
KF	Kalman Filter
LF	Low-Frequency
MCH	Metacentric Height
NED	North East Down reference frame
ODE	Ordinary Differential Equation
p.d.f	Power Density Function
RP	Random Process
SISO	Single input Single output
SNAME	Society of Naval Architects and Marine Engineers
UKF	Unscented Kalman Filter
UT	Unscented Transformation
WF	Wave Frequency
WSS	Wide Sense Stationary

Greek letters

α_ϕ	-	Phase of the change in the linear spring term
ϕ	-	Roll angle
$\dot{\phi}$	-	Rate of roll angle
∇	-	Displaced water volume
ω_0	-	Modal wave frequency
ω_e	-	Encounter frequency
ρ	-	Water density
ν	-	Kinematic viscosity
τ_1	-	Propeller thrust
λ	-	Relative damping factor
σ	-	σ_{gain}
ω_0	-	Resonance frequency
Φ	-	Discretization matrix
∇	-	Discretization matrix
Γ	-	Discretization matrix
σ	-	Standard deviation
σ^2	-	Variance
χ	-	Sigma points
α	-	Scaling parameter
κ	-	Scaling parameter
β	-	Scaling parameter
ψ	-	Heading
ψ_d	-	Desired heading
χ	-	Course angle
β_w	-	Wave direction
β_w^n	-	Wave angle expressed in the NED reference frame
δ	-	Rudder angle
$\tau_{control}$	-	Forces due to control
τ_{wind}	-	Forces due to wind
τ_{waves}	-	Forces due to waves
λ_1	-	Weighting factor
λ_2	-	Weighting factor

Roman letters

m_{44}	-	Sum of the rigid-body moment of inertia and added mass in roll
d_{44}	-	Linear damping parameter
k_{44}	-	Linear spring term
$k_{\phi t}$	-	Amplitude of the change in the linear spring term
I_x	-	Rigid-body Inertia (inertia in roll)
A_{44}	-	Added mass
B_{44}	-	Linear damping coefficient
C_{44}	-	Restoring moment factor
K_{ϕ^3}	-	Restoring force
$\overline{GM}_m$	-	Mean value of metacentric height, MCH
$\overline{GM}_a$	-	Amplitude of MCH change
m	-	Mass of the ship
A_{11}	-	Added mass
S	-	Wetted surface (intercepted water plane area)
g	-	Gravitational acceleration
L_{pp}	-	Ship length between perpendiculars
u	-	Ship forward speed
B_{11}	-	Non-linear hydrodynamic resistance
k	-	Form factor
R_n	-	Reynolds number
r_x	-	Roll radius of gyration
B	-	Beam amidships
T	-	Draught amidships
GM_t	-	Transverse MCH
$h_{wave}(s)$	-	1st-order wave-induced disturbances
K_w	-	Gain of the transfer function $h_{wave}(s)$
w	-	White noise
u	-	Control signal
u_{nl}	-	Term to ”cancel” process’ nonlinearities
v_c	-	Term for control of linear system
k_p	-	Feedback linearization controller gain
m	-	Mean
$E[X^j]$	-	j.th order moment
con	-	Covariance
R_x	-	Autocorrelation
S_x	-	Power spectrum
$\mathcal{O}$	-	Observability matrix
L	-	Dimension of state vector

Q	-	Process noise covariance
R	-	Measurement noise covariance
$W_i^{(m)}, W_i^{(c)}$	-	Weights
K	-	Kalman filter gain
x	-	State vector
$\hat{x}$	-	Estimated states
y	-	Measurement vector
z	-	Measurement equation
u	-	Control input
A	-	Jacobian
P	-	Covariance
P_{xy}	-	Cross-covariance
$U(t)$	-	Time-varying forward speed
u	-	Surge velocity
v	-	Sway velocity
r	-	Yaw rate
M_{RB}	-	Rigid-body mass matrix
M_A	-	Hydrodynamic added mass matrix
C_{RB}	-	Coriolis-centripetal matrix
$d(V_{rc}, \gamma_c)$	-	Current and damping forces
K	-	Static yaw rate gain
T_1	-	Time constant
T_2	-	Time constant
T_3	-	Time constant
T	-	Effective yaw rate time constant
K_p	-	Proportional gain
K_d	-	Derivative gain

Chapter 1

Introduction

This chapter provides some perspective to the work described in this thesis, defines the problem, states the main contributions I believe that the thesis has to offer, and finally presents an outline of the rest of the master thesis.

1.1 Motivation

Among problems in ship dynamics, parametric roll resonance is an interesting topic, due to its high complexity, and the problems it may cause. The physics are quite well understood, and nowadays there is allot of ongoing research to try to design controllers which attempt to suppress the parametrically excited rolling.

The occurrence of the parametric roll phenomenon is related to the hull forms that experience large volumetric changes, in the submerged portion, during the wave passage. Most commercial vessels have hull-forms to maximize cargo capacity, with large bow and stern flares, such as containerships. A longitudinal sea can yield parametrically excited transversal oscillations, which may rapidly increase the amplitude and finally lead to capsizing of the vessel.

The emergence of parametric roll in container ships has refocused attention on the problem. Studies are underway at the American Bureau of Shipping (ABS), [1], the Society of Naval Architects & Marine Engineers (SNAME), [42], the International Maritime Organization (IMO), [30], and others.

In this thesis, observability and nonlinear state estimation for parametric resonance is investigated. The purpose of the thesis is to investigate different observers for

estimation of the frequency of encounter and the associated sea state parameter for a ship exhibiting parametric roll. This includes estimation of the wave spectrum frequency and the direction of the incoming waves.

The parametric roll resonance phenomenon may induce costly ship operation problems; in the last decades serious accidents of parametric rolling were reported that have resulted in loss and damage of cargo, [22]. Due to its practical importance and modelling challenges the phenomenon has been an attractive research subject during the last years.

This thesis is motivated by maritime disasters caused by the parametric roll resonance phenomenon, and the financial and dangerous situations that may occur. The main concern related to container ships is the loss of containers overboard; a worry to shippers and insurers due to the high value cargoes carry. Two spectacular incidents are described below.

Another motivation for this project is damping of the roll for less dramatic and more common reasons; like preventing seasickness, preventing injuries due to falls, making it easier to work on the boat, avoiding potential capsize, and comfort while at sea.

1.1.1 APL China

In October 1998, the APL vessel APL China, eastbound from Kaohsiung to Seattle, encountered a storm in the North Pacific Ocean. During the 12 hours long storm, roll angles as great as 35 – 40 degrees were reported. The container ship was the victim of the largest ever loss of containers while on passage. 388 containers went overboard, and in addition, 400 containers were damaged in the hold. Containers and cargoes hung over both sides of the vessel, as can be seen in Figure 1.1.

Lawyers estimated that the lost cargo was worth more than the value of the ship, more than $50 million. This may have been the largest maritime shipping loss in history. (Account taken from [22]).

1.1.2 Maersk Carolina

The Maersk Carolina incident happened in January 2001. The Panamax container vessel encountered a storm in the North Atlantic sailing from Algeciras, Spain, to Halifax, Nova Scotia, Canada. The ship was loaded with a total cargo of 36 021

Figure 1.1: The APL China Incident. Images from `www.cargolaw.com`.

metric tons. Driving with reduced speed turned directly into the waves the vessel unexpectedly began rolling upwards of 47 degrees. The large roll angle had built up in just a few cycles. 133 containers were lost overboard, and 50 others severe or moderately damaged. Cargo claims exceeded $4 million, and the vessel itself had structural damage. (Account taken from [22]).

In Figure 1.2, you can view some of the damage done to the cargo.

Figure 1.2: The Maersk Carolina arriving Halifax Harbor. Images from `www.cargolaw.com`.

1.1.3 Literature Overview

Observations of parametric resonance dates all the way back to 1831, when Faraday reported crispations produced when a large glass of water was made to sound by passing a wet finger around the edge, [14]. In 1859, the German physicist Melde carried out an experiment, and observed that a periodic variation in the tension of a taut string parametrically excites transverse waves in the string when the frequency of the change of the tension is about twice the natural frequency of any transverse mode. The French mathematician and astronomer Mathieu gave a mathematical description of the phenomenon of parametric resonance in 1868, [35].

Paulling started research on parametric roll in Germany in the 1930's, [43]. In their collaborative work, Paulling and Rosenberg continued the study of parametric roll for ships in the 1950's, [44].

In recent years mathematical models, with varying complexity, have been proposed to describe the dynamics of ships subject to parametric resonance.

Single degrees of freedom (DoF) models have been widely used to analyze the critical parameters of the phenomenon, and derive stability conditions. The Mathieu equation is the simplest model of parametric resonance, and can describe the onset of instability in regular waves. Hence many of the models rely on the Mathieu equation. But because the Mathieu equation has a linear stiffness term, it is incapable of predicting the amplitude of roll, and not applicable in regular waves. Still ABS used it, to establish estimates for ship susceptibility criteria to parametric excitation, [1].

A 1.5 DoF model was proposed by Bulian, [11]. Dynamic interaction between the vertical motions and the roll oscillation was relaxed by the assumption of quasi-static heave and pitch.

Galeazzi, [21], makes use of a second order model for parametric resonance, roll and surge, for which he analyses the effect of non-constant speed.

Neves and Rodríguez, [39], proposed a derivative mathematical model in which the heave, roll and pitch motions, and wave passage effects, were described with coupling terms up to the third order. They used this to study parametric resonance of a fishing vessel in regular waves.

Holden, [29], deals with a third order model for parametric roll, based on the model

by Neves and Rodríguez. A Matlab/Simulink[1] benchmark for the simulation of parametric roll resonance for a large container ship has been implemented and validated against experimental results.

Breu and Fossen, [8], use both heading and speed in an attempt to avoid parametric resonance. Extremum seeking control is applied to ships experiencing parametric roll resonance, and controllers are developed to guarantee asymptotically stable origins of the tracking error dynamics.

In more complex models with 6 DoF the irregular sea surface are often substituted with either an effective wave spectrum or an equivalent wave group. This gives an advantage compared to full nonlinear calculations in the sense that it has much faster computational speed.

1.2 Problem Definition

1.2.1 Parametric Resonance

When the driving frequency equals the natural frequency of the oscillator in a system, the energy per cycle transferred to the oscillator is maximum. The natural frequency of the system is thus called the resonance frequency, [50].

Parametric resonance is the parametrical resonance phenomenon of mechanical excitation and oscillation at certain frequencies, [37]. This effect exhibits the instability phenomenon. In parametric resonance the amplitude of the unstable solution grows exponentially to infinity. Figure 1.3 picture the amplitude of oscillation against frequency. As the natural frequency of the system is approached, the amplitude of the oscillation becomes greater.

1.2.2 Parametric Roll Resonance

Parametric roll resonance is an auto-parametric resonance phenomenon whose onset causes a sudden rise in roll oscillations. It is a nonlinear parametric resonance phenomenon, and different from ”normal” resonance, which is characterized by external forces. Parametric resonance takes place in systems characterized by periodic variations of some of its coefficients.

[1]Matlab and Simulink are registered trademarks of The Mathworks, Inc., http://www.mathworks.com/.

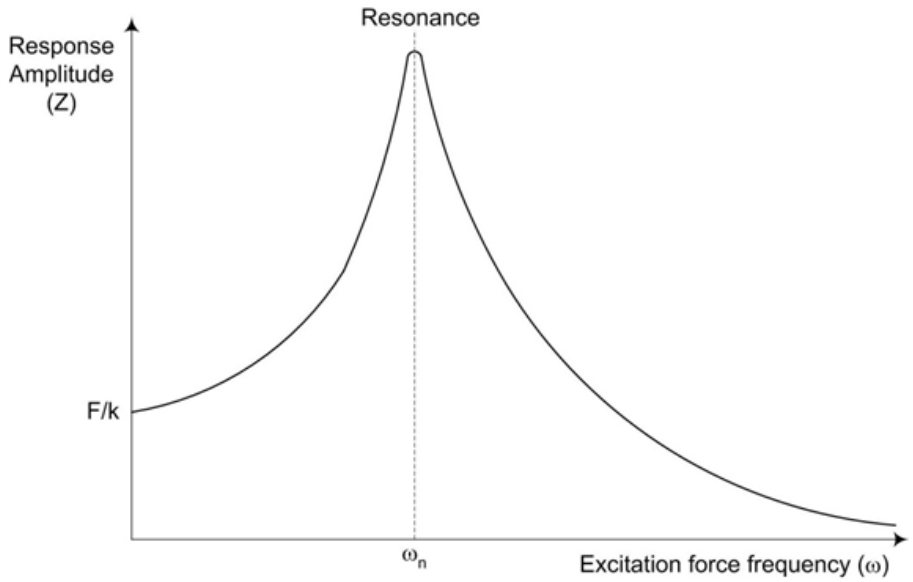

Figure 1.3: System motion amplitude versus the forcing function frequency. Image from `www.usna.edu`.

Parametric rolling, or low-cycle resonance, is caused by a Mathieu-type build up of a large roll motion. This large roll motion arises when a number of factors are combined, such as the natural roll frequency is a multiple of half the encounter frequency, the roll damping is low, and the wavelength is comparable to the ship length.

Parametric roll resonance in ships is a hot research topic nowadays. The maritime society has known of this problem since the early fifties, but in the last decades there has been an increased interest in parametric roll resonance. This is partly due to the growth in container trade, and several incidents at different types of vessels.

Although this is a dangerous phenomenon, it does not happen to every ship at any time. Certain empirical criteria are required for the onset of parametric roll resonance, [29]:

- Encounter frequency approximately equal to twice the natural roll frequency.
- Wave length approximately equal to ship length.
- Sufficient wave height.
- Sufficiently low roll damping.

When ships sail in longitudinal waves, and the encounter frequency and wave length satisfy the conditions listed above, passage of wave crest and wave trough along the hull continuously amplifies the roll motion at half the frequency of encounter. This gives the onset of a resonance condition, and it affects several kinds of ships like fishing vessels, cruise ships and container ships, due to their specific hull shapes. Nowadays, container ships are characterized by large bow flare and stern overhang, which makes them especially prone to parametric roll resonance. It is a dangerous resonance phenomenon, and in worst case scenarios parametric roll resonance can cause roll angles of 30-40 degrees. When it happens, the roll angle can reach up to 40°, at the most extreme, in just a few roll periods. Large roll oscillations may also possibly lead to capsizing.

There has been done allot of work in mathematical modelling for this phenomenon. A 3 DoF nonlinear model accounting for heave, pitch and roll has been proposed by Neves and Rodríguez, [39]. But models with lower complexity, single DoF models, have been widely used and tested for uncoupled roll motion to derive stability conditions. Different articles and books discuss ways to combat, cope with and prevent parametric resonance, including passive design measures and active control methods.

The physical nature of parametric roll has been known for many years. Theoretically, the resonance behaviour is well understood and it can be reproduced by quasi-periodic changes in parameters of nonlinear differential equations that describe ship motion. Practically, the challenge is whether detection and stabilization can be achieved in time to avoid damage.

1.2.3 Detection, Avoidance and Control

There have been made different approaches to try to control parametric roll, both through direct and indirect control methods. Indirect control attempts to control the phenomenon by violating the empirical conditions necessary for the onset of parametric roll to develop. Indirect control also includes frequency detuning control; this controller has been published in, [29]. Changing the ships heading or speed is most likely the easiest attempt to control the problem. Another way to try to avoid parametric roll resonance will be through adjustments to the hull form in the bow and stern areas, but it is thought that concentration on operational factors might be more productive.

The direct methods are aimed at directly controlling the roll motion by generating an opposing roll moment. Among these controllers are u-tanks, fins and gyro

stabilizers, [27]. Direct methods include active u-tank control; this controller has been published in [28]. A combination of direct and indirect methods is also possible.

Parametric resonance starts to evolve long before it is visually noticeable. It can build up in just a few time cycles. An automatic detection system is needed to detect parametric roll resonance sufficiently early enough, so that precautions can be made. Det Norske Veritas have tried to assist watch keepers by producing an on board wave radar and motion sensor based system which continuously analyses the wave formations being encountered, [2]. Nowadays ships use online detection schemes, like that, to predict parametric roll resonance.

1.2.4 Observability and Estimation

Basically we want to stabilize the roll of the ship, and in turn be able to control the vessel in such a way that parametric roll resonance does not occur.

A problem that often arises in control techniques is that the internal states of the system cannot be directly observed. Most nonlinear design techniques assume state feedback; that is, measurements of all state variables are available. In many practical problems we cannot measure all state variables. Then state feedback is not possible. One form of dynamic compensation is to use observers that asymptotically estimate the state from output measurements. The observer will attempt to determine an estimate of the states through mathematical relations. These estimates can then in turn be fed to the controller. Figure 1.4, depicts a proposed approach for controlling the vessel. The estimated states, $\hat{x}$, are to be sent as inputs to the controller, which in turn will control the ship.

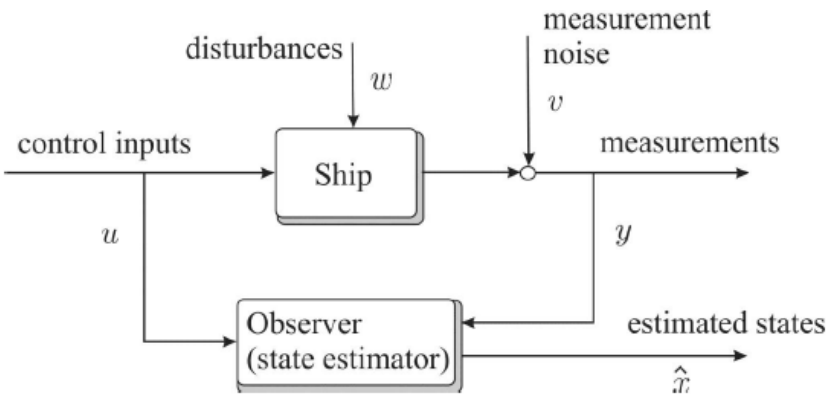

Figure 1.4: State Estimator

This thesis revolves around making nonlinear state estimators for a ship in parametric resonance, the bottom part of the block diagram in Figure 1.4. The first step in achieving this objective is to derive the nonlinear equations of motion for roll angle, roll rate, and surge speed. The resulting model will be a nonlinear state-space model with a measurement equation, containing all relevant linear and nonlinear terms. Then we derive conditions for nonlinear observability. Moreover different observers, for estimation of the frequency of encounter, and the associated sea state parameter for a ship exhibiting parametric roll, are to be investigated. This includes estimation of the wave spectrum frequency and the direction of the incoming waves.

Through this work, we attempt to implement an extended Kalman filter (EKF) for estimation of the frequency of encounter, and the wave spectrum modal frequency. We proceed to implement an unscented Kalman filter (UKF), and compare the results of the different observers. We also investigate if it is possible to estimate the direction of the incoming waves.

1.3 Main Contribution

Parametric roll is a critical phenomenon for ships, which may lead to dangerous situations. The susceptibility of container ships to parametric roll resonance has caused considerable research activities in the development of control to avoid large roll motions. A major obstacle in the actual implementation of feedback control laws is a lack of information about the current state of the system. A common strategy is to implement the control law using a state estimate $\hat{x}$ instead of the actual state. $\hat{x}$ is often designed as the output of an observer. The main goal of this thesis has been to develop nonlinear observers for a ship exhibiting parametric roll. The main contributions of this thesis are considered to be:

- Conducting an observability analysis of the model.
- Achieving estimation of the frequency of encounter, through nonlinear observers.
- Estimation of the direction of the incoming waves.

1.4 Thesis Organization

The rest of this thesis is organized as follows:

Part II This part contains the simulator model for generation of measurements.

Chapter 2 This chapter presents the mathematical model. The standard model for parametric roll is expressed in state-space form, and wave-induced forces are added. The augmented state-space model is written down, including the measurement equation.

Chapter 3 In this chapter the conditions for nonlinear observability are derived. Local observability is concluded. The results of this chapter is used as basis condition for the observers, in the thesis.

Chapter 4 In this chapter we propose speed and heading controllers.

Chapter 5 In this chapter we implement a Matlab/Simulink model of the system to check the reliability of the system equations.

Part III This part contains the section for observer design of this thesis.

Chapter 6 This chapter gives a brief introduction to Kalman filtering, and describes different filtering techniques. The algorithms for the state estimators are given. It contains no novel results.

Chapter 7 This chapter presents how the different Kalman filters are implemented in Matlab, the plots of the results are presented, and the encounter frequency plot and value is given.

Chapter 8 Here we compare the results from the observers.

Part IV This part contains the section for wave direction estimation.

Chapter 9 Here we investigate estimation of the direction of the incoming waves. We make an augmented state-space model, implement an observer for wave direction estimation, and do a simulation study.

Part V This part contains the closing remarks of this thesis.

Chapter 10 Here we discuss the results in the report.

Chapter 11 This chapter presents the conclusions and lists avenues for future work.

Appendices Here are found some auxiliary results.

Appendix A This appendix contains a table of the main characteristics of the container ship.

Appendix B In this appendix the most common coordinate frames used in navigation are described.

Appendix C This appendix summarizes the basics of the JONSWAP wave spectra. It contains no novel contributions, and is presented here only as necessary background.

Appendix D This appendix contains calculation of the Lie Derivatives, up to an order of 2, for the observability check presented in Chapter 3. Moreover, codelines used to conduct the observability check in Maple is given. In addition, terms, for the observability matrix in Sec. 3.3.1, that are too long to include in the report, are given here.

Appendix E This appendix contains the 9×9 Jacobian matrix for the observer for wave direction estimation.

Appendix F This appendix contains the Matlab/Simulink model of the system.

Appendix G In this appendix, the M-files to generate the results are given.

References Lists the references used in this thesis.

Part II

Simulator Model for Generation of Measurement

Chapter 2

Modeling of Parametric Roll on Ships

In Chapter 1, an introduction to parametric rolling was made, and an overview of literature and previous work was clarified. In this chapter, the model of the ship, used for the rest of the thesis work, is explained.

2.1 Modeling of the System

We need a mathematical model to be able to derive observability conditions, and to design a state observer.

In recent years, several ship models of different complexity for parametric roll have been developed [12, 20, 29, 38, 39, 48, 55]. A prevalent, simple model to describe parametrically excited roll motion is the Mathieu equation:

$$m_{44}\ddot{\phi} + d_{44}\dot{\phi} + [k_{44} + k_{\phi t}\cos(\omega_e t + \alpha_\phi)]\phi = 0 \tag{2.1}$$

Here, m_{44} is the sum of the moment of inertia and the added moment of inertia in roll, d_{44} the linear hydrodynamic damping coefficient, and k_{44} the linear restoring moment coefficient. The amplitude of the change in the linear restoring coefficient is $k_{\phi t}$, ω_e is the encounter frequency, and α_ϕ is a phase angle. All the parameters are considered constant.

To be able to derive hydrodynamic forces and moments on a ship, for a 6 DoF

model, certain assumptions are required, [9]:

A 2.1. *There is no current.*

A 2.2. *The hull can be split into triangular or quadrangular panels, where each panel can be parametrized as a two-dimensional surface embedded in* $\Re^3$.

A 2.3. *The frequency-dependent parameters of the damping, added mass, and Coriolis/centripetal matrices are constant.*

A 2.4. *The ocean is infinitely deep.*

A 2.5. *The pressure field in the ocean is unchanged by the passage of the ship (in effect, waves are traveling "through" the ships' hull, [17]).*

The 6 DoF model can be used for any sea state in any condition, [9]. To create an implementation suitable for parametric roll, we assume that the waves are planar and sinusoidal. The model can be simplified to the three most important degrees of freedom for ships in parametric roll resonance, heave, roll and pitch, [29, 39], assuming steady, planar waves, [9].

Parametric resonance occurs among others in auto-parametric systems. Auto-parametric systems consist of two subsystems, a primary and a secondary one. In the case of the motions of a ship we have a longitudinal, and a lateral subsystem. The primary system can be externally forced. It can be self-excited, parametrically excited, or a combination of both. The secondary system is coupled to the first one in a nonlinear way.

The model proposed by Neves and Rodríguez, [39], is used to describe parametric roll resonance as an auto-parametric system. In 3 DoF the model combines the primary system, heave and pitch, with the secondary system, roll, into one system. Heave and pitch are already coupled, and during parametric resonance these transfer energy to roll. There is coupling due to energy transfer and Doppler frequency shift. Wave motion excites the primary system, and the secondary one is parametrically excited by the primary system. So waves excites all modes. If the reader is interested in a thorough description, more detailed explanations are found in [29, 39].

For the study of parametric roll resonance, in this thesis, an uncoupled version of the equation from [39] is presented and investigated.

2.1.1 One DoF Standard Model for Parametric Roll

In the present a simplified model of a ship, for parametric rolling, is considered, where roll motion has been modeled as an uncoupled Mathieu type equation, Eq. (2.1). The sway and yaw motions are ignored as the vertical motions have the largest influence on the instantaneous GZ curve. The speed and the heading are treated as time-varying measured signals.

We make the following extra assumption to derive a 1 DoF model, [9]:

A 2.6. *The ship is travelling directly into the waves.*

This means the analysis is restricted to the head sea condition.

With these simplifications, and under the stated assumptions (A.2.1 - A.2.6) the one degree of freedom roll motion equation in head seas can be derived from models presented in [16, 29]. It is given by:

$$(I_x + A_{44})\ddot{\phi} - B_{44}\dot{\phi} + C_{44}(\overline{GM}_m + \overline{GM}_a \cos\left(\int_0^t \omega_e(\tau)d\tau\right))\phi - K_{\phi^3}\phi^3 = 0 \quad (2.2)$$

$$C_{44} = \rho g \nabla \quad (2.3)$$

where I_x is the rigid-body inertia, A_{44} is the added mass, B_{44} is a damping coefficient, C_{44} is the restoring moment coefficient with water density ρ, gravitational acceleration g, and displaced water volume ∇, $\overline{GM}_m$ is the mean value of metacentric height (MCH), $\overline{GM}_a$ is the amplitude of MCH change, ω_e is the encounter frequency, K_{ϕ^3} is the restoring force, and $K_{\phi^3}\phi^3$ is a cubic restoring term. This is a Mathieu type equation with linear damping and non-linear restoring terms. In regular waves, the Mathieu equation can explain the onset of heavy roll motion in head seas. It is clear that this model is very simplistic, but it is well suited to illustrate the procedure as it can model parametric rolling and resonance excitation.

For a vessel sailing in head seas, the inertial forces due to mass and added mass, the drag forces due to wave resistance and the thrust supplied by the propeller are forces acting along the longitudinal direction. The non-linear surge dynamics can also be derived from [29], and [16], and is given by:

$$(m + A_{11})\dot{u} + B_{11}|u|u = \tau_1 \quad (2.4)$$

where

$$B_{11} = \frac{1}{2}\rho S(1+k)\frac{0.075}{(\log_{10} R_n - 2)^2}, \quad R_n = \frac{uL_{pp}}{\nu} \tag{2.5}$$

where m is the mass of the ship, A_{11} is the added mass term, $B_{11}|u|u$ is the ships non-linear hydrodynamic resistance, u is the ship forward speed, τ_1 is the thrust generated by the propeller, ρ is the water density, S is the wetted surface, k is a factor, R_n is the Reynolds number, L_{pp} is the ship length and ν is the kinematic viscosity.

For a ship moving at forward speed $u > 0$, there will be a shift in the wave spectrum peak frequency ω_0. The shifted frequency is referred to as the frequency of encounter ω_e, and it depends on ship speed u, modal wave frequency ω_0 and wave direction β_w. The speed $u(t)$ and the heading $\psi(t) = \beta_w^n - \beta_w(t)$ are treated as time-varying measured signals. Note that for head seas $\beta_w^n = \pi$. The heading angle of the ship is depicted in Figure 2.1. The equation for the encounter frequency can be written as, [8],

$$\omega_e = \omega_0 - \frac{\omega_0^2}{g}u(t)\cos(\beta_w) \tag{2.6}$$

where ω_0 is the frequency of the waves, g is the gravitational acceleration and u is the forward speed of the ship. The encounter frequency is defined as the frequency of the waves as seen from the ship.

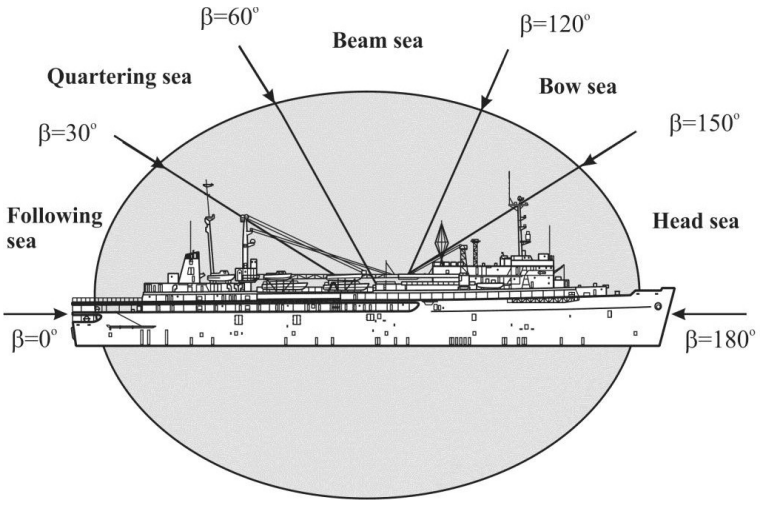

Figure 2.1: Heading Angle of a Ship, [17].

When the encounter angle, β, is 180°, that is the ship is sailing in head seas, the wave encounter frequency is given by,

$$\omega_e = \omega_0 + \frac{\omega_0^2}{g} u \tag{2.7}$$

where ω_0 is modal wave frequency. For the simulations in this project the modal frequency of the waves is assumed to be constant, or slightly time-varying.

The dynamics of the encounter frequency, and modal wave frequency, can be described by,

$$\dot{\omega_e} = \frac{\omega_0^2}{g} \dot{u} \tag{2.8}$$

$$\dot{\omega_0} = 0 \tag{2.9}$$

2.1.2 The Model in State-Space Form

We can choose the states of the system as

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} \phi \\ \dot{\phi} \\ u \\ \omega_0 \\ \cos\left(\int_0^t \omega_e(\tau) d\tau\right) \\ \frac{d}{dt} \cos\left(\int_0^t \omega_e(\tau) d\tau\right) \end{bmatrix} \tag{2.10}$$

Then, the ship model can be rewritten as follows, in a system matrix $f(x)$,

$$\dot{x} = f(x) = \begin{bmatrix} x_2 \\ \frac{B_{44}}{I_x + A_{44}} x_2 - \frac{C_{44}}{I_x + A_{44}} (\overline{GM}_m + \overline{GM}_a x_5) x_1 + \frac{K_{\phi^3}}{I_x + A_{44}} x_1^3 \\ \frac{\tau_1}{m + A_{11}} - \frac{B_{11}}{m + A_{11}} |x_3| x_3 \\ 0 \\ x_6 \\ -x_5 (x_4 + \frac{x_4^2}{g} x_3)^2 + \frac{x_4 x_6}{x_3 x_4 + g} (\frac{\tau_1}{m + A_{11}} - \frac{B_{11}}{m + A_{11}} |x_3| x_3) \end{bmatrix} \tag{2.11}$$

The first two equations in Eq. (2.11) are the roll motion, the third equation represents the speed, the fourth one is the modal frequency of the waves, whilst the fifth and sixth equation is an oscillator.

2.2 Wave Induced Forces

We have to include the effects of the wave characteristics onto the speed dynamics. The waves are hitting the ship from the front, and thus cause an oscillation about the nominal speed. This can be implemented in the system by modelling the wave spectrum as a linear second order system.

The transfer function is,

$$h_{\text{wave}}(s) = \frac{K_w s}{s^2 + 2\lambda\omega_0 s + \omega_0^2} \tag{2.12}$$

with

$$K_w = 2\lambda\omega_0\sigma \tag{2.13}$$

$h_{\text{wave}}(s)$ is a 2nd order wave-induced disturbance.

The wave disturbance model can be written in state space as,

$$\dot{x}_7 = x_8 \tag{2.14}$$
$$\dot{x}_8 = -\omega_0^2 x_7 - 2\lambda\omega_0 x_8 + K_w w \tag{2.15}$$
$$y = x_8 \tag{2.16}$$

The input w is white noise.

We can add the wave disturbance as an output disturbance to the speed equation. The wave disturbance is modeled as a second-order system driven by white noise. The second state of this wave model can then be added to the measurement of the speed.

The wave response model can be modeled, by a linear approximation to a JONSWAP spectrum, [17], see Appendix C, with the numerical values of Table 2.1.

We add the wave-induced forces to the nonlinear state-space model from Eq.(2.11).

Quantity	Symbol	Value	
Relative damping factor	λ	0.1018	m
σ_{gain}	σ	1.2871	m
Resonance frequency	ω_0	0.4684	m

Table 2.1: Parameters in the wave response model

The augmented state-space model is given as,

$$\dot{x} = f(x) = \begin{bmatrix} x_2 \\ \frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi^3}}{I_x+A_{44}}x_1^3 \\ \frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}|x_3|x_3 \\ 0 \\ x_6 \\ -x_5(x_4 + \frac{x_3 x_4^2}{g})^2 + \frac{x_4 x_6}{x_3 x_4 + g}(\frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}|x_3|x_3) \\ x_8 \\ -\omega_0^2 x_7 - 2\lambda\omega_0 x_8 + K_w w \end{bmatrix} \tag{2.17}$$

We assume that the forward speed, roll angle and roll rate are measured.

$$z = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^\top \tag{2.18}$$

Then we add wave-induced forces, and the measurement equation becomes,

$$z = \begin{bmatrix} x_1 & x_2 & x_3 + x_8 \end{bmatrix}^\top \tag{2.19}$$

2.3 Model Parameters

The ship parameters are identified from the hull shape of a container vessel, and upon wave characteristics, [29]. The values of the parameters are obtained by experiments on a 1:45 container ship model in a towing tank. The main characteristics of this ship are to be found in Table A.1. Data was achieved by varying ship forward velocity, wave frequency and wave height.

The numerical values obtained by these experiments are calculated to full scale ships, and listed in Table 2.2. Thus the data achieved is valid for full scale vessels.

Quantity	Symbol	Value
Rigid-body inertia	I_x	1.4014×10^{10} kg
Added mass	A_{44}	2.17×10^{9} kg
Linear damping	B_{44}	-3.20×10^{8} $\frac{\text{kgm}^2}{\text{s}}$
Water displacement	∇	76468 m^3
Mean value of MCH	$\overline{GM}_m$	1.91 m
Amplitude of MCH change	$\overline{GM}_a$	0.84 m
Modal wave frequency	ω_0	0.4764 $\frac{\text{rad}}{\text{s}}$
Restoring force	K_{ϕ^3}	-2.974×10^{9} $\frac{\text{kgm}^2}{\text{s}^2}$
Mass of the ship	m	7.6654×10^{7} kg
Added mass	A_{11}	7.746×10^{6} kg
Water density	ρ	1025 $\frac{\text{kg}}{\text{m}^3}$
Wetted surface	S	6600 m^2
Gravitational acceleration	g	9.81 $\frac{\text{m}}{\text{s}^2}$
Ship length	L_{pp}	281 m
Kinematic viscosity	ν	1.519×10^{-6} $\frac{\text{m}}{\text{s}^2}$
Form factor	k	0.1

Table 2.2: Ship Model Parameters

Chapter 3

Observability

3.1 Observability of a System

The concept of observability is regarding whether the state information can be extracted from the observations of the inputs and outputs. This is an important issue since only the inputs and outputs are measurable in many practical situations. A system is said to be observable if, for any possible sequence of state and control vectors, the current state can be determined in finite time using only the outputs. If we manage to get the equations in a state space representation we can check the observability of the system with a convenient test for a linear time-invariant system. If the observability matrix has full row rank, the system is observable. If n rows are linearly independent, then each of the n states is viewable through linear combinations of the output variables y.

Definition 3.1 (Observability of linear time-invariant systems)
A linear, time-invariant system with system- and measurement matrix (A, C) is observable if the $(n \times n)$ observability matrix, [4]:

$$\mathcal{O} = [C^\top | A^\top C^\top | \dots | (A^\top)^{n-1} C^\top] \tag{3.1}$$

is of full rank.

A necessary and sufficient condition for $\mathcal{O}$ to have full rank is:

$$det\mathcal{O} > 0 \tag{3.2}$$

Definition 3.2 (Observability of linear time-varying systems)
A linear, time-varying system with system- and measurement matrix $(A(t), C(t))$ is observable if $\exists T > 0$, and $\beta > \alpha \geq 0$ such that, [15]:

$$\alpha I \leq \frac{1}{T}\int_{t_0}^{t_0+T} exp(A^\top(\tau)\tau)C^\top(\tau)C(\tau)exp(A(\tau)\tau)d\tau \leq \beta I \tag{3.3}$$

$\forall t_0 \in R_+$. This means that the integral of matrix $exp(A^\top\tau)C^\top Cexp(A\tau)$ is uniform, positive definite over every interval of length T.

These definitions does not carry over to nonlinear systems.

3.1.1 Observability of Nonlinear Systems

Because of states x_5 and x_6, namely $\cos\left(\int_0^t \omega_e(\tau)\,d\tau\right)$ and $\frac{d}{dt}\cos\left(\int_0^t \omega_e(\tau)\,d\tau\right)$, in our model, Eq. (2.10), the system cannot be linearized.

As a consequence of this we cannot check the observability as we would do for a linearized system, as in definition 3.1, and 3.2. But there are tools to check the observability for nonlinear systems, namely the observation space. Nonlinear observability is intimately tied to the Lie derivative. The Lie derivative is the derivative of a scalar function along a vector field, and we can use them to calculate the observation space.

3.1.2 Lie Derivatives

Consider the SISO system

$$\dot{x} = f(x) + g(x)u \tag{3.4}$$
$$y = h(x) \tag{3.5}$$

Where f, g and h are sufficiently smooth in a domain $D \subset R^n$. The mappings $f : D \to R^n$ and $g : D \to R^n$ are called vector fields on D. The derivative, $\dot{y}$, is given by

$$\dot{y} = \frac{\partial h}{\partial x}[f(x) + g(x)u] \equiv L_f h(x) + L_g h(x)u \tag{3.6}$$

where

$$L_f h(x) = \frac{\partial h}{\partial x} \tag{3.7}$$

is called the Lie Derivative of h with respect to f, or along f. This is the familiar notion of the derivative of h along the trajectories of the system $\dot{x} = f(x)$.

Following notation can be used, [32]:

$$L_g L_f h(x) = \frac{\partial L_f h}{\partial x} g(x) \tag{3.8}$$

$$L_f^2 h(x) = L_f L_f h(x) = \frac{\partial L_f h}{\partial x} f(x) \tag{3.9}$$

$$L_f^k h(x) = L_f L_f^{k-1} h(x) = \frac{\partial L_f^{k-1} h}{\partial x} f(x) \tag{3.10}$$

$$L_f^0 h(x) = h(x) \tag{3.11}$$

3.1.3 Local Observability of Nonlinear Systems

Consider the system

$$\dot{x} = f(x, u) \tag{3.12}$$

$$z = h(x) = [h_1(x), h_2(x), \ldots, h_p(x)]^\top \tag{3.13}$$

Two states x_0 and x_1 are **distinguishable** if there exists an input function u^* such that

$$z(x_0) \neq z(x_1) \tag{3.14}$$

The system is **locally observable**, that is **distinguishable at a point** x_0, if there exists a neighbourhood of x_0 such that in this neighbourhood, [23],

$$x_0 \neq x_1 \Rightarrow z(x_0) \neq z(x_1) \tag{3.15}$$

A test for local observability, from [23], is that:

$$\mathcal{O}(x_0, u^*) \equiv \left. \frac{\partial l(x_0, u^*)}{\partial x} \right|_{x=x_0} \tag{3.16}$$

must have rank n, where n is the rank of x and

$$l(x_0, u^*) \equiv \begin{bmatrix} L_f^0(h_1) \\ \vdots \\ L_f^{n-1}(h_p) \end{bmatrix} \tag{3.17}$$

3.2 Observability of Our Model

Our model for the system is given in Eq. (2.17).

For this system the $l(x_0, u^*)$ vector becomes

$$l(x_0, u^*) = [L_f^0(h_1), L_f^0(h_2), L_f^0(h_3), L_f^1(h_1), L_f^1(h_2) \dots L_f^{n-1}(h_n)]^\top \tag{3.18}$$

We calculate all the terms up to an order of 2, see Appendix D.

Then we insert the terms calculated into the $l(x_0, u^*)$ vector.

$$\begin{aligned}
&l(x_0, u^*) \\
&\quad = \left[L_f^0(h_1), L_f^0(h_2), L_f^0(h_3), L_f^1(h_1), L_f^1(h_2), L_f^1(h_3), L_f^2(h_1), L_f^2(h_2), L_f^2(h_3)\right]^\top \\
&\quad = \begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_2 \\
\frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi^3}}{I_x+A_{44}}x_1^3 \\
\frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}|x_3|x_3 \\
\frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi^3}}{I_x+A_{44}}x_1^3 \\
-\frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{3K_{\phi^3}}{I_x+A_{44}}x_1^3 + \frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}\overline{GM}_a x_1}{I-x+A_{44}}x_5 \\
-\frac{B_{11}}{m+A_{11}}(x_3 sign(x_3) + |x_3|)\left(\frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}x_3|x_3|\right)
\end{bmatrix}
\end{aligned} \tag{3.19}$$

Then we proceed to calculate the $\mathcal{O}$ matrix, with the gradient of the l vector.

$$\begin{aligned}
&\mathcal{O}(x_0, u^*) \equiv \left.\frac{\partial l(x_0, u^*)}{\partial x}\right|_{x=x_0} \\
&= \left[\frac{\partial L_f^0(h_1)}{\partial x} \quad \frac{\partial L_f^0(h_2)}{\partial x} \quad \frac{\partial L_f^0(h_3)}{\partial x} \quad \frac{\partial L_f^1(h_1)}{\partial x} \quad \frac{\partial L_f^1(h_2)}{\partial x} \quad \frac{\partial L_f^1(h_3)}{\partial x} \quad \frac{\partial L_f^2(h_1)}{\partial x} \quad \frac{\partial L_f^2(h_2)}{\partial x} \quad \frac{\partial L_f^2(h_3)}{\partial x}\right]^\top
\end{aligned}$$

The calculations are to be found in Appendix D.

These gradient terms are to be inserted into the matrix from Eq. (3.16).

$\mathcal{O}$ must have rank n, equal to the order of our system, for the system to be observable. For this case that equals $n = 8$.

Clearly, the $\mathcal{O}$ matrix we gain by the calculations above is not of full rank. For instance values for the fourth state are always equal to zero. A worst case scenario would be if the roll angle, $x_1 = \phi$, equals zero. For column 6 we have the term $-\frac{C_{44}\overline{GM}_a x_1}{I_x + A_{44}}$. This will cause a problem for the observability of the system when the roll angle equals zero, $x_1 = \phi = 0$. For column 5 we have the term $-\frac{C_{44}\overline{GM}_a x_2}{I_x + A_{44}} - \frac{B_{44}C_{44}\overline{GM}_a x_1}{(I_x + A_{44})^2}$. This is better because the states x_1 and x_2 will not be zero at the same instances of time. Since the roll angle is a sinusoid it is obvious that it, and its derivative, the roll rate, does not cross the zero-axis simultaneously. While calculating the Lie derivatives we also notice that Lie derivatives for the output x_2 are basically the same as for the output x_1, that is

$$L_f^1(h_1) = L_f^0(h_2) \tag{3.20}$$

$$L_f^2(h_1) = L_f^1(h_2) \tag{3.21}$$

and so on. They are interchangeably the same, and we will not gain much if we include the same Lie derivatives into the $\mathcal{O}$ matrix. Thus, to possibly obtain a $\mathcal{O}$ matrix of rank 8, we must continue to calculate higher order Lie derivatives. This has been done in Maple[1], Sec. 3.2.1.

3.2.1 Observability Check in Maple

First we add the libraries *with(linalg)*, and *with(Student[VectorCalculus])*, to enable linear algebra operations, and gradient calculation. Then we define the system matrix with the expression $f := [\,]$.

We calculate the Lie Derivatives up to an order of 8. The calculations are first done along the line of the first state, x_1, since $x_1 = 0$ will most likely be a worst case scenario.

Command lines used to calculate the Lie Derivatives, in Maple, are given in Appendix D.

The complexity of the Lie Derivatives becomes quite high as the order is increased. The terms we get from the calculations above can be inserted into the l vector, Eq. (3.18). We then proceed to calculate the $\mathcal{O}$ matrix, Eq. (3.20), from the gradients of the l vector, with the commands given in Appendix D.

[1]Maple is a registered trademark of MapleSoft, A Cybernet Group Company, http://www.maplesoft.com/

From the calculations made we will now have to make choices of which terms we want to insert into the $\mathcal{O}$ matrix, in order for it to have full rank. Due to readability, the terms we get for the Lie derivatives, and the possible terms for the $\mathcal{O}$ matrix are omitted in the report.

As we have seen before the columns for state 1, 2 and 3 are ok. With ones for state x_1, x_2 and x_3, respectively, these columns will not interfere with any of the other columns, and they will never be equal to zero.

We evaluate terms for column 4, 5 and 6, to see what will happen with the observability of the system if we have states of zero value.

The term for state 4 is evaluated with both roll angle and state $x_5 = \cos\left(\int_0^t \omega_e(\tau)\, d\tau\right)$ equal to zero with the Maple command $eval(Od_6^+, [x_1 = 0, x_5 = 0])[4]$. From this we get,

$$-\frac{3C_{44}GM_a\left(\frac{x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{x_3x_4+g}-\frac{x_4x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)x_3}{(x_3x_4+g)^2}\right)x_2}{I_x+A_{44}} \tag{3.22}$$

As we can see the term is still dependent on both state 2, 3, 4 and 6, and clearly not equal to zero. We will use this term for column 4 in the observability matrix.

By the command $eval(Od_4^+)[6]$, we get

$$-\frac{C_{44}GM_ax_1}{I_x+A_{44}} \tag{3.23}$$

From Eq. (3.23) we can see that the term for state 6, for the Lie Derivative of 3rd order depends only on x_1. We therefore want a Lie Derivative term of higher order for column 6, as the term above will be equal to zero if the roll angle is of zero value.

The command $eval(Od_5^+, [x_1 = 0])[6]$ gives us,

$$-\frac{2C_{44}GM_ax_2}{I_x+A_{44}} \tag{3.24}$$

From this evaluation for state 6, we can see that even if the roll angle is zero, we still have a term dependent on x_2, which is the rate of the roll angle. These terms will not be equal to zero at the same time, as described earlier. We can therefore use this term for column 6 in the $\mathcal{O}$ matrix.

For column 5 and 6, in the observability matrix, it seems, from the results in Maple, and evaluation done, that calculations of the Lie Derivatives up to an order of 4 are sufficient. From $Od5 := Gradient(L_{15}(1), [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])$, we pick the following terms for state 5 and 6 respectively

$$\begin{aligned}
&-\frac{B_{44}C_{44}GM_a x_2}{(I_x+A_{44})^2} - \frac{C_{44}GM_a\left(\frac{B_{44}x_2}{I_x+A_{44}} - \frac{C_{44}(GM_m+GM_a x_5)x_1}{I_x+A_{44}} + \frac{K_{\phi 3}x_1^3}{I_x+A_{44}}\right)}{I_x+A_{44}} \\
&-\frac{\left(-\frac{C_{44}(GM_m+GM_a x_5)}{I_x+A_{44}} + \frac{3K_{\phi 3}x_1^2}{I_x+A_{44}} + \frac{B_{44}^2}{(I_x+A_{44})^2}\right)C_{44}GM_a x_1}{I_x+A_{44}} \\
&+\frac{C_{44}GM_a x_1\left(x_4 + \frac{x_4^2 x_3}{g}\right)^2}{I_x+A_{44}}
\end{aligned} \tag{3.25}$$

and

$$-\frac{2C_{44}GM_a x_2}{I_x+A_{44}} + \frac{B_{44}C_{44}GM_a x_1}{(I_x+A_{44})^2} - \frac{C_{44}GM_a x_1 x_4\left(\frac{\tau}{m+A_{11}} - \frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{(I_x+A_{44})(x_3x_4+g)} \tag{3.26}$$

Since the waves are considered as output disturbance to the speed equation we will not get terms for it from calculating Lie Derivatives for roll. We therefore continue the observability check by calculating the Lie Derivatives of the 3rd output. Here we have $x_3 + x_8$, according to Eq. (2.19).

The calculations are done along the line of the third output, $x_3 + x_8$, by changing the command $h_1 := x_1$ to $h_1 := x_3 + x_8$.

We calculate Lie Derivatives in the same manner as for calculations along the line of the first state. Then we evaluate the terms for state 3, 7 and 8.

By command lines $eval(Od_2^+)[7]$, and $eval(Od_1^+)[8]$, in Maple, we get the terms $-\omega_0^2$ and 1, for state 7 and 8 respectively. $eval(Od_1^+)[3]$ gives 1 for state 3. All these terms are constant and does not depend on any of the other states. They will not interfere with any other states, and we will therefore be able to use them in the observability matrix.

With Lie Derivatives, chosen in order to obtain full rank, we get a matrix $\mathcal{O}$;

$$\mathcal{O}(x_0,u^*) \equiv \left.\frac{\partial l(x_0,u^*)}{\partial x}\right|_{x=x_0} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x[4] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x[5] & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x[6] & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.27}$$

where, $x[5]$ is given in Eq.(3.25), $x[6]$ is given in Eq.(3.26), and
$x[4] =$

$$\begin{aligned}
&- \frac{C_{44}GM_a\left(-2x_5\left(x_4+\frac{x_4^2x_3}{g}\right)\left(1+\frac{2x_4x_3}{g}\right)+\frac{x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{x_3x_4+g}-\frac{x_4x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)x_3}{(x_3x_4+g)^2}\right)x_2}{I_x+A_{44}} \\
&- \frac{C_{44}GM_ax_1}{I_x+A_{44}}\left(-\frac{2x_5\left(1+\frac{2x_4x_3}{g}\right)x_4^2}{g}-\frac{4x_5\left(x_4+\frac{x_4^2x_3}{g}\right)x_4}{g}-\frac{2x_4x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{(x_3x_4+g)^2}\right. \\
&+ \frac{2x_4^2x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)x_3}{(x_3x_4+g)^3}+\frac{x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3abs(1,x_3)}{m+A_{11}}\right)}{x_3x_4+g} \\
&\left.- \frac{x_4x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3abs(1,x_3)}{m+A_{11}}\right)x_3}{(x_3x_4+g)^2}\right)\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right) \\
&+ \frac{2C_{44}GM_ax_1\left(x_4+\frac{x_4^2x_3}{g}\right)\left(1+\frac{2x_4x_3}{g}\right)x_6}{I_x+A_{44}}+\left(-\frac{C_{44}GM_ax_1\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{(I_x+A_{44})(x_3x_4+g)}\right. \\
&\left.+ \frac{C_{44}GM_ax_1x_4\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)x_3}{(I_x+A_{44})(x_3x_4+g)^2}\right)\left(-x_5\left(x_4+\frac{x_4^2x_3}{g}\right)^2+\frac{x_4x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{x_3x_4+g}\right) \\
&+ \left(-\frac{2C_{44}GM_ax_2}{I_x+A_{44}}-\frac{B_{44}C_{44}GM_ax_1}{(I_x+A_{44})^2}-\frac{C_{44}GM_ax_1x_4\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{(I_x+A_{44})(x_3x_4+g)}\right) \\
&\left(-2x_5\left(x_4+\frac{x_4^2x_3}{g}\right)\left(1+\frac{2x_4x_3}{g}\right)+\frac{x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)}{x_3x_4+g}-\frac{x_4x_6\left(\frac{\tau}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A-11}\right)x_3}{(x_3x_4+g)^2}\right)
\end{aligned}$$

3.2.2 Conclusion Regarding Observability

The observability matrix we obtain, Eq. (3.27), is of rank 8, equal to the order of our system.

From calculations of Lie Derivatives, evaluation of terms calculated, and from theory for observability of nonlinear systems, we conclude that our system is locally observable.

3.3 Observability Analysis for Wave Direction Estimation

In Chapter 9, we make an augmented state-space model for estimation of the direction of the incoming waves. As a consequencw we need to perform a new observability analysis. This time on the system in Eq. (9.19)-(9.24). The observability property ensures that all the system states can be retrieved from the measured output, and if we want to be able to estimate the direction of the incoming waves this is a necessary condition.

We now consider a state-space representation of the following nonlinear system:

$$\dot{x} = f(x, u) \tag{3.28}$$
$$y = h(x) \tag{3.29}$$

where $x \in \Re^n$ is the state vector, $u = [u_1, \ldots, u_l]^\top \in \Re^l$ is the vector of control inputs, and $y = [y_1, \ldots, y_m]^\top \in \Re^m$ is the measurement vector, with $y_k = h_k(x), k = 1 \ldots m$.

In Sec. 3.1.3 we defined the observability matrix for nonlinear systems, namely Eq. (3.16).

The important role of this matrix in the observability analysis of a nonlinear system is captured by the following proposition, [24]:

Proposition 3.1 (Observability Rank Condition).
If the observability matrix, Eq. (3.16), of the nonlinear system defined in Eq. (3.28) is full rank, then the system is locally weakly observable.

In general, there exists no systematic method for selecting the suitable Lie derivatives and corresponding rows of $\mathcal{O}$ when examining the observability of a system. Instead, we will calculate the possible terms for the $\mathcal{O}$ matrix in Maple, See Sec. 3.3.1. The selection will thereby be performed by sequentially considering the directions of the state space along which the gradient of each of the candidate Lie derivatives provides information.

3.3.1 Observability Check in Maple for Wave Direction Estimation

As in Sec. 3.2.1, we first add the libraries *with(linalg)*, and *with(Student[VectorCalculus])*, to enable linear algebra operations, and gradient calculation. Then we define the system matrix with the expression $f := Matrix(9, 1, [\,])$.

We then proceed to calculate the Lie Derivatives. The calculations are first done along the line of the first state, x_1, and then we continue to calculate Lie Derivatives of the other outputs, x_2 and $x_3 + x_8$.

Command lines used to calculate the Lie Derivatives, for the first output, are as follows

$$h_1 := x_1 \tag{3.30}$$
$$L_{11} := h_1 \tag{3.31}$$
$$L_{12} := Gradient(L_{11}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9])^{+}.f \tag{3.32}$$
$$\vdots \tag{3.33}$$
$$L_{1i} := Gradient(L_{1(i-1)}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9])^{+}.f \tag{3.34}$$

The terms we get from the calculations above can be inserted into the l vector, as before. We calculate the $\mathcal{O}$ matrix, from the gradients of the l vector, with the command given below,

$$Od_i := Gradient(L_{1i}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9]) \tag{3.35}$$

From the calculations made we will now have to make choices of which terms we want to insert into the $\mathcal{O}$ matrix, in order for it to have full rank. Due to readability, the terms we get for the Lie derivatives, and the possible terms for the $\mathcal{O}$ matrix are omitted in the report.

In order to prove that the system is locally weakly observable, it suffices to show that the observability matrix, is full rank. We evaluate terms for column 1 to 9, to see what will happen with the observability of the system if we have states of zero value.

The term for state 4 is calculated in Maple with
$Od_5 := Gradient(L_{15}(1), [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9])$. We evaluate it with both roll angle, encounter angle, inputs u and state x_5 equal to zero with the Maple command $eval(Od_5^+, [x_1 = 0, x_5 = 0, x_9 = 0, u = 0])[4]$, and get

$$-\frac{3C_{44}GM_a\left(\frac{x_6B_{11}x_3|x_3|}{(m+A_{11})(g-x_3x_4)}+\frac{x_4x_6B_{11}x_3^2|x_3|}{(m+A_{11})(g-x_3x_4)^2}\right)x_2}{I_x+A_{44}} \tag{3.36}$$

As we can see the term is still dependent on both state 2, 3, 4 and 6, and clearly not equal to zero. We will use this term for column 4 in the observability matrix.

The term for state 9 is calculated in Maple with
$Od_5 := Gradient(L_{15}(1), [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9])$. We evaluate it with both roll angle, inputs u and state $x_5 = \cos\left(\int_0^t \omega_e(\tau)\,d\tau\right)$ equal to zero with the Maple command $eval(Od_5^+, [x_1 = 0, x_5 = 0, u = 0])[9]$, and get

$$-\frac{3C_{44}GM_a\left(-\frac{x_4x_6B_{11}x_3|x_3|\sin(x_9)}{(m+A_{11})(g-x_3x_4\cos(x_9))}-\frac{x_4^2x_6B_{11}x_3^2|x_3|\cos(x_9)\sin(x_9)}{(m+A_{11})(g-x_3x_4\cos(x_9))^2}\right)x_2}{I_x+A_{44}} \tag{3.37}$$

As we can see the term is non-zero and dependent on state 2, 3, 4, 6 and 9. As a result we will use this term for column 9 in the observability matrix.

With Lie Derivatives, chosen in order to obtain full rank, we get a matrix $\mathcal{O}$;

$$\mathcal{O}(x_0,u^*) \equiv \left.\frac{\partial l(x_0,u^*)}{\partial x}\right|_{x=x_0} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x[4] & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x[5] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x[6] & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_0^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x[9] \end{bmatrix} \tag{3.38}$$

where the term for $x[4]$ and $x[9]$ is given in Appendix D,

$$x[5] = -\frac{C_{44}GM_a x_2}{I_x + A_{44}} - \frac{B_{44}C_{44}GM_a x_1}{(I_x + A_{44})^2} \tag{3.39}$$

and

$$\begin{aligned} x[6] = &- \frac{2C_{44}GM_a x_2}{I_x + A_{44}} - \frac{B_{44}C_{44}GM_a x_1}{(I_x + A_{44})^2} \dots \\ &- \frac{C_{44}GM_a x_1 \left(-\frac{x_4\left(\frac{u_1}{m+A_{11}} - \frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(x_9 - u_2)}{g - x_3 x_4 \cos(x_9 - u_2)} + \frac{\cos(x_1)x_3 x_4 u_3 \sin(x_9 - u_2)}{g - x_3 x_4 \cos(x_9 - u_2)} \right)}{I_x + A_{44}} \end{aligned} \tag{3.40}$$

3.3.2 Conclusion Regarding Observability for Wave Direction Estimation

The observability matrix we get, Eq. (3.38), is of rank 9, equal to the order of our system.

From calculations of Lie Derivatives, evaluation of terms calculated, and from theory for observability of nonlinear systems, we conclude that our system is locally weakly observable.

Chapter 4

Speed and Heading Control Systems

In this chapter we propose a feedback linearization controller for the input, τ_1, to the system. Furthermore, a controller for the heading subsystem is designed.

4.1 Speed Controller

Feedback linearization is a common approach used in controlling nonlinear systems, [32]. Through a change of variables, and with a suitable control input, we can transform the nonlinear system into an equivalent linear system.

In feedback linearization control design, [25], we decompose the control signal u into two components with distinct functions:

$$u = u_{nl} + v_c \tag{4.1}$$

where

1. u_{nl} is used to "cancel" the process' nonlinearities
2. v_c is used to control the resulting linear system

From Eq. (2.17) we have

$$\dot{x_3} = \dot{u} = \frac{\tau_1 - B_{11}|x_3|x_3}{m + A_{11}} \tag{4.2}$$

Inspection of the state equation shows that we can choose τ_1 as

$$\tau_1 = B_{11}|x_3|x_3 + v_c(m + A_{11}) \tag{4.3}$$

to cancel the nonlinear term $B_{11}x_3|x_3|$. This cancellation results in the linear system

$$\dot{u} = (m + A_{11})^{-1}(B_{11}|x_3|x_3 - B_{11}|x_3|x_3 + v_c(m + A_{11})) = v_c \tag{4.4}$$

We can use linear methods to design a stabilizing linear state feedback control, v_c, that stabilizes the closed-loop system.

For our system we choose v_c as a P-controller of the form,

$$v_c = k_p(u - u_c), \quad k_p = k_p^\top < 0 \tag{4.5}$$

and obtain

$$\tau_1 = B_{11}|x_3|x_3 + k_p(u - u_c)(m + A_{11}) \tag{4.6}$$

4.2 Heading Controller

In Chapter 9, we add more states to our system, and end up with a new augmented state-space model, Eq. (9.19)-(9.24). As a consequence we get a new control input, namely the rudder angle, δ. We have to design a controller to give us the required rudder angle to follow the desired trajectory of the heading angle.

In order to track the reference trajectory, a controller for the heading subsystem is designed. The heading dynamics are represented by Eq. (**??**). The rudder deflection is chosen to be

$$\delta = K_p(\psi - \psi_d) - K_d r \tag{4.7}$$

where $K_p > 0$ and $K_d > 0$ are the proportional and derivative gains, respectively. ψ_d is the desired heading angle.

The controller gains can be computed using steady-state solutions, [17],

$$K_p = \sqrt{\frac{1}{\lambda_2}} \tag{4.8}$$

$$K_d = \frac{L}{U}\frac{\sqrt{1 + 2K_pK'T' + K'^2(U/L)^2(\lambda_1/\lambda_2)} - 1}{K'} \tag{4.9}$$

K' and T' have to be known with sufficient accuracy for accurate steering. In our controller we will be using the numerical values for K and T from Table 9.1.

Numerical values for the weighting factors, for a tanker and a cargo ship, are given by [53, 54], as:

Tanker:	$L_{pp} = 300m,$	$\lambda_1 = 15000,$	$\lambda_2 = 8.0$
Cargo ship:	$L_{pp} = 200m,$	$\lambda_1 = 1600,$	$\lambda_2 = 6.0$

Chapter 5

Simulink Model of the System

In this chapter we implement a Matlab/Simulink model for the simulation of parametric roll resonance, for a container ship. The model is given in Appendix F.

This is done to test for the reliability of our ship model. We need to check if our model for parametric roll in fact do resonate, with the system equations we have, and the parameters we use.

We make use of the feedback linearization controller proposed in Chapter 4.

The model parameters we use are obtained by experiments on a 1:45 container ship model in a towing tank, [29]. The values of the parameters are given in Table 2.2.

Figure 5.1, shows the roll angle we get from running the simulation. It is noticeable that the ship is originally in parametric roll resonance condition.

The curve in Figure 5.1 shows that the ship is experiencing severe roll oscillation. After about 500 seconds the roll amplitude settles to oscillations of about 25 degrees.

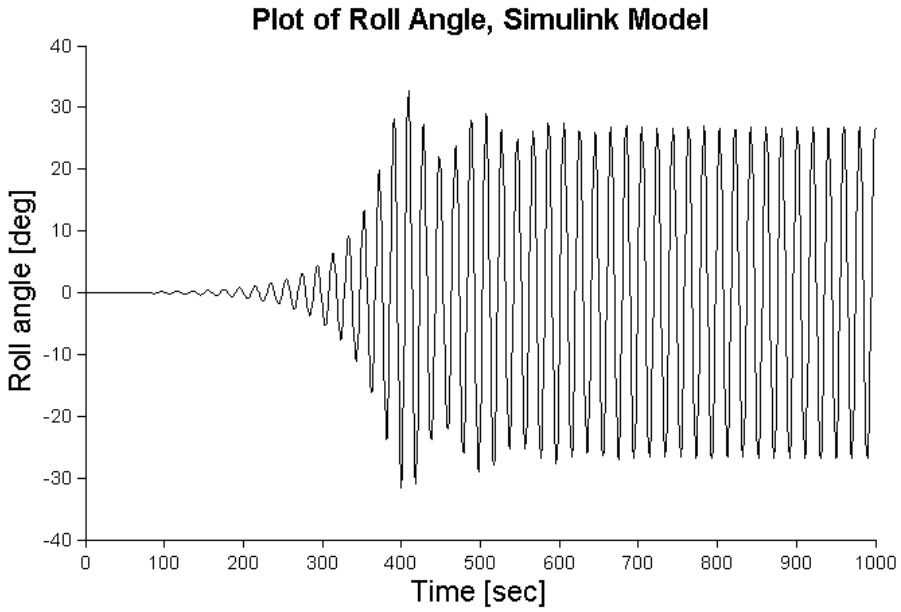

Figure 5.1: The roll amplitude of the system simulated in Matlab/Simulink.

Part III

Observer Design

Chapter 6

Kalman Filtering

> "... the Kalman filter represents the most widely applied and demonstrably useful result to emerge from the state variable approach of *modern control theory*".
>
> Harold W. Sorenson
> *Kalman Filtering: Theory and Application*,
> IEEE Press, 1985

This chapter gives an introduction to Kalman filtering, and describes different observer techniques. The content of the Kalman filter theory presented here is largely gathered from [10, 17].

6.1 Definitions

Random Process

A random variable X is a mapping between the sample space and the real numbers. A random process (RP) is a mapping from the sample space into an ensemble of time functions (known as sample functions).

Mean, Moment, Variance, Covariance of Random Process

Let $f(t,x)$ be the probability density function (p.d.f.) associated with a random process $X(t)$. If the p.d.f. is independent of time t, i.e., $f(t,x) = f(x)$, then the corresponding RP is said to be stationary. For this type of random processes, we define:

Mean (or **expectation**):

$$\mathrm{m} = E[X] = \int_{-\infty}^{\infty} x\mathrm{f}(x)\mathrm{d}x \tag{6.1}$$

Moment (j.th order moment):

$$E[X^j] = \int_{-\infty}^{\infty} x^j\mathrm{f}(x)\mathrm{d}x \tag{6.2}$$

Variance:

$$\sigma^2 = E[(x-m)^2] = \int_{-\infty}^{\infty} (x-m)^2\mathrm{f}(x)\mathrm{d}x \tag{6.3}$$

Covariance of two random processes:

$$\mathrm{con}(\mathrm{v},\mathrm{w}) = E[(\mathrm{v} - E[\mathrm{v}])(\mathrm{w} - E[\mathrm{w}])] \tag{6.4}$$

Two random processes v and w are said to be **independent** if their joint p.d.f.

$$\begin{aligned} &f(v,w) = f(v)f(w) \rightarrow \\ &E[vw] = \int\int_{-\infty}^{\infty} \mathrm{vwf}(v,w)\mathrm{d}v\mathrm{d}w = \int_{-\infty}^{\infty} \mathrm{vf}(v,w)\mathrm{d}v \int_{-\infty}^{\infty} \mathrm{wf}(v,w)\mathrm{d}w = E[v]E[w] \end{aligned} \tag{6.5}$$

Autocorrelation Function and Power Spectrum

Autocorrelation function is used to describe the time domain property of a random process. Given a random process v, its **autocorrelation function** is defined as follows:

$$R_x(t_1,t_2) = E[\mathrm{v}(t_1)\mathrm{v}(t_2)] \tag{6.6}$$

If v is a wide sense stationary (WSS) process,

$$R_x(t_1, t_2) = R_x(t_2 - t_1) = R_x(\tau) = R_x(t, t + \tau) = E[\mathrm{v}(t)\mathrm{v}(t + \tau)] \tag{6.7}$$

Note that $R_x(0)$ is the time average of the power energy of the random process.

Power spectrum of a random process is the Fourier transform of its autocorrelation function. It is a frequency domain property of the random process. To be more specific, it is defined as

$$S_x(\omega) = \int_{-\infty}^{\infty} R_x(\tau) e^{-j\omega t} \mathrm{d}\tau \tag{6.8}$$

White Noise, Colored Noise and Gaussian Random Process

White Noise is a random process with a constant power spectrum, and an autocorrelation function

$$R_x(\tau) = q \times \delta(\tau) \tag{6.9}$$

which implies that a white noise has an infinite power and thus it is non-existent in real life. It is not correlated in time. The power spectral density of white noise is flat and contains all frequencies. And it has infinite bandwidth, and therefore infinite variance and energy. Though white noise is not physically plausible, many noises (or the so-called coloured noises, or noises with finite energy and finite frequency components) can be modeled as the outputs of linear systems with an injection of white noise into their inputs, i.e., a coloured noise can be generated by a white noise

$$\text{white noise} \rightarrow \boxed{\text{Linear System}} \rightarrow \text{coloured noise} \tag{6.10}$$

Gaussian Process v is also known as a normal process and it has a p.d.f.

$$f(v) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(v-\mu)^2}{2\sigma^2}} \tag{6.11}$$

where μ is the mean, and σ^2 is variance.

Properties of a normally distributed process:

1. Normality is preserved through linear transformations.

2. The distribution is described by two parameters, mean value and variance.

One of the key assumptions of the Kalman filter is that all stochastic functions are assumed normally distributed.

The Discrete Kalman Filter

Rudolf E. Kalman started state-space description of linear systems in the 1960's. He presented the Kalman Filter at the first International Federation of Automatic Control (IFAC) conference in Russia. His paper describing a recursive solution of the discrete-data linear problem filtering was published in 1960. Advances in digital computer technology made it possible to consider implementing his recursive solution in a number of real-time applications, and Kalman filtering became popular. It is used in a wide range of engineering applications, such as navigation, signal processing, and control theory, and is implemented in just about every modern military and commercial control system, [10].

The Kalman filter is an efficient recursive filter that estimates the state of a linear or nonlinear dynamic system from a series of noisy measurements. It is simply an optimal recursive data processing algorithm, mostly known to be optimal with respect to minimum variance. The Kalman filter incorporates all information that can be provided to it, processes all available measurements, and combine the data to generate an overall best estimate. If there is temporarily loss of measurements the filter equations works as a predictor. When new measurements are available, the predictor is corrected and updates online to give the minimum variance estimate.

The key assumption when designing a Kalman filter is that the system model is observable. This is necessary in order to obtain convergence of the estimated states, $\hat{x}$, to the actual states, x.

6.2 Kalman Filters for Nonlinear Systems

The main limitation of the Kalman filter is that it relies on linearity, while in the real world most systems are nonlinear, [56]. Nonlinear systems are modeled in state-space and input-output representations. Some non-linear identification techniques include nonlinear filtering techniques, such as the extended Kalman

filter.

Some of the most successful applications of Kalman filtering have been in situations with nonlinear dynamics and/or nonlinear measurement relationships, [58].

6.2.1 The Extended Kalman Filter

If linearization of the system takes place about the filter's estimated trajectory, rather than a precomputed nominal trajectory, the filter is called an *extended Kalman filter*, [10]. Thus, the gain sequence is not predetermined by the process model assumptions as in the usual Kalman filter. The EKF is likely to diverge in unusual situations, especially when initial uncertainty and measurement errors are large, but still the EKF has been used in a variety of applications.

Extended Kalman Filter Design

If we take a nonlinear system in the form

$$\dot{\mathrm{x}} = f(\mathrm{x}, \mathrm{t}) + \mathrm{Bu} + \mathrm{Ew} \tag{6.12}$$

$$\mathrm{y} = \mathrm{Hx} + \mathrm{v} \tag{6.13}$$

where $f(\mathrm{x}, \mathrm{t})$ is a nonlinear vector field, the state vector can be estimated using the discrete-time extended Kalman filter algorithm, from [17], which is given in Algorithm 6.1.

Algorithm 6.1 Extended Kalman Filter

Design matrices

$$Q(k) = Q^\top(k) > 0 \tag{6.14}$$
$$R(k) = R^\top(k) > 0 \tag{6.15}$$

Initial conditions

$$\overline{x}(0) = x_0 \tag{6.16}$$
$$\overline{P}(0) = E[(x(0) - \hat{x}(0))(x(0) - \hat{x}(0))^\top] = P_0 \tag{6.17}$$

Kalman gain matrix

$$K(k) = \overline{P}(k)H^\top(k)[H(k)\overline{P}(k)H^\top(k) + R(k)]^{-1} \tag{6.18}$$

State estimate update

$$\hat{x}(k) = \overline{x}(k) + K(k)[y(k) - H(k)\overline{x}(k)] \tag{6.19}$$

Error covariance update

$$\hat{P}(k) = [I - K(k)H(k)]\overline{P}(k)[I - K(k)H(k)]^\top + K(k)R(k)K^\top(k), \tag{6.20}$$
$$\hat{P}(k) = \hat{P}(k)^\top > 0 \tag{6.21}$$

State estimate propagation

$$\overline{x}(k+1) = \mathcal{F}(\hat{x}(k), u(k)) \tag{6.22}$$

Error covariance propagation

$$\hat{P}(k+1) = \Phi(k)\hat{P}(k)\Phi^\top(k) + \Gamma(k)Q(k)\Gamma^\top(k) \tag{6.23}$$

$\mathcal{F}(\hat{x}(k), u(k))$,$\Phi(k)$ and $\Gamma(k)$ can be found, for instance, by using forward Euler integration.

6.2.2 The Unscented Kalman Filter

The extended Kalman filter has become a standard technique in the engineering literature, see Sec. 6.2.1. It is probably the most widely used estimator for nonlinear systems.

The EKF applies the Kalman filter method to nonlinear systems by simply linearizing all the nonlinear models so that the traditional linear Kalman filter equations can be applied. However, in practice, the use of the EKF has two well-known drawbacks, [31]:

- Linearization can produce highly unstable filters if the assumptions of local linearity is violated.
- The derivation of the Jacobian matrices are nontrivial in most applications and often lead to significant implementation difficulties.

Julier and Ulhmann, [31], proposed an extension to the KF-method, which generalises to nonlinear systems, without the linearisation steps required by the EKF, namely the unscented Kalman filter. The basic premise behind the UKF is that *it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function*, [51]. The UKF uses a deterministic sampling approach, known as the unscented transform, to capture the mean and covariance estimates with a minimal set of sample points. This technique removes the requirement to explicitly calculate Jacobians, so the algorithm has superior implementation properties to the EKF.

The Unscented Transformation

The unscented transformation (UT), is founded on the intuition that *it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function or transformation*, [51].

We find a set of deterministic vectors called sigma points. We compute a collection, and store the sigma points in the columns of the $L \times (2L + 1)$ sigma point matrix χ_{k-1} where L is the dimension of the state vector. The columns of χ_{k-1} are computed by, [49],

$$(\chi_{k-1})_0 = \hat{x}_{k-1} \tag{6.24}$$

$$(\chi_{k-1})_i = \hat{x}_{k-1} + (\sqrt{(L+\lambda)P_{k-1}})_i, i = 1 \dots L \tag{6.25}$$

$$(\chi_{k-1})_i = \hat{x}_{k-1} - (\sqrt{(L+\lambda)P_{k-1}})_{i-L}, i = L+1 \dots 2L, \tag{6.26}$$

where $(\sqrt{(L+\lambda)P_{k-1}})_i$ is the ith column of the matrix square root, and λ is defined by

$$\lambda = \alpha^2(L+\kappa) - L, \tag{6.27}$$

where α is a scaling parameter which determines the spread of the sigma point and κ is a secondary scaling parameter. The matrix square root should be calculated using a numerically efficient and stable method. With Cholesky decomposition a symmetric and positive definite matrix can be efficiently decomposed into a lower and upper triangular matrix.

The algorithm for the transformation procedure, [31], is given in Algorithm 6.2.

The unscented transform, given in Algorithm 6.2, is better suited than linearisation for filtering applications, [31]. Given its properties, and since it can predict the mean and covariance with second order accuracy, any filter that uses UT will have the same performance as the Truncated Second Order Gauss Filter, without requiring the derivation of Jacobians or Hessians, [3].

Unscented Kalman Filter Design

The unscented transformation developed in the previous section can be generalized to give the unscented Kalman filter.

The unscented Kalman filter can be summarized, from [31, 49, 57], as in Algorithm 6.3.

Algorithm 6.2 Unscented Transformation

1. Instantiate each point trough the function to yield the set of transformed sigma points,

$$(\chi_k)_i = f((\chi_{k-1})_i), i = 0 \dots 2L \tag{6.28}$$

2. The mean is given by the weighted average of the transformed points. With $(\chi_k)_i$ calculated, the *a priori* state estimate is

$$\hat{x}_k^- = \sum_{i=0}^{2L} W_i^{(m)} (\chi_k)_i, \tag{6.29}$$

where $W_i^{(m)}$ are weights defined by

$$\begin{aligned} W_0^{(m)} &= \frac{\lambda}{(L+\lambda)} \\ W_i^{(m)} &= \frac{1}{2(L+\lambda)}, i = 1 \dots 2L. \end{aligned} \tag{6.30}$$

3. The covariance is the weighted outer product of the transformed points. We calculate the *a priori* error covariance with

$$P_k^- = \sum_{i=0}^{2L} W_i^{(c)} \{(\chi_k)_i - \hat{x}_k^-\}\{(\chi_k)_i - \hat{x}_k^-\}^\top + Q_k, \tag{6.31}$$

where Q_k is once again the process error covariance matrix, and the weights are defined by

$$\begin{aligned} W_0^{(c)} &= \frac{\lambda}{(L+\lambda)} + (1 - \alpha^2 + \beta) \\ W_i^{(c)} &= \frac{1}{2(L+\lambda)}, i = 1 \dots 2L. \end{aligned} \tag{6.32}$$

Algorithm 6.3 Unscented Kalman Filter

1. Initialization:

$$\hat{x}_0^+ = E(x_0) \tag{6.33}$$

$$P_0^+ = E[(x_0 - \hat{x}_0^+)(x_0 - \hat{x}_0^+)^\top] \tag{6.34}$$

2. Iteration for each time step k:
   - Calculating the sigma points:

$$\hat{x}_{k-1}^{(i)} = \begin{pmatrix} \hat{x}_{k-1}^+ & \hat{x}_{k-1}^+ \pm \sqrt{(L+\lambda)P_{k-1}^+} \end{pmatrix} \tag{6.35}$$

   - Time update:

$$\hat{x}_k^{(i)} = f(\hat{x}_{k-1}^{(i)}, u_k, t_k) \tag{6.36}$$

$$\hat{x}_k^- = \sum_{i=1}^{2L} W_i^{(m)} \hat{x}_k^{(i)} \tag{6.37}$$

$$P_k^- = \sum_{i=1}^{2L} W_i^{(c)} [\hat{x}_k^{(i)} - \hat{x}_k^-][\hat{x}_k^{(i)} - \hat{x}_k^-]^\top + Q_{k-1} \tag{6.38}$$

$$\hat{y}_k^{(i)} = h(\hat{x}_k^{(i)}, t_k) \tag{6.39}$$

$$\hat{y}_k = \sum_{i=1}^{2L} W_i^{(m)} \hat{y}_k^{(i)} \tag{6.40}$$

   - Covariance and cross covariance

$$P_y = \sum_{i=1}^{2L} W_i^{(c)} [\hat{y}_k^{(i)} - \hat{y}_k][\hat{y}_k^{(i)} - \hat{y}_k]^\top + R_k \tag{6.41}$$

$$P_{xy} = \sum_{i=1}^{2L} W_i^{(c)} [\hat{x}_k^{(i)} - \hat{x}_k^-][\hat{y}_k^{(i)} - \hat{y}_k]^\top \tag{6.42}$$

$$\tag{6.43}$$

   - Measurement update:
     UKF Kalman gain:

$$K_k = P_{xy} P_y^{-1} \tag{6.44}$$

     Updated state estimate and covariance:

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - \hat{y}_k) \tag{6.45}$$

$$P_k^+ = P_k^- - K_k P_y K_k^\top \tag{6.46}$$

The L in the previous equations is the dimension of the extended state space. In Eq. (6.38) we add Q_{k-1}, the process noise covariance, to the end of the equation to take the process noise into account. To take measurement noise into account, we should add R_k, the measurement noise covariance, to the end of Eq. (6.41). $W_i^{(m)}$, and $W_i^{(c)}$ are the weights calculated by Eq. (6.30), and Eq. (6.32) respectively.

Chapter 7

Kalman Filter Implementation

In the previous chapter two kinds of Kalman filters were exlained. In this chapter the implementation of the observers are described, the plots of the results are presented, and the encounter frequency plot and value is given.

7.1 Extended Kalman Filter

In this section we design an extended Kalman filter.

The objective of the observer is to design the Kalman filter gain, K, such that we find the optimal estimate $\hat{x}(t)$. Since the system is observable, see Sec. 3.2.2, the state vector, $\mathbf{x}$, can be reconstructed recursively through the measurement vector, $\mathbf{y}$, and the control input vector, $\mathbf{u}$.

We can write our nonlinear system on the form

$$\dot{x} = f(x,t) + Bu \tag{7.1}$$
$$y = Hx \tag{7.2}$$

where u equals the control input. In our model this is basically the thrust, τ_1.

$$\dot{x} = \begin{bmatrix} x_2 \\ \frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi 3}}{I_x+A_{44}}x_1^3 \\ -\frac{B_{11}}{m+A_{11}}|x_3|x_3 \\ 0 \\ x_6 \\ -x_5(x_4 + \frac{x_3 x_4^2}{g})^2 - \frac{x_4 x_6 B_{11}}{(x_3 x_4+g)(m+A_{11})}|x_3|x_3 \\ x_8 \\ -\omega_0^2 x_7 - 2\lambda\omega_0 x_8 + K_w w \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m+A_{11}} \\ 0 \\ 0 \\ \frac{x_4 x_6}{x_3 x_4+g}\frac{1}{m+A_{11}} \\ 0 \\ 0 \end{bmatrix} \tau_1 \tag{7.3}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x \tag{7.4}$$

7.1.1 Linearization of The Model

The extended Kalman filter uses a two step predictor–corrector algorithm. As part of the algorithm the Jacobian, A, of the system is required. For our system model the Jacobian is calculated by

$$A = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}) \tag{7.5}$$

The Jacobian A, is different at each time step, but the time step subscript k has been omitted for simplicity.

For our system, the Jacobian becomes,

$$A = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x})$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial f_{[2]}}{\partial x_{[1]}} & \frac{B_{44}}{I_x+A_{44}} & 0 & 0 & -\frac{C_{44}\overline{GM}_a\hat{x}_1}{I_x+A_{44}} & 0 & 0 & 0 \\ 0 & 0 & -\frac{B_{11}(|\hat{x}_3|+\hat{x}_3 sign(\hat{x}_3))}{m+A_{11}} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{\partial f_{[6]}}{\partial x_{[3]}} & \frac{\partial f_{[6]}}{\partial x_{[4]}} & \frac{\partial f_{[6]}}{\partial x_{[5]}} & \frac{\partial f_{[6]}}{\partial x_{[6]}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_0^2 & -2\lambda\omega_0 \end{bmatrix} \tag{7.6}$$

where,

$$\frac{\partial f_{[2]}}{\partial x_{[1]}} = -\frac{C_{44}(\overline{GM}_m + \overline{GM}_a\hat{x}_5)}{I_x + A_{44}} + \frac{3K_{\phi^3}\hat{x}_1^2}{I_x + A_{44}} \tag{7.7}$$

$$\frac{\partial f_{[6]}}{\partial x_{[3]}} = -\frac{2\hat{x}_4^3\hat{x}_5}{g} - \frac{\hat{x}_3\hat{x}_4^4\hat{x}_5}{g^2} - \frac{\hat{x}_4\hat{x}_6 B_{11}[(\hat{x}_3\hat{x}_4 + g)(|\hat{x}_3| + \hat{x}_3 sign(\hat{x}_3)) - (|\hat{x}_3|\hat{x}_3\hat{x}_4)]}{(m + A_{11})(\hat{x}_3\hat{x}_4 + g)^2} \tag{7.8}$$

$$\frac{\partial f_{[6]}}{\partial x_{[4]}} = -\hat{x}_4\hat{x}_5 - \frac{2\hat{x}_3\hat{x}_4^2\hat{x}_5}{g} - \frac{\hat{x}_3^2\hat{x}_4^3\hat{x}_5}{g^2} - \frac{\hat{x}_3\hat{x}_6 B_{11}}{(\hat{x}_3\hat{x}_4 + g)(m + A_{11})} \tag{7.9}$$

$$\frac{\partial f_{[6]}}{\partial x_{[5]}} = -\hat{x}_4^2 - \frac{2\hat{x}_3\hat{x}_4^3}{g} - \frac{\hat{x}_3^2\hat{x}_4^4}{g^2} \tag{7.10}$$

$$\frac{\partial f_{[6]}}{\partial x_{[6]}} = -\frac{\hat{x}_3\hat{x}_4 B_{11}}{(\hat{x}_3\hat{x}_4 + g)(m + A_{11})} \tag{7.11}$$

7.1.2 Discretization of The Model

To implement the observer on a computer, the model needs to be discretized. The discretization has the form, [17],

$$x(k+1) = \Phi x(k) + \Delta u(k) + \Gamma w(k) \tag{7.12}$$

$$y(k) = Hx(k) + v(k) \tag{7.13}$$

where

$$\Phi = exp(Ah), \tag{7.14}$$
$$\Delta = A^{-1}(\Phi - I)B, \tag{7.15}$$
$$\Gamma = A^{-1}(\Phi - I)E, \tag{7.16}$$

h is the sampling time, and the equivalent discrete-time noises $w(k)$ and $v(k)$ are Gaussian and white with zero mean.

7.1.3 The Extended Kalman Filter in Matlab

We implement the EKF by use of the discrete-time extended Kalman filter algorithm, see Algorithm 6.1. The Matlab file for the EKF, EKF.M, is given in Appendix G.

Assumptions

For implementation of the extended Kalman filter we made some assumptions;

- The modal wave frequency, ω_0 is constant, or slowly time-varying.
- The ship is in a head-sea condition.
- We assume that forward speed, roll angle and roll rate are measured.
- The second state of the wave model can be added to the measurement of the speed.
- We also assume that the unknown output, the encounter frequency, is slowly changing, almost constant.

Parameter Values

To implement a Kalman filter, the parameters of the model are necessary. The parameter values we need, for the ship model, we get from running the file INIT.M, given in Appendix G.

Input Value

The input, u, to the model, basically the thrust, τ_1, is computed through a feedback linearization controller, as described in Chapter 4. The Matlab file for the calculation of the input, COMPUTE_INPUT.M, is given in Appendix G.

Design Matrices

The process covariance matrix Q, used in the filter model, is meant to account for modeling errors. Thus, the choice of the matrix Q is a critical part of the filter design parameters.

Matrix R is the measurement noise covariance. It is used in the filter model to account for measurement errors. It is often much easier to set up, than the Q matrix. This is because the variance of the measurement noise is often given in commercial instruments of measurement.

If the components of the noise vector do not affect each other, the matrices Q and R are diagonal matrices, as in our case.

When choosing the covariance matrices, the relationship between Q and R are the most important issue, regarding the result of the Kalman filter. Matrices Q and R, to give desired response, were found by trial and error.

For the discretization, see Sec. 7.1.2, matrix Φ and Γ are computed by

$$[\texttt{PHI}, \texttt{GAMMA}] = c2d(A, E, h); \tag{7.17}$$

where $c2d$ is a built-in Matlab function which convert systems from continuous- to discrete-time models.

Noise Generation

The Box-Muller transform, [5], is a pseudo-random number sampling method for generating pairs of independent, standard, normally distributed (zero expectation, unit variance) random numbers, given a source of uniformly distributed random numbers.

We use this method to generate normally distributed Gaussian noise to our measurement equation. The M-file for the Box-Muller transform, BoxMuller.M, is given in Appendix G.

Integration

To integrate the system of ordinary differential equations (ODEs) we make use of Runge-Kutta's 4th-order method. More accurate integration algorithms have also been derived and are sometimes used, but fourth-order Runge-Kutta integration is generally considered a good trade-off between accuracy and computational effort, [49]. The input, u, is constant for each time step, and we have a step size of 0.1. The M-file for the Runge-Kutta method, RK4.M, from [19], is given in Appendix G.

7.2 Plot of Simulation Results from EKF

After implementing an extended Kalman filter for the complete system, including the effects of the wave characteristics onto the speed dynamics, we plot the reults.

Figure 7.1 - Figure 7.8, depicts the results of the simulation with the EKF. The plots shows values of the 8 different states in Eq. (2.17), against their estimates. Every plot also depicts the error between simulation and estimate, for each state respectively.

Figure 7.1 shows estimated roll angle against simulated value. The estimation error is small, and almost constant. The graph shows that the EKF can produce an almost noise-free estimate, even with the measurement exposed to noise.

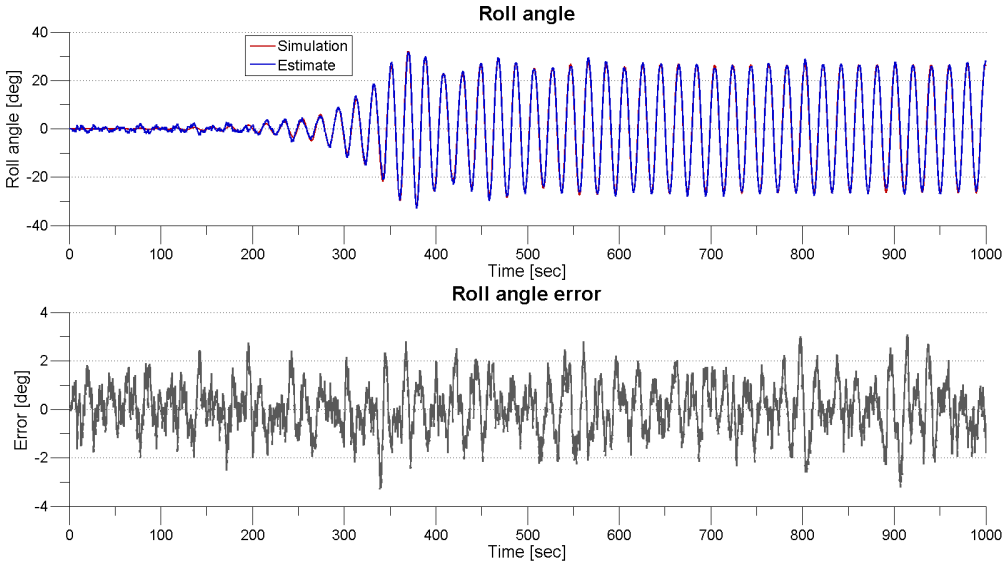

Figure 7.1: Roll angle ϕ, and error plot, with EKF.

In Figure 7.2 we see the estimated rate of roll angle against the simulated value. The EKF can produce an almost noise-free estimate also for the rate of the roll angle, even though the measurement and the process is contaminated by white noise.

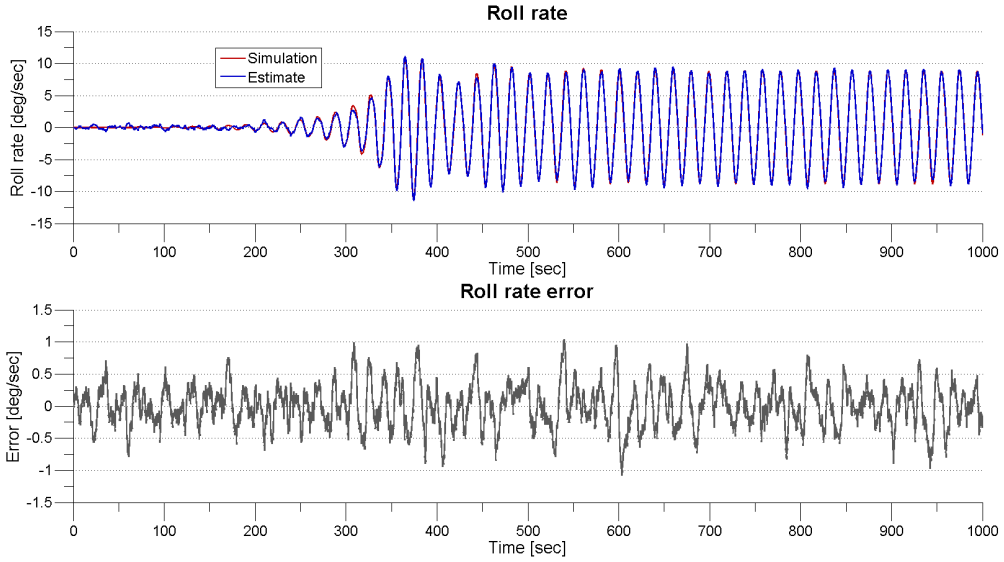

Figure 7.2: Roll rate $\dot{\phi}$, and error plot, with EKF.

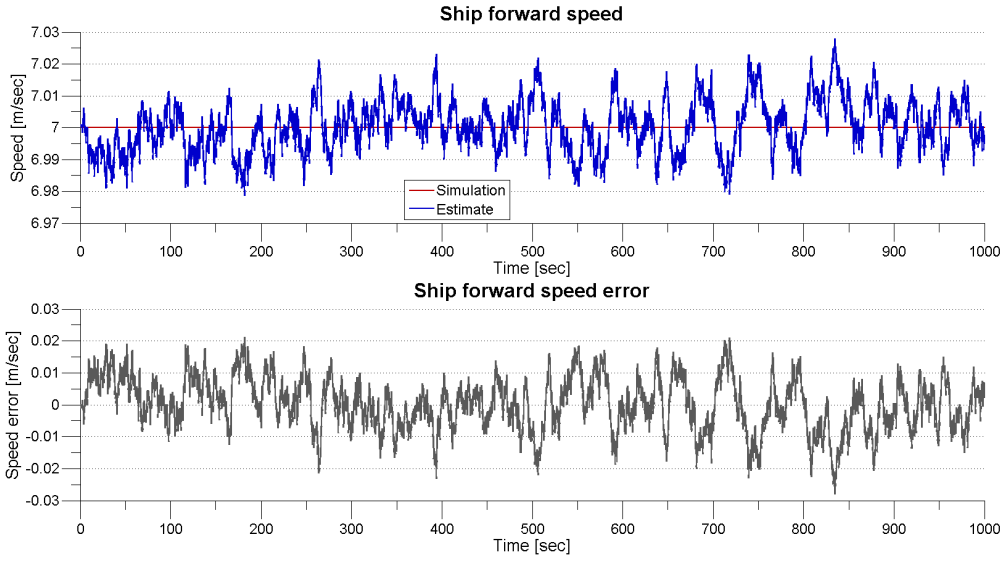

Figure 7.3: Ship speed u, and error plot, with EKF.

Estimated speed against simulation is shown in Figure 7.3. We can see that the simulated speed goes to a value of $7[m/s]$ very fast, and stays there. The estimated signal follows the simulated one, just with noise added to it. From the plot of the speed error, in the bottom part of Figure 7.3, we see that the estimation value is contaminated by noise, but has an error of about maximum $0.025[m/sec]$.

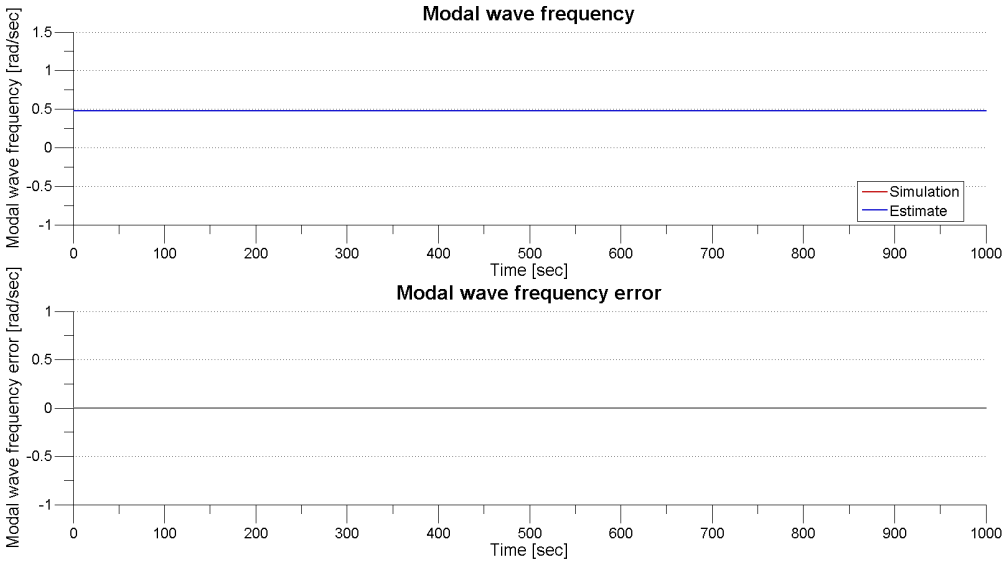

Figure 7.4: Modal wave frequency ω_0, and error plot, with EKF.

Figure 7.4 depicts estimation of the modal wave frequency against simulated value. We can see that the simulated value is constant equal to $0.4764[rad/s]$. The estimate follows the simulated value without any error, which can also be seen in the plot of the modal wave frequency error, in the bottom part of Figure 7.4.

Figure 7.5 and Figure 7.6 depicts the plots of the cos term, $\cos(\int_0^t \omega_e(\tau)d\tau)$, and the rate of the cos term, $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$, respectively. We see their estimates against simulated values. It can be seen that the estimate is almost equal to the simulation value, for both the plot in Figure 7.5, and the plot in Figure 7.6. From the bottom part of both figures, we see that the error between the estimate and simulated value of $\cos(\int_0^t \omega_e(\tau)d\tau)$, and its rate, is almost similar. But the error of $\cos(\int_0^t \omega_e(\tau)d\tau)$ has a greater amplitude than the error of $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$. For $\cos(\int_0^t \omega_e(\tau)d\tau)$ we have an error of up to almost $0.1[rad/sec] \approx 5.72[deg/sec]$, while the error for $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$ is 10^{-3}.

From Figure 7.5 we can see that the EKF implemented is able to estimate the frequency of encounter, from a noise-contaminated process.

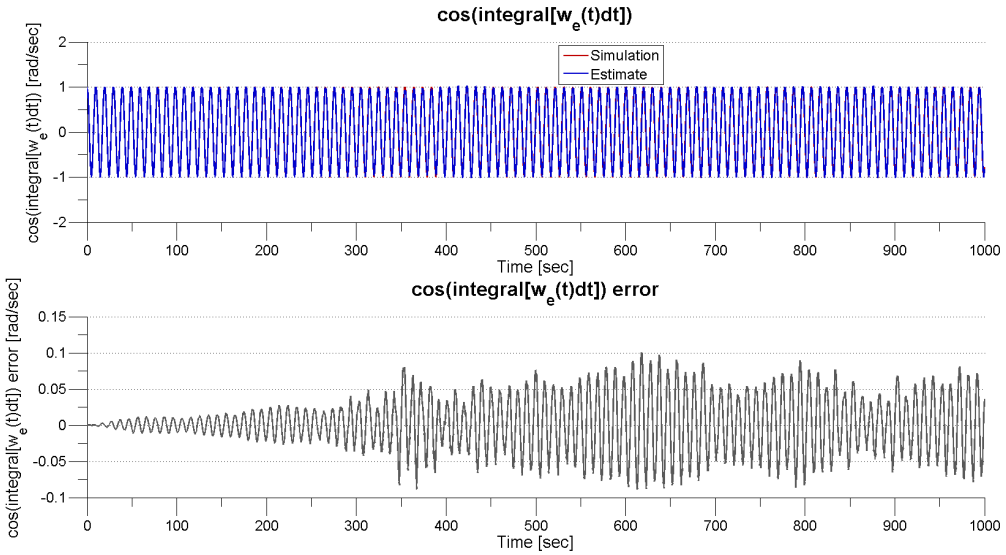

Figure 7.5: Cosinus term $\cos(\int_0^t \omega_e(\tau)d\tau)$, and error plot, with EKF.

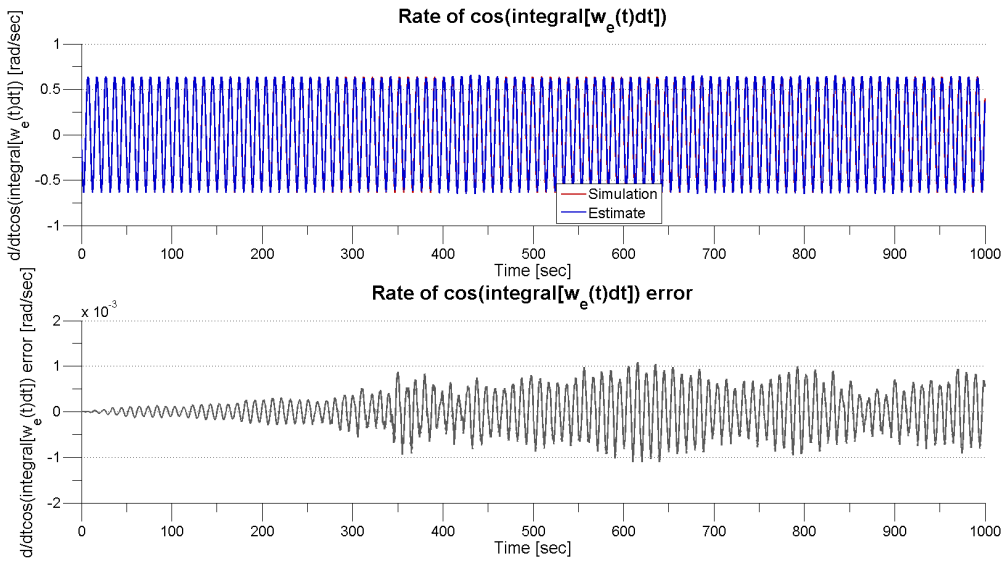

Figure 7.6: Rate of cos term $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$, and error plot, with EKF.

Figure 7.7 and Figure 7.8 depicts the plots of the wave induced forces, and their rate, respectively. The wave induced forces are contaminated by white noise. The estimated values are not equal to the simulated ones, but as we can see in the error plots, the difference between simulation and estimate is small.

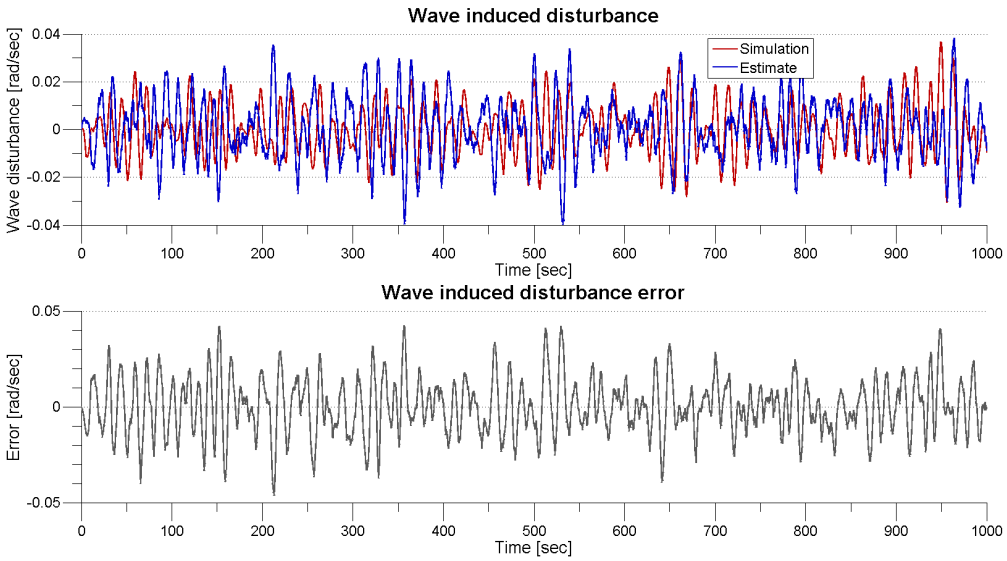

Figure 7.7: Wave induced disturbance h_{wave}, and error plot, with EKF.

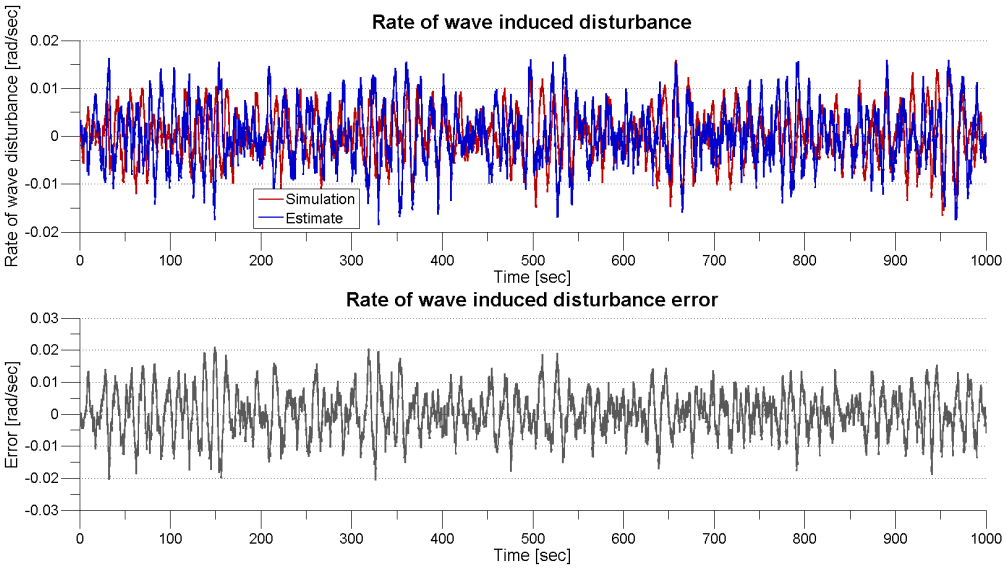

Figure 7.8: Rate of wave induced disturbance $\frac{d}{dt}h_{wave}$, and error plot, with EKF.

7.2.1 Slowly Time-Varying Modal Wave Frequency in EKF

One assumption when implementing the EKF was a constant, or slowly time-varying, modal wave frequency ω_0. For the above simulations ω_0 was constant equal $0.4764[rad/sec]$. We also conducted simulations where ω_0 was changed at each iteration of the algorithm.

Figure 7.9 shows plot of a time-varying ω_0. As we can see, the EKF is not able to estimate it. It might be that the filter needs to be reset every time ω_0 changes.

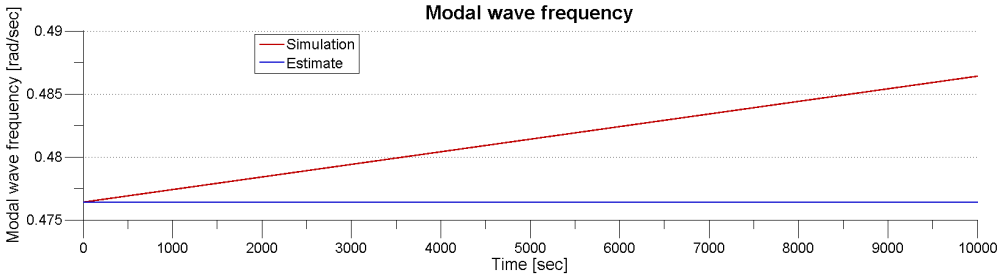

Figure 7.9: Modal wave frequency ω_0, and error plot, with UKF.

In Figure 7.10 we see state 5, from 7000 to 8000 seconds of the simulation (10 000 sec in total), from running the EKF with a slowly time-varying ω_0. We can see from the plot that the estimate has lower amplitude than the actual value. The error increased throughout the simulation, as ω_0 got bigger.

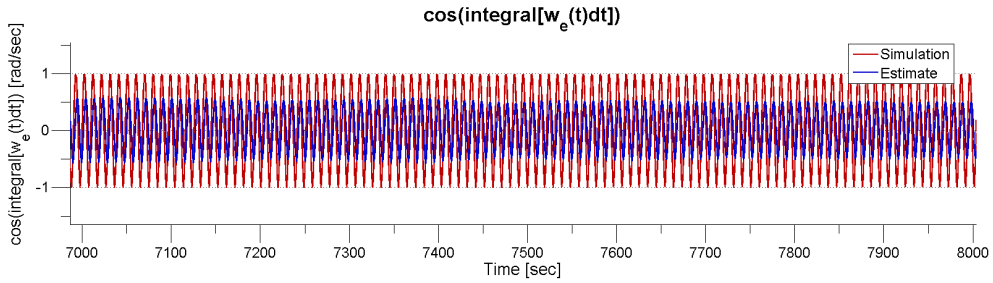

Figure 7.10: State 5, and error plot, with slowly time-varying ω_0 in EKF.

The EKF does not make a good estimate of state 5 $\cos(\int_0^t \omega_e(\tau)d\tau)$, when the modal wave frequency is slowly time-varying. And thus a bad estimation of the wave encounter frequency.

7.2.2 The Estimated Wave Encounter Frequency from EKF

We have estimated the term $\cos(\int_0^t \omega_e(\tau)d\tau)$ with the EKF. An example is given in Figure 7.5. Now we want to find the wave encounter frequency ω_e, from this signal.

A common use of Fourier transforms is to find the frequency components of a signal buried in a noisy time domain signal. It is difficult to identify the frequency components by looking at the original signal. Converting to the frequency domain, the discrete Fourier transform of the noisy signal is found by taking the fast Fourier transform (FFT). The Matlab file FFT_EncFreq.M, given in Appendix G, calculates the discrete Fourier transform of state 5, finds the value of ω_e, and plots a single-sided amplitude spectrum.

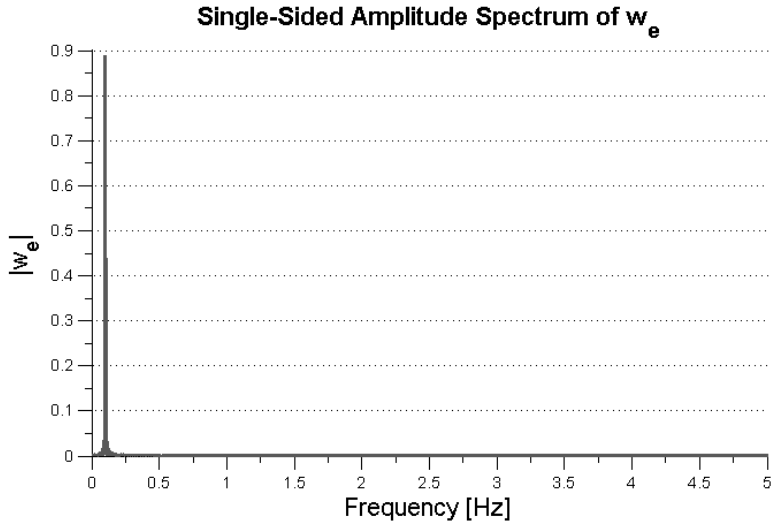

Figure 7.11: Frequency-domain data for nonlinear parametric roll resonance.

Figure 7.11 shows the single-sided amplitude spectrum of the wave encounter frequency. The main reason the amplitudes is not exactly at 1 is because of the noise. Another reason is that we have a finite length signal.

This plot was conducted through the EKF, with constant modal wave frequency. For this reason we get a constant wave encounter frequency as well.

We get the value of ω_e by multiplying the maximum value of the single-sided amplitude spectrum with $2 \times \pi$.

From this we get that the wave encounter frequency, estimated with the EKF, equals $0.6365[rad/s]$.

7.3 Unscented Kalman Filter

In this section we design an unscented Kalman filter.

7.3.1 Unscented Transformation

While the EKF uses a linearizing Jacobian matrix, Eq. (7.6), which is a first order approximation, the unscented Kalman filter uses a deterministic sampling approach to capture the mean and covariance estimates with a minimal set of sample points. It has a 3rd order (Taylor series expansion) accuracy for Gaussian error distribution for any nonlinear system, [57].

The unscented transformation is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation. There is one unscented transformations for the prediction step, and one for the update. The M-file UT_WEIGHTS.M computes unscented transformation weights, see Appendix G.

7.3.2 Sigma Point Propagation of the UKF

The implementation of the unscented Kalman filter needs to create sigma points from a mean and a covariance. The estimated state and covariance are augmented with the mean and covariance of the process noise. A set of sigma points are then derived from the augmented state and covariance. The function SIGMAS.M, given in Appendix G, returns the sigma points, χ, around a reference point. The matrix square root is calculated using Cholesky decomposition, a numerically efficient and stable method. We use the lower triangular matrix of the Cholesky factorization, which is computed with the built-in Matlab function *chol*.

For the prediction step the sigma points are propagated through the transition function f. And then the weighted sigma points are recombined to produce the predicted state and covariance.

In the update step, the sigma points are projected through the observation function h, before the weighted sigma points are recombined to produce the predicted measurement and predicted measurement covariance.

7.3.3 The Unscented Kalman Filter in Matlab

We implement the UKF procedure as in Algorithm 6.3. The Matlab file for the unscented Kalman filter, UKF.M, is given in Appendix G.

Assumptions

For implementation of the unscented Kalman filter we make the same assumptions as for the extended Kalman filter. See Sec. 7.1.3.

Parameter Values

We get all the parameter values we need, for the ship model, from running the file INIT.M, given in Appendix G.

For the UKF, we also needed to set the α, β and κ parameters. After running a couple of simulations, we found that $\alpha = 10^{-4}, \beta = 2$ and $\kappa = 0$ were appropriate numerical values for these parameters.

Input Value

The input, u, to the model, basically the thrust, τ_1, is computed through a feedback linearization controller, as described in Chapter 4. The Matlab file for the calculation of the input, COMPUTE_INPUT.M, is given in Appendix G.

Design Matrices

We needed to determine the R and Q covariance matrices. The off-diagonal entries were set to 0. And the values on the diagonal in both the Q and R matrix were defined by trial and error.

Noise Generation

We use the Box-Muller transform to generate normally distributed Gaussian noise to our measurement equation. The M-file for the Box-Muller transform is given in Appendix G.

Integration

As in the EKF, we make use of a 4th-order Runge Kutta integration between the time steps, with input u constant for each time step, and a step size of 0.1. The M-file, RK4.M, from [19], is given in Appendix G.

7.4 Plot of Simulation Results from UKF

After making an unscented Kalman filter for the complete system, including the effects of the wave characteristics onto the speed dynamics, we plot the reults.

Figure 7.12 - Figure 7.19, depicts the results of the simulation with the UKF. The plots shows values for the 8 different states in Eq. (2.17), against their estimates. Every plot also depicts the error between simulation and estimate, for each state respectively.

Figure 7.12 depicts estimated roll angle against simulated value. We can see that the estimation error is small, and almost constant. The graph show that the estimate given by the UKF is almost noise-free, even with the roll angle measurement exposed to noise.

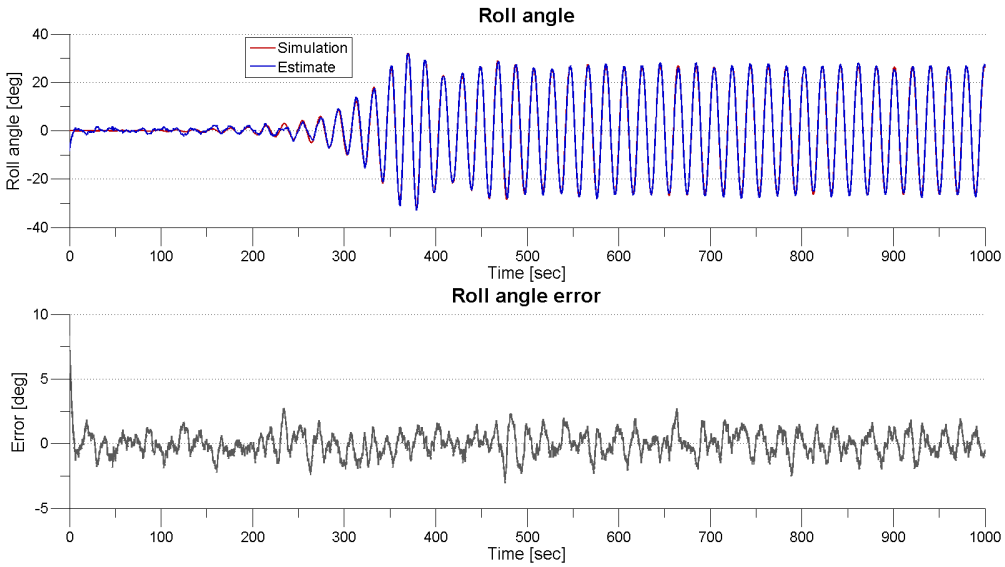

Figure 7.12: Roll angle ϕ, and error plot, with UKF.

This can also be seen from Figure 7.13, which shows the estimate of the rate of

roll angle against simulation. The error is very small, and nearly constant. The graph for the roll rate error can be seen in the bottom plot in Figure 7.13.

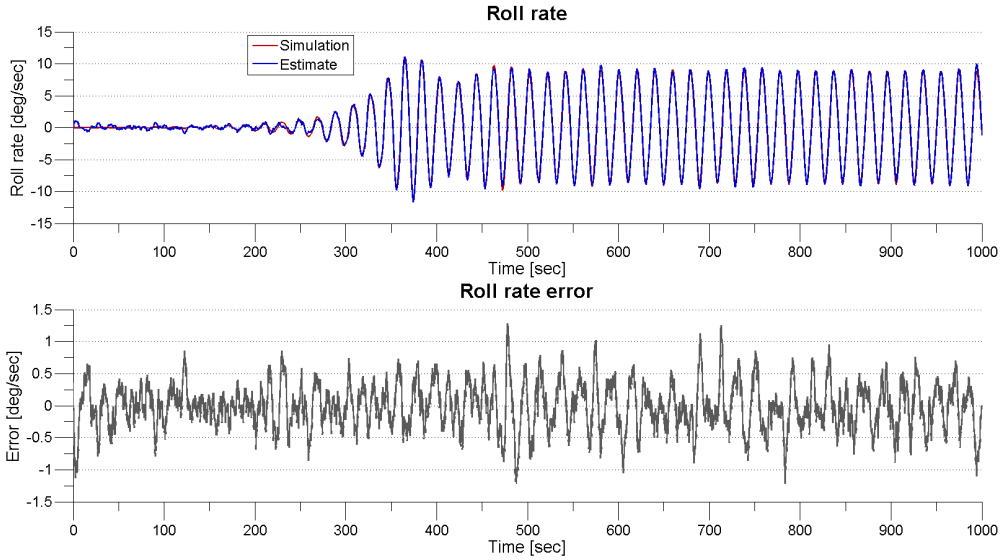

Figure 7.13: Roll rate $\dot{\phi}$, and error plot, with UKF.

In Figure 7.14, the actual speed almost immediately gets a value of $7[m/sec]$, and stays constant the whole simulation. The estimate is almost equal to the simulated value. This can also be seen in the error plot, with error of power 10^{-9}.

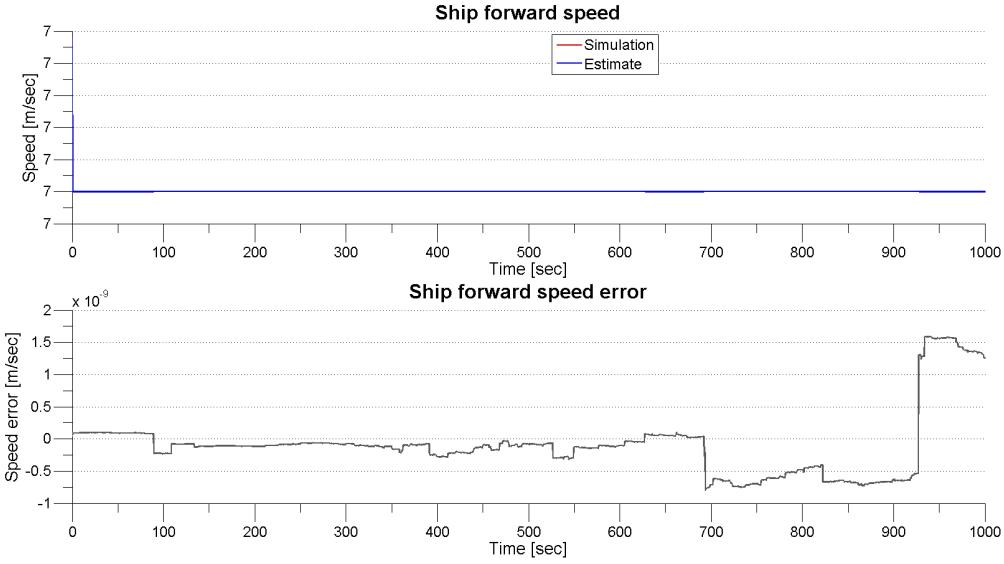

Figure 7.14: Ship speed u, and error plot, with UKF.

We assumed that modal wave frequency is constant, or slightly time-varying. ω_0 is here set to a constant value equal to $0.4764[rad/sec]$, which is reflected in the constant simulation value. Figure 7.15 shows the estimate follow this value perfectly, and an error signal of zero value.

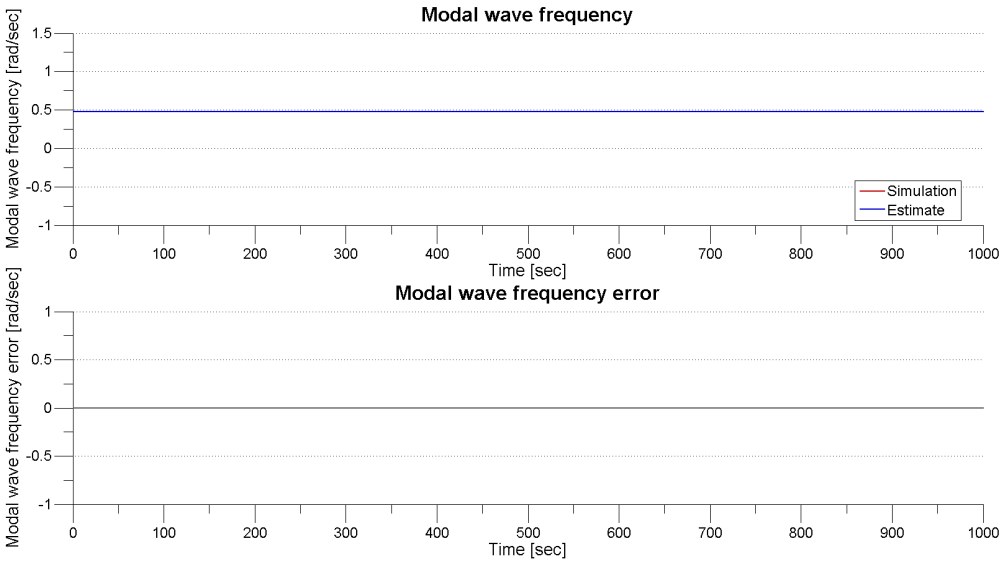

Figure 7.15: Modal wave frequency ω_0, and error plot, with UKF.

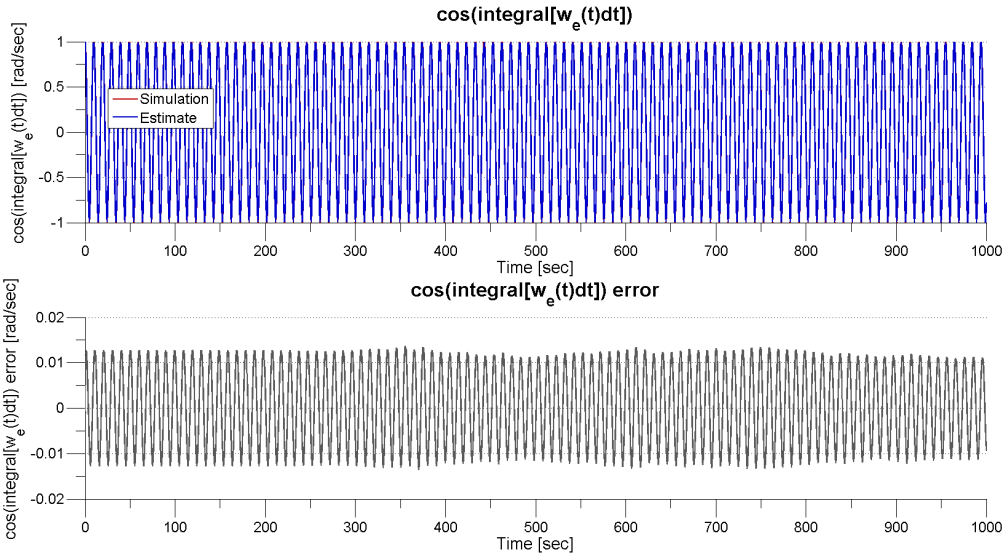

Figure 7.16: Cosinus term $\cos(\int_0^t \omega_e(\tau)d\tau)$, and error plot, with UKF.

Figure 7.16 and Figure 7.17 depicts the plots of the cos term, $\cos(\int_0^t \omega_e(\tau)d\tau)$, and $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$, respectively. The graphs in the error plots have the same shape,

and we can see that the oscillations gets bigger to the end of the simulation. The error of state 5 is greater than the error of state 6. From Figure 7.16 we can see that the implemented UKF is able to estimate the frequency of encounter, from a noise-contaminated process.

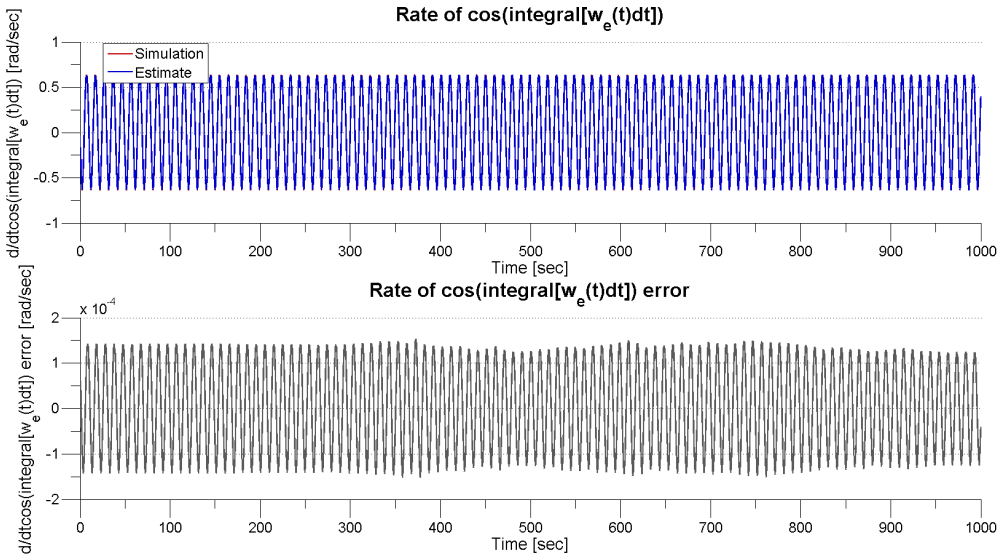

Figure 7.17: Rate of cos term $\frac{d}{dt}\cos(\int_0^t \omega_e(\tau)d\tau)$, and error plot, with UKF.

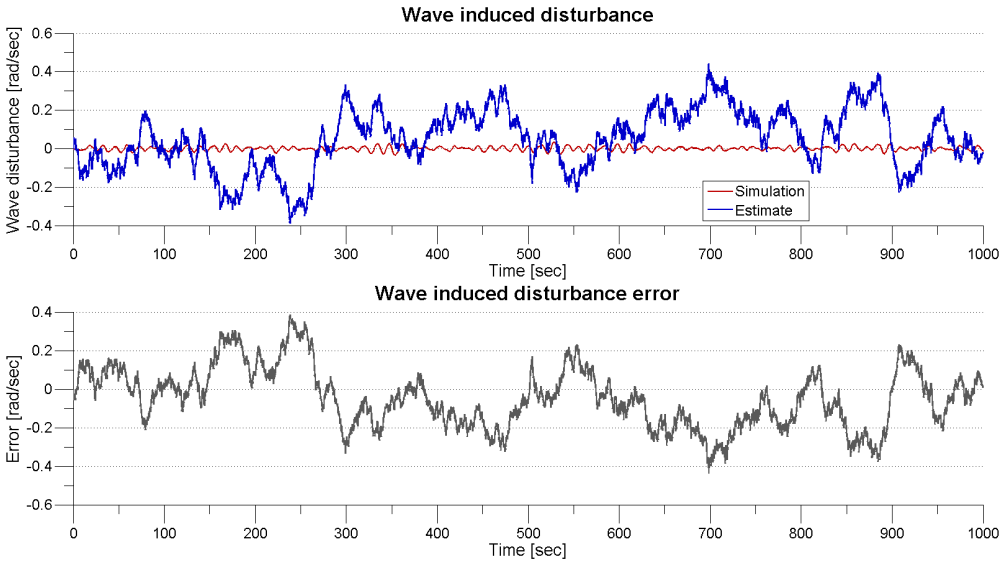

Figure 7.18: Wave induced disturbance h_{wave}, and error plot, with UKF.

Figure 7.18 depicts the plot of the wave induced forces. The wave induced forces are contaminated by white noise, and we can see from the plots that the simulation

value never stabilizes. The estimated value is a wave with higher amplitude.

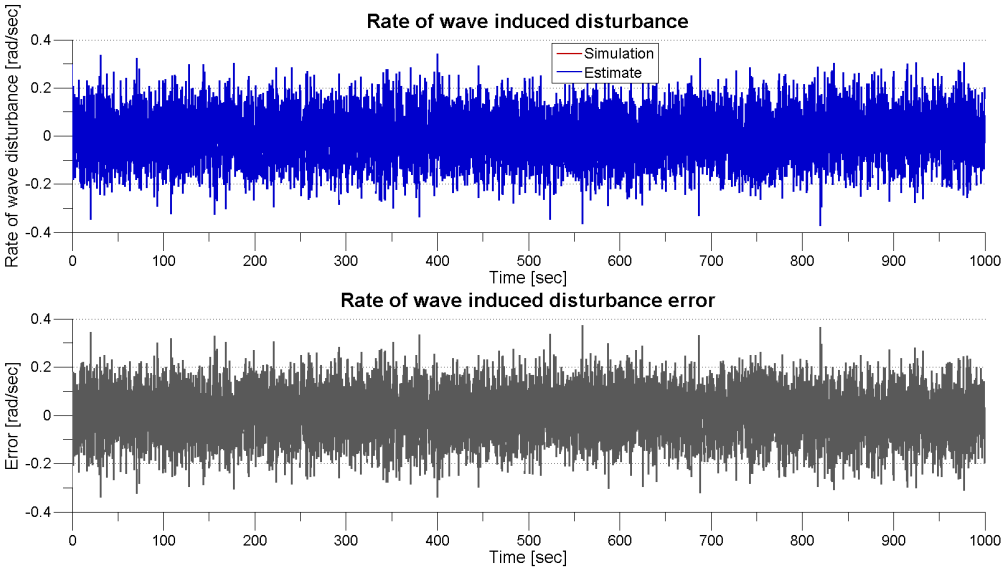

Figure 7.19: Rate of wave induced disturbance $\frac{d}{dt}h_{wave}$, and error plot, with UKF.

In Figure 7.19 we see the plot of the rate of the wave induced forces. This is noise contaminated, and the error plot shows that the simulated, and the estimated value is equally noisy. The UKF is not able to make a good estimate of the wave induced forces.

7.4.1 Slowly Time-Varying Modal Wave Frequency in UKF

As mentioned earlier one of the assumptions for the UKF algorithm implementation was that the modal wave frequency, ω_0, is to be constant, or slowly time-varying. For the above simulations ω_0 had a constant value of $0.4764[rad/sec]$. We have also done simulations where the value of the modal wave frequency was changed slowly at each iteration of the algorithm.

The result for the estimate of the modal wave frequency, when running the UKF with a slightly time-varying ω_0 was equal to the plot in Figure 7.9. The UKF, as the EKF, is not able to estimate the modal wave frequency, when it is not kept constant. It might be that the UKF also needs to be reset every time the modal wave frequency changes.

In Figure 7.20 we can see the estimated state 5, $\cos(\int_0^t \omega_e(\tau)d\tau)$, from 7000 to 8000 seconds of the simulation in the UKF. This plot was made with a slightly time-

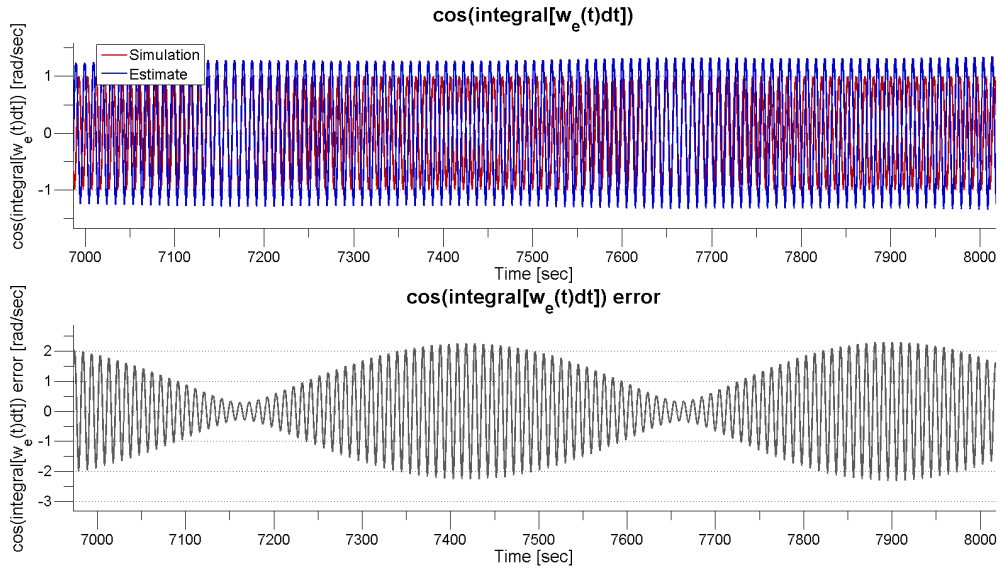

Figure 7.20: State 5, and error plot, with slowly time-varying ω_0 in UKF.

varying ω_0 in the filter. We can see that the estimate has a much larger amplitude than the simulated value. The error is shown in the bottom plot of Figure 7.20.

The UKF, as well as the EKF, works much better, for estimation of the wave encounter frequency, if the modal wave frequency is kept constant throughout the simulation.

7.4.2 The Estimated Wave Encounter Frequency from UKF

In Sec. 7.2.2 we showed how we could find the wave encounter frequency from estimated state 5 from the EKF, namely the term $\cos(\int_0^t \omega_e(\tau)d\tau)$. Likewise, we want to find the estimated encounter frequency, ω_e from the UKF.

We make use of the FFT in the same way as in Sec. 7.2.2. The plot of the single sided amplitude spectrum of ω_e against the frequency [Hz], is given in Figure 7.21.

Like for the EKF, the FFT signal was made from estimation of state 5 with constant modal wave frequency. Thus we get a constant wave encounter frequency from the UKF as well.

We get the value of ω_e by multiplying the maximum value of the single-sided amplitude spectrum with $2 \times \pi$.

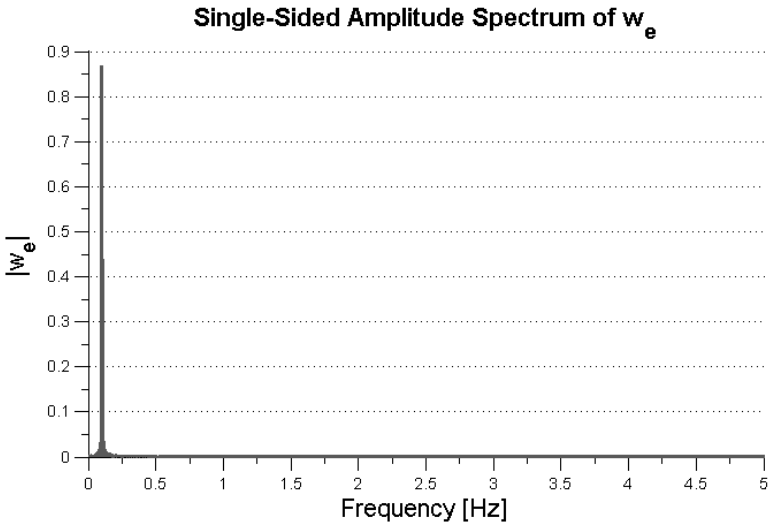

Figure 7.21: Frequency-domain data for nonlinear parametric roll resonance, from UKF.

From this we get that the wave encounter frequency, estimated with the UKF, equals $0.6365[rad/s]$.

Chapter 8

Comparison of the Observers

> It appears that no particular approximate [nonlinear] filter is consistently better than any other, though ... any nonlinear filter is better than a strictly linear one.
>
> - Lawrence Schwartz and Edwin Stear [47]

Chapter 7 explained how the EKF, and UKF, were implemented in Matlab, for estimation of the frequency of encounter, and modal wave frequency. In this chapter, we compare the results, and findings, of the different nonlinear state estimators.

8.1 A Comparison of Extended and Unscented Kalman Filtering for Estimating Parametric Roll

Many scientific studies require nonlinear state-space models with which the linear Kalman filter is inapplicable. The extended Kalman filter has become a standard technique in the engineering literature. It is the most widely used nonlinear state estimation technique that has been applied in the past few decades, [49]. The EKF approximates a nonlinear model by its first order Taylor expansion, so the traditional KF can be applied. Unfortunately, the EKF has two important potential drawbacks, [46]. First, the approximation error becomes non-negligible with strongly nonlinear models. Another drawback is that the EKF requires calculating the Jacobian, which may not be available, or very complex. The linear

approximations to the nonlinear functions can be complex, causing implementation difficulties.

The EKF is a set of mathematical equations which uses an underlying process model to make an estimate of the current state of the system and corrects the estimate using any available sensor measurements. The state distribution is approximated by a Gaussian random variable which is propagated analytically through the "first-order" linearization of the nonlinear system. As such, the EKF can be viewed as providing "first-order" approximations to the optimal terms, [57]. Using this predictor-corrector mechanism, it approximates an optimal estimate from the linearization of the process and measurements models, [10]. These approximations, however, can introduce large errors in the true posterior mean and covariance of the transformed (Gaussian) random variable, which may lead to sub-optimal performance and sometimes divergence of the filter.

It is necessary to point out that a fundamental issue with the EKF is that the distributions of the various random variables are no longer normal after undergoing their respective nonlinear transformations. The EKF is simply an *ad hoc* state estimation that only approximates the optimality of Bayes' rule by linearization, [10].

The unscented Kalman filter provide ways to reduce the linearization errors that are inherent in the EKF. Instead of relying on linearization techniques employed by the EKF, the UKF extends Kalman filters to nonlinear models using a deterministic sampling scheme with a minimal set of carefully chosen sample points. In each step, the UKF generates a set of sigma points and updates the prediction formulas based on these sigma points. When propagated through the true nonlinear system, they capture the posterior mean and covariance accurately to the 3rd order (Taylor series expansion) for any nonlinearity.

A demonstration of the accuracy of the scaled unscented transformation for mean and covariance propagation is shown in Figure 8.1, for a 2-dimensional system. The left plot shows the true mean and covariance propagation using Monte-Carlo sampling, the center plots show the results using a linearization approach as would be done in the EKF, and the right plots show the performance of the unscented transformation (note only 5 sigma points are required). For the UT there is almost no bias error in the estimate of the mean, and the estimated covariance is also much closer to the true covariance. The superior performance of the sigma-point approach is clearly evident.

The Unscented Kalman Filter is a straightforward extension of the UT to recursive estimation. It can give greatly improved estimation performance (compared to the

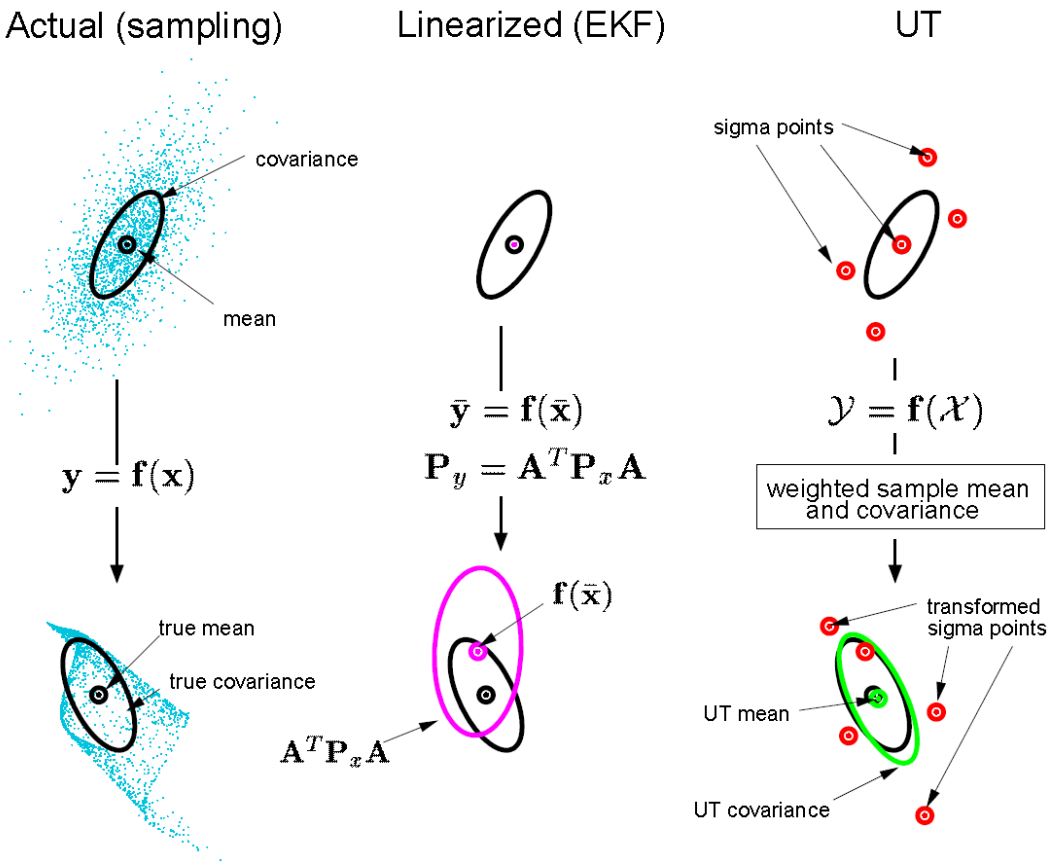

Figure 8.1: UT for mean and covariance propagation;
a) actual, b) first-order linearization (EKF), c) sigma-point. Image from [36].

extended Kalman filter) for nonlinear systems, but at the price of higher complexity and computational expense, [36].

One of the main advantages of the UKF is that it does not require the calculation of Jacobian matrices. In many applications, Jacobian matrix evaluation can be nontrivial and lead to implementation difficulties. However, in our application, the calculation of the Jacobian matrix is quite simple based on the equations of the system model. Therefore, the UKF does not provide us with any additional benefit in this case.

Time Complexity

We can measure code performance, for instance, with Matlab functions *tic* and *toc*, which gives us the execution time of the algorithms. *tic* starts a stopwatch timer, and the timer is stopped with the *toc* function. The output represents the time elapsed between the two commands, in seconds.

We ran 100 iterations of both the UKF and EKE algorithm, and stored the time-values we obtained. We then calculated the minimum value, the max value, and the average elapsed time for both filters. Table 8.1 displays the results. For the EKF the average elapsed time between *tic* and *toc* was 13.7029 seconds. For the UKF we got an average elapsed time of 16.6968 seconds. The difference in the average run time for the two filters is $2.9939 \approx 3$ seconds.

By recording running times for both algorithms, we get that the EKF is superior to the UKF, with an average time of almost 3 seconds. The overall number of computations, and codelines, in the UKF, are the same order as for the EKF, but the UKF algorithm takes significantly longer time, about 18%, to make an estimate, because it has to handle all the sigma points.

The different runtime reflect that the UKF has the highest computational cost, and the EKF is the most efficient algorithm.

Filter	**Min Time**	**Max Time**	**Average Time**
EKF	13.7206	17.6340	13.7029
UKF	16.6438	20.2028	16.6968

Table 8.1: Min, max and average elapsed time from running the observers 100 times each in Matlab

However, 3 seconds is not a crucial time for a ship prone to parametric roll. As we can see from Figure 8.2 it takes almost 350 seconds before the parametric roll is fully developed.

Method Complexity

Another issue is method complexity.

Our model for the ship, Eq. (2.17), is a high order (8 order), nonlinear system. In fact, at every time step the computational cost of calculating the Jacobian matrix

in the EKF is less than that of directly propagating the $2L+1$ sigma points through the nonlinear dynamics of the ship model in the UKF. In addition, the Cholesky decomposition should be computed at every time step when selecting the sigma points in the UKF. So, from the point of method complexity, the complexity of the UKF is higher than the EKF for our system. This conclusion can be verified by the runtime of the filters in Table 8.1.

The tuning of the error covariance matrix, Q, and the measurement noise matrix, R is difficult for both filters. The process noise covariance matrix Q is a 8×8 matrix and the measurement noise covariance matrix R is a 3×3 matrix. Assuming that the noise signals are not correlated, both Q and R are diagonal. There is no theoretical guidance on how to tune the matrices Q and R to obtain the best performance, so we can only perform a number of controlled simulations to find good values. It is found that the relationship between Q and R are the most important issue, regarding the result of the Kalman filter.

For the UKF, we also needed to set the α, β and κ parameters, to get the best possible effect in performance of the filter, for estimation of parametric roll. We have optimized these parameters by hand through considerable simulations, and they can be regarded as good values for the filter.

Dynamic performance

To compare the performances of state estimation based on the two Kalman filters - the EKF and the UKF - we conducted a series of contrastive simulations to determine which Kalman filter is the most preferable one for state estimation of a ship in parametric roll resonance condition.

Figure 8.2 - Figure 8.5, depicts the simulation of the 8 different states from Eq. (2.17) against their estimate, from both the EKF and UKF. As we can see the estimates are almost equal to the simulated values for both observers, except for state 3, 7 and 8, namely the ship forward speed, the wave induced forces, and the rate of the wave induced forces.

In the top plot in Figure 8.3 we see that the estimated speed from the EKF is much more influenced by noise than the one from the UKF.

Figure 8.5 depicts the plots of the simulated wave induced forces, and rate of the wave forces, compared to their estimates from both observers. From the plots it seems like the UKF is unable to estimate the wave induced forces and their rate. The EKF on the other hand, makes a fairly good estimate.

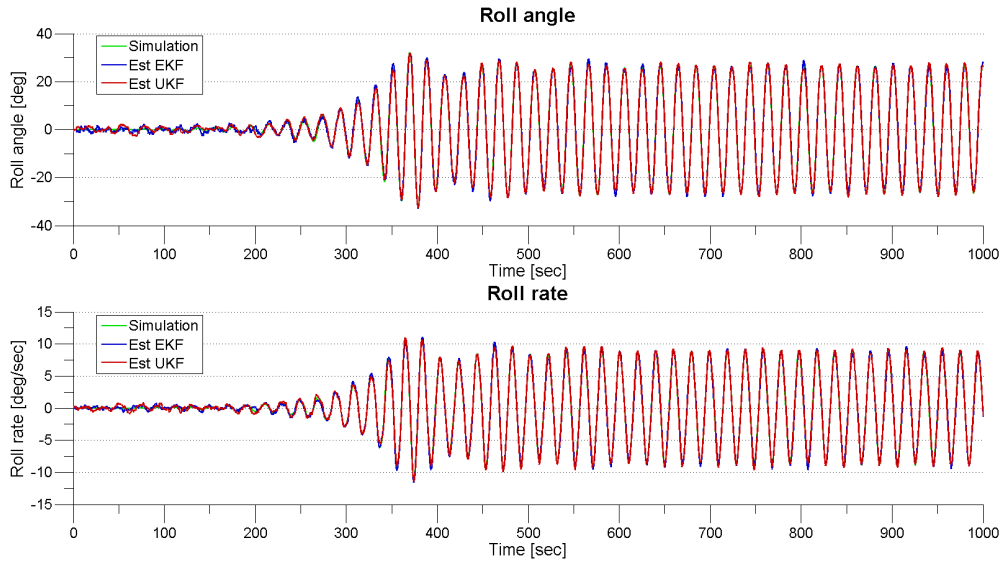

Figure 8.2: Simulated roll angle (top plot), and roll rate (bottom plot).

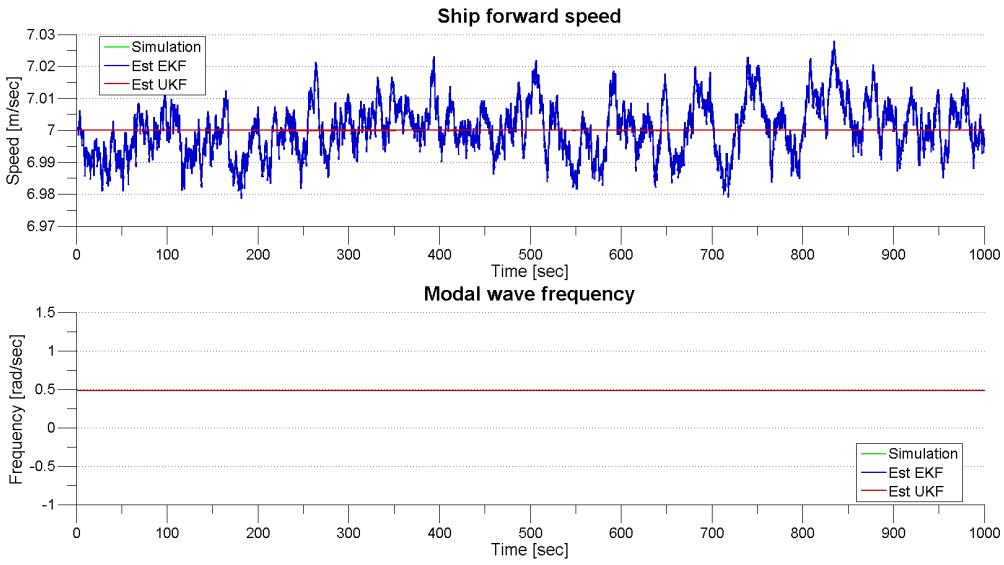

Figure 8.3: Simulated speed (top plot), and modal wave frequency (bottom plot).

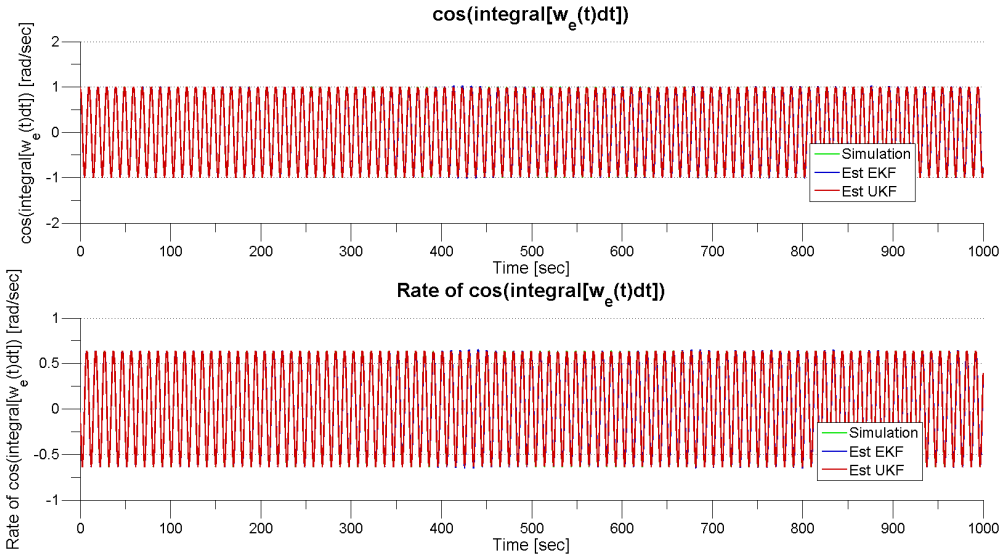

Figure 8.4: Simulated state 5 (top plot), and state 6 (bottom plot).

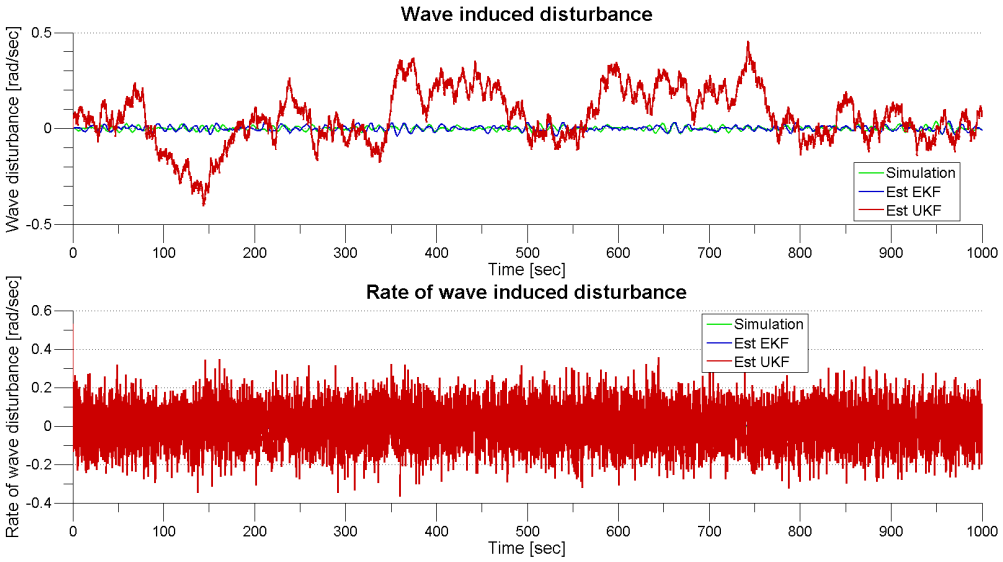

Figure 8.5: Simulated wave induced forces (top plot), and rate of wave forces (bottom plot).

Figure 8.6 - Figure 8.13, depicts the error between simulation value and the estimate from UKF, as well as the error between simulation value and the estimate from EKF, for the 8 different states from Eq. (2.17)

From Figure 8.6, which shows the error of simulated roll angle against its estimate from both filters, we see that the error are in the same order. The error from UKF is a bit smaller in amplitude, and seems less affected by noise.

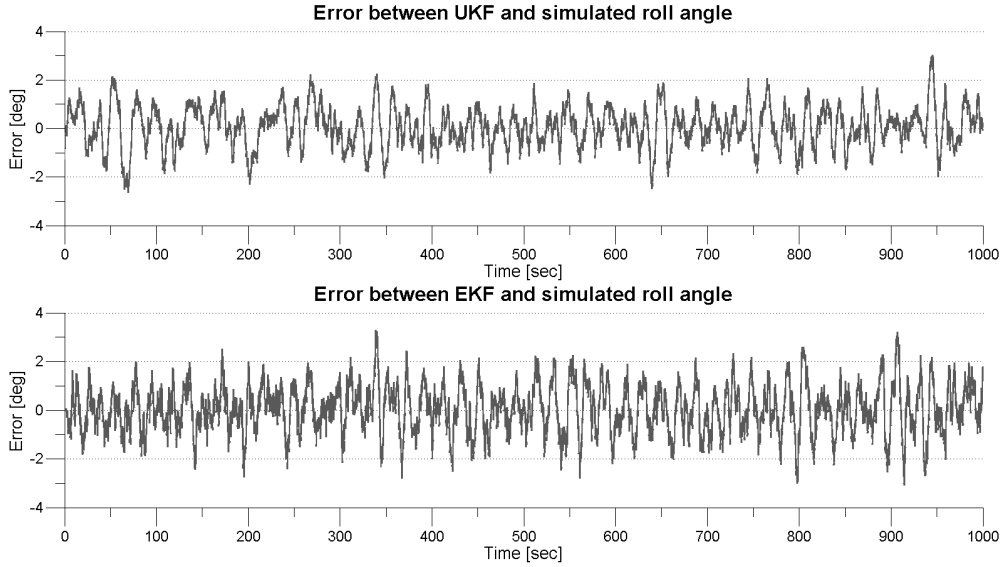

Figure 8.6: Roll angle error from UKF (top plot), and EKF (bottom plot).

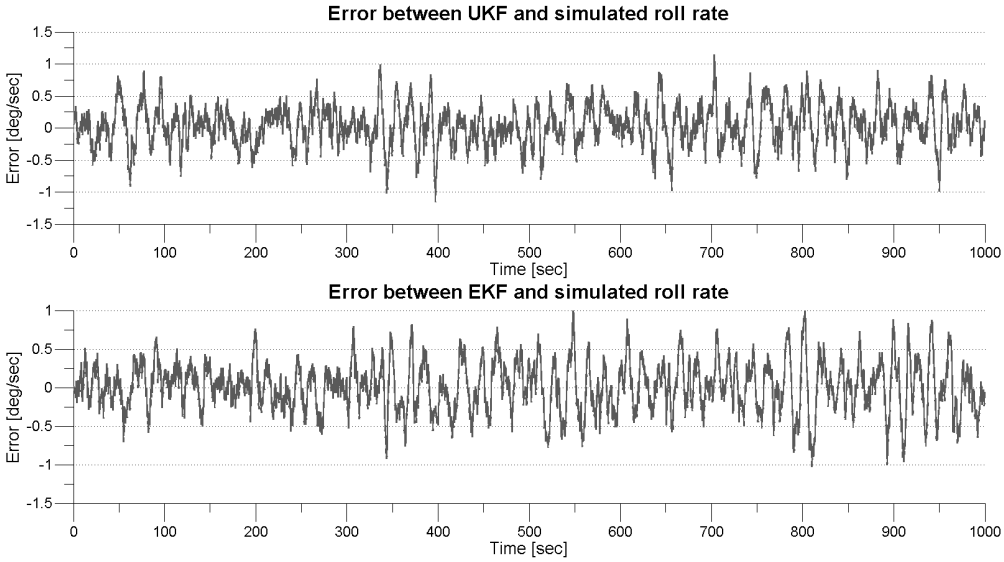

Figure 8.7: Roll rate error from UKF (top plot), and EKF (bottom plot).

Figure 8.7 displays the simulated roll rate compared to its estimate from UKF and EKF, respectively. Here we can see that the error plots are almost the same regarding amplitude, and noise impact.

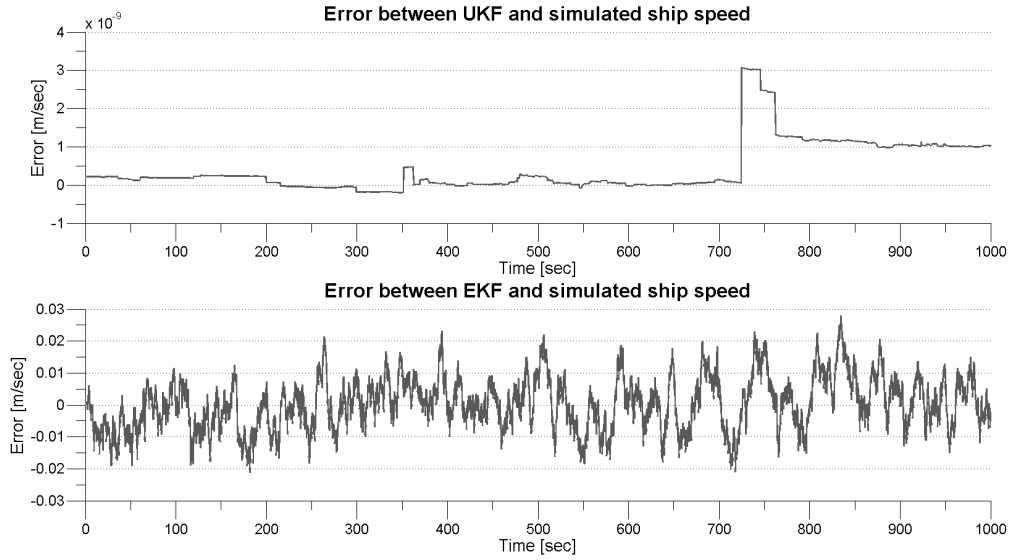

Figure 8.8: Ship speed error from UKF (top plot), and EKF (bottom plot).

For the error plots from estimation of speed, in Figure 8.8, we see that the UKF has an error of 10^{-9}, while the EKF estimate error is in an order of 10^{-2}. Error plots for the modal wave frequency are both equal to zero, see Figure 8.9.

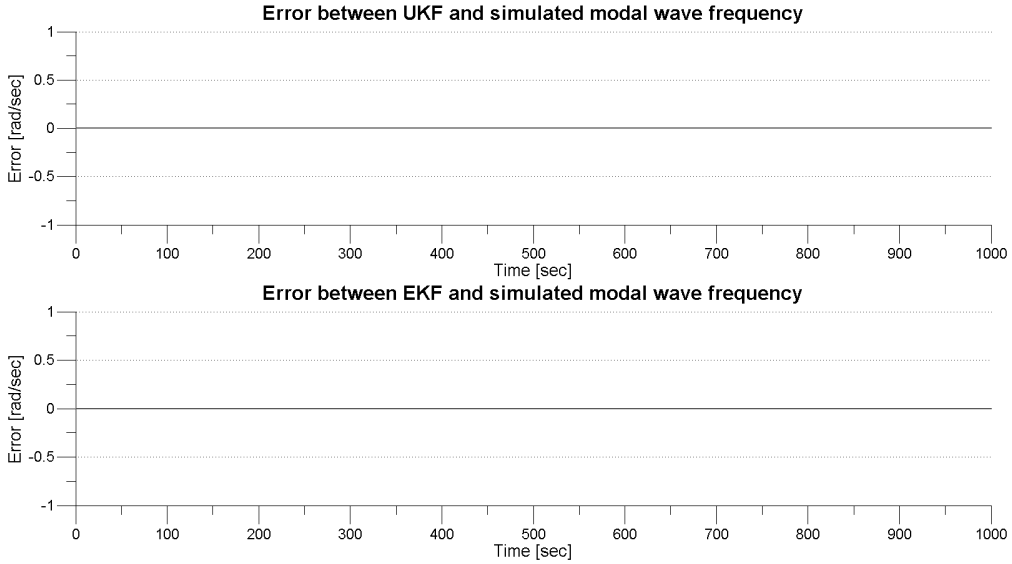

Figure 8.9: Modal wave frequency error from UKF (top plot), and EKF (bottom plot).

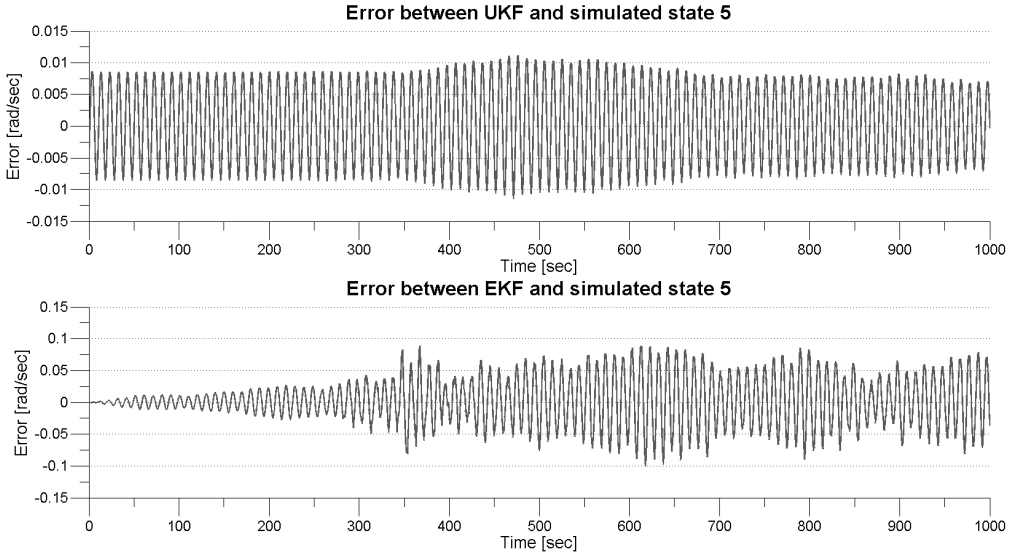

Figure 8.10: State 5 error from UKF (top plot), and EKF (bottom plot).

Figure 8.10 and Figure 8.11, shows plot of error between simulation value and estimate, for the cosinus term with the encounter frequency, and the rate of this cos term, respectively. We can see that the error from the UKF is more or less constant, while the error from the EKF increases in amplitude from the start of the simulation till 250 seconds of the simulation has been conducted.

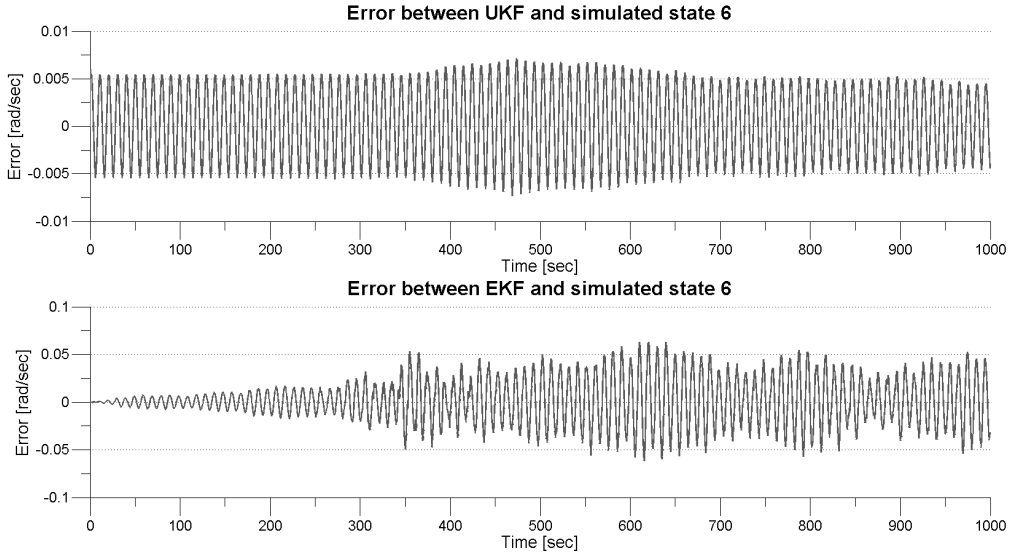

Figure 8.11: State 6 error from UKF (top plot), and EKF (bottom plot).

Max value for the error for state 5, from carrying out a simulation by help of the UKF, is just above $0.01[rad/sec]$, while the peak for the error from running the EKF is approximately $0.08[rad/sec]$.

For state 6 the error between simulated value, and estimate, from the unscented Kalman filter, is maximum $0.0073[rad/sec]$, while the biggest error for state 6, from the extended Kalman filter, is $0.0631[rad/sec]$.

The plot for error between simulated and estimated wave induced disturbance can be seen in Figure 8.12. From the figure it is evident that the error from running the UKF is larger than for the EKF. As mentioned before, it seems like the UKF has trouble estimating the wave induced forces. The UKF implemented may be less rigorous, due to the fact that it treats the noise as additive, as seen in Eqs. (6.38) and (6.41). This can also possibly be seen in Figure 8.13. We can see that the error from the UKF is almost constant, is very much affected by noise, and has a much larger amplitude, than the error for the same case, from the EKF.

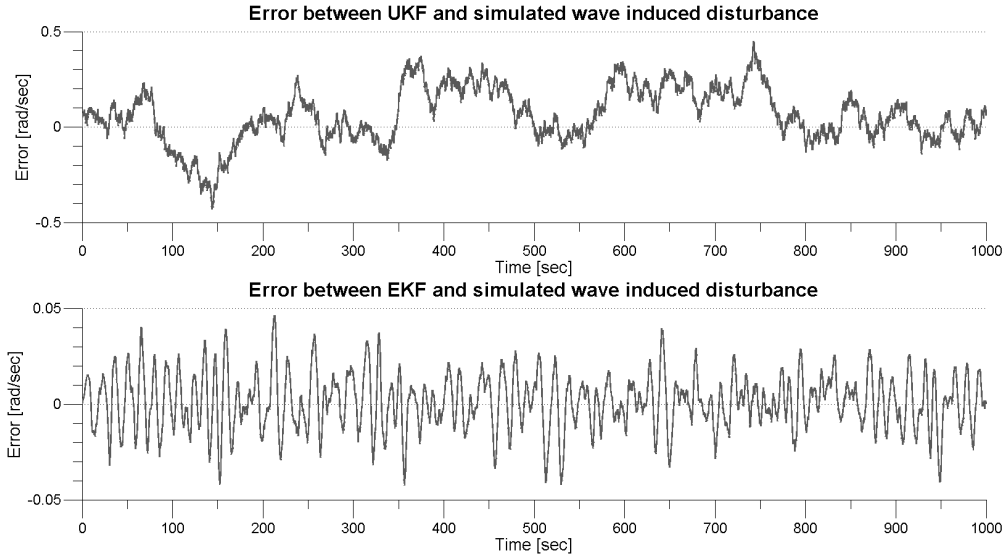

Figure 8.12: Wave induced disturbance error from UKF (top plot), and EKF (bottom plot).

However, the wave induced forces is not the most important states for us to estimate. Regarding parametric roll for ships, estimation of the wave encounter frequency, and the modal wave frequency, is much more important. The roll angle, and roll rate, is also more important states, than the wave induced forces, and the rate of them.

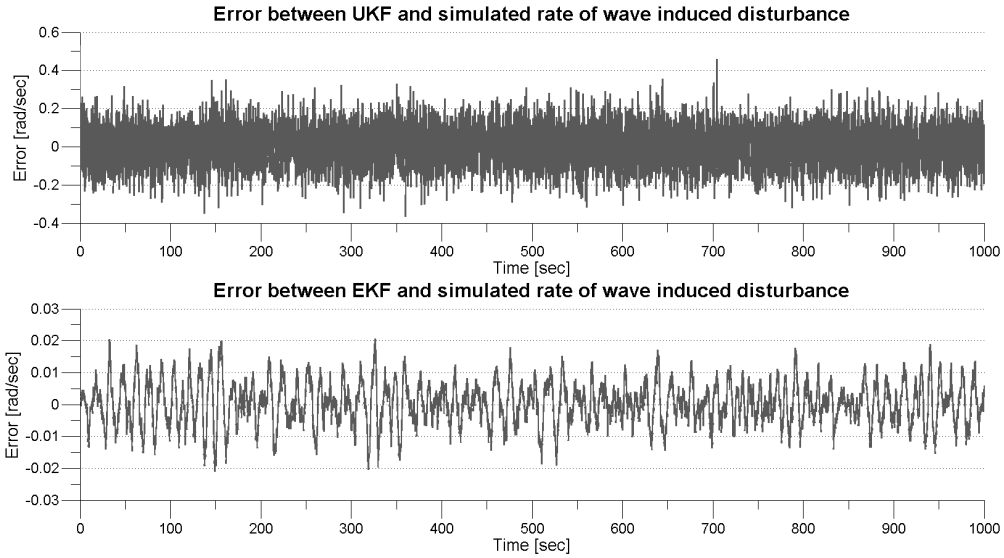

Figure 8.13: Rate of wave induced disturbance error from UKF (top plot), and EKF (bottom plot).

To determine how well the EKF and UKF algorithms are performing, we need comparison data. After running both filters, and plotting the error for each of the states, the average error for both filters were calculated. The values we obtained are gathered in Table 8.2. From the table we can see that the UKF has a better average error for state 1, 2, 3, 5, and 6, wile the EKF has a better average error for state 2, 7 and 8. The error for state 2, namely the roll rate, is almost equal for the two state estimators. The EKF is better than the UKF for estimation of state 7, and 8, as it have been mentioned earlier. Aside from the estimates for the wave induced forces, the UKF has a much better average error than the EKF.

Average Error			
State	**UKF**	**EKF**	
x_1	0.0127	0.0133	$[rad]$
x_2	0.0048	0.0043	$[rad/sec]$
x_3	4.4830×10^{-10}	0.0067	$[m/sec]$
x_4	0	0	$[rad/sec]$
x_5	0.0055	0.0298	$[rad/sec]$
x_6	0.0035	0.0191	$[rad/sec]$
x_7	0.1218	0.0115	$[rad/sec]$
x_8	0.0795	0.0054	$[rad/sec]$

Table 8.2: Average error, UKF and EKF

Furthermore, we also calculated the standard deviation of the estimation errors as in Eq. (8.1). The standard deviation shows how much variation or "dispersion" exists from the average mean.

The standard deviation of a random variable, statistical population, data set, or probability distribution is the square root of its variance.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}} \tag{8.1}$$

The results are given in Table 8.3.

Std Dev of Estimation Errors			
State	**UKF**	**EKF**	
x_1	0.5701	0.58039	$[rad]$
x_2	0.22335	0.19503	$[rad/sec]$
x_3	1.377×10^{-7}	0.26556	$[m/sec]$
x_4	0	0	$[rad/sec]$
x_5	0.16453	1.3914	$[rad/sec]$
x_6	0.10521	0.89147	$[rad/sec]$
x_7	5.652	0.48266	$[rad/sec]$
x_8	3.507	0.23455	$[rad/sec]$

Table 8.3: Standard deviation of estimation errors, UKF and EKF

We can see from Table 8.3 that, for states 1 till 6, the UKF get the lowest standard deviation of estimation errors. This indicates that the data points tend to be very close to the mean. Whereas the higher standard deviations, we get for the EKF, indicates that the data points are spread out over a larger range of values. For state 7, and 8, the EKF has the lowest standard deviations.

8.2 Conclusion Regarding Comparison of the Observers

Two algorithms, the EKF and the UKF, have been investigated in detail by considerable simulations for state estimation of a ship in parametric roll resonance condition.

Theoretically the UKF has several advantages over the EKF. First, the UKF does not need to calculate the Jacobian matrix. Second, the UKF uses a deterministic scheme to choose the sigma points.

On the other hand UKF increase the method complexity and computational cost, and make the practical implementation more difficult. One reason for this is that the order of the system is relatively high, which makes the EKF computationally more efficient than the UKF.

It is shown that for this specific object most of the estimates, and the standard deviations, from the UKF are slightly less biased and less than those of the EKF, respectively. The UKF is less noise sensitive, and has more accuracy for the nonlinear model, and measurement model, than the EKF.

For the situation at hand, the UKF does not display its outstanding advantages, as in other fields. Nevertheless, results suggest that the UKF has improved estimation performance compared to the EKF.

Part IV

Wave Direction

Chapter 9

Direction of the Incoming Waves

In this chapter we investigate if it, in addition to estimating the frequency of encounter, is possible to also estimate the direction of the incoming waves. We make a new augmented state-space model, implement an extended Kalman filter for estimation of wave direction, and do a simulation study.

9.1 Surge-Roll Mathematical Model

In Chapter 2, the mathematical model for parametric roll were presented, both the standard equations, Sec. 2.1.1, and the model in state-space form, Eq. (2.17). In Sec. 2.1.1 the standard model for parametric roll is given, and Figure. 2.1 depicts the heading angle of the ship.

9.1.1 Forward Speed

Since the sway motion is neglected, the time-varying forward speed $U(t)$ is approximated as

$$U(t) = \sqrt{u^2 + v^2} \approx u, \tag{9.1}$$

where u and v are the surge and sway velocities, respectively.

9.1.2 Heading Angle

The sailing condition of a vessel is given by its forward speed U and its encounter angle, i.e., the heading angle relative to the waves.

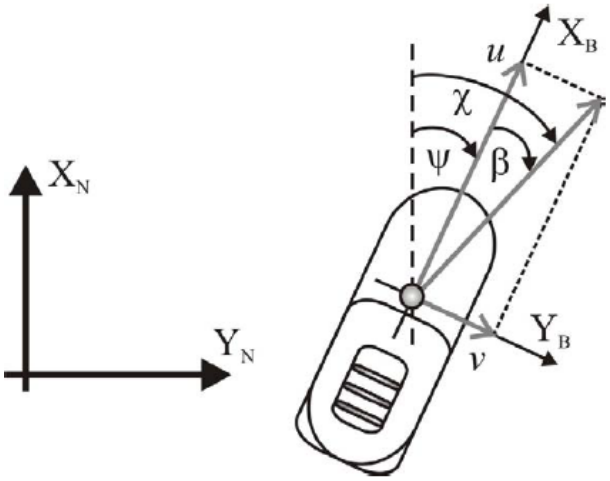

Figure 9.1: The heading angle ψ represents the orientation of the BODY frame relative to the NED frame, the course angle χ represents the orientation of the vessel velocity vector relative to the NED frame, while β signifies the difference between the course and the heading, [6].

To describe the motion of a ship, two reference frames are used, a local geographical Earth-fixed frame and a body-fixed frame, which is attached to the vessel, see Appendix B. Figure 9.1 illustrates that the direction of the vessel velocity vector is generally not equal to the direction of the vessel heading. The relationship between the course angle χ and the heading angle ψ is defined as , [7],

$$\chi = \psi + \beta \tag{9.2}$$

If the waves are observed from a reference frame that moves at a constant speed, the frequency observed is the encounter frequency. The wave encounter frequency ω_e is related to the variations of the ship speed and heading angle by

$$\omega_e = \omega_0 - \frac{\omega_0^2}{g} U(t) \cos(\beta_w^n - \psi) \tag{9.3}$$

In Eq. (9.3), ψ is the heading angle. The relation between the wave encounter

angle β_w and the wave angle, expressed in the NED reference frame, β_w^n is given by $\beta_w = \beta_w^n - \psi$.

The dynamics of the encounter frequency can be established as

$$\dot{\omega_e} = -\frac{\omega_0^2}{g}\dot{U}(t)\cos\left(\beta_w^n - \psi\right) - \frac{\omega_0^2}{g}U(t)\sin(\beta_w^n - \psi)\dot{\psi} \tag{9.4}$$

where the kinematic equation, [17],

$$\dot{\psi} = \cos(\phi)r \tag{9.5}$$

must be augmented to the system model.

9.2 The Nomoto Model for Ship Heading Response

9.2.1 Second-Order Nomoto Model (Yaw Subsystem)

A model for vessel dynamics can be expressed as, [18],

$$\dot{\eta} = R(\psi)\nu, \tag{9.6}$$

$$(M_{RB} + M_A)\dot{\nu} + C_{RB}(\nu)\nu + d(V_{rc}, \gamma_c) = \tau_{control} + \tau_{wind} + \tau_{waves}. \tag{9.7}$$

where M_{RB} is a positive-definite, rigid-body mass matrix, M_A is a positive-definite hydrodynamic added mass matrix, C_{RB} is a skew-symmetric Coriolis-centripetal matrix, $C_{RB}(\nu)\nu$ is Coriolis-centripetal terms, $d(V_{rc}, \gamma_c)$ represents the current and damping forces, and $\tau_{control}$, τ_{wind}, and τ_{waves} are vectors of forces due to control, wind, and waves, respectively. For further details about kinematic models used in marine control systems and transformations, see [45].

Model-based heading controllers for marine craft are usually based on the model representation of Nomoto *et al.*, [40]. Autopilots are used to correct deviations from a desired equilibrium heading, and thus a linear model is sufficient for control design, [13]. A linear autopilot model for heading control can be derived from the manoeuvring model by choosing the yaw rate r as output:

$$r = c^\top \nu, \qquad c^\top = [0, 1] \tag{9.8}$$

If the roll mode is neglected, we have

$$\frac{r}{\delta}(S) = \frac{K(1+T_3S)}{(1+T_1S)(1+T_2S)} \tag{9.9}$$

Eq. (9.9) is referred to as *Nomoto's second-order model*, [40], where K is the static yaw rate gain, and T_1, T_2 and T_3 are time constants. The zero term $(1+T_3S)$ and the high frequency pole term $(1+T_2S)$ are due to the coupling effect from the sway mode.

9.2.2 First-Order Nomoto Model (Yaw Subsystem)

In practice, because the pole term $(1+T_2S)$ and the zero term $(1+T_3S)$ in Eq. (9.9) nearly cancel each other, a further simplification on Eq. (9.9) can be done, by defining the *equivalent time constant:*

$$T := T_1 + T_2 - T_3 \tag{9.10}$$

With $\dot{\psi} = r$, the first-order Nomoto model, [40], yields

$$\frac{r}{\delta}(S) = \frac{K}{1+TS} \tag{9.11}$$

Eq. (9.11) can be expressed in the time domain as

$$T\dot{r} + r = K\delta \quad \xrightarrow{\dot{\psi}=r} \quad T\ddot{\psi} + \dot{\psi} = K\delta \tag{9.12}$$

9.2.3 Parameters for the Nomoto Model

The time constant and low-frequency gain from Eq. (9.11) are given by

$$T = \frac{I_z - N_{\dot{r}}}{-N_r}, \quad K = -\frac{N_\delta}{N_r} \tag{9.13}$$

which can be estimated from trials in calm water, [17]. Numerical values of the parameters in Eq. (9.9) are given by a Mariner class cargo ship, [17], and can be seen in Table. 9.1.

Parameter	$K(1/s)$	$T_1(s)$	$T_2(s)$	$T_3(s)$	T
Value	0.185	118.0	7.8	18.5	107.3

Table 9.1: Nomoto Model Parameters for a Mariner Class Cargo Ship

9.3 Heading Sensors

The main sensor components for a heading controlled marine craft are, [17]:

- Magnetic and/or gyroscopic compasses measuring ψ
- Yaw rate gyro measuring r

Compass
A compass is a navigational instrument that measures directions in a frame of reference that is stationary relative to the surface of the earth. Compasses may operate on magnetic or gyroscopic principles or by determining the direction of the Sun or a star. Magnetic and gyroscopic compasses are the primary devices onboard commercial ships and rigs, [17].

Inertial Measurement Systems
Today inertial measurement technology is available for commercial users thanks to a significant reduction in price the last decade, [17].

An inertial measurement unit (IMU) consists of an inertial sensor assembly (ISA), hardware to interface the ISA, and low level software that performs down-sampling, temperature calibration, and vibration compensation. The ISA is a cluster of three gyros and three accelerometers that measure angular velocity and linear acceleration, respectively.

Our ship model is in 1 DoF. For this case the compass measurement is taken as the sum of the low-frequency (LF), and wave frequency (WF) signals, [17]:

$$y_1 = \psi + \psi_w \tag{9.14}$$

where ψ is the response due to control action and LF disturbance, and ψ_w represents the first-order wave-induced motion.

And the measured yaw rate can be decomposed into, [17],

$$y_2 = r + r_w \tag{9.15}$$

where r is the LF signals, and r_w represents the WF signals.

Wave Filter
In its simplest form the first-order wave-induced motion components ψ_w and r_w are filtered out from the measurements in Eq. (9.14) and (9.15), and consequently prevented from entering the feedback loop. This is known as *wave filtering*, where the output of the filter is the LF motion components ψ and r. This is necessary to avoid excessive rudder action.

9.4 Augmented State Space Model for Estimation of Wave Direction

To be able to investigate which direction incoming waves come from we need another state in the ship model from Eq. (2.17), namely the encounter angle

$$x_9 = \beta_w^n \tag{9.16}$$

To be able to add the subsystem to our existing model, Eq. (2.17), we need the derivative of state x_9

$$\dot{\beta_w^n} = 0 \tag{9.17}$$

State 9 can be written in state-space as,

$$\dot{x_9} = 0 \tag{9.18}$$

We assume that a yaw rate gyro is available for yaw rate feedback, and a compass measures the heading angle, which is needed for feedback. We use the measured signals in the observer. The measured *psi* and r enter the model as inputs. Heading angle ψ and yaw rate r will enter the other equations as time-varying signals.

With 9 states in the system, the resulting state-space model becomes

$$\dot{x} = f(t, x, u) \tag{9.19}$$
$$y = h(t, x, u), \tag{9.20}$$

where

$$\begin{aligned} x = [\phi \quad \frac{d}{dt}\phi \quad u \quad \omega_0 \quad \cos\left(\int_0^t \omega_e(\tau)d\tau\right) \ldots \\ \frac{d}{dt}\cos\left(\int_0^t \omega_e(\tau)d\tau\right) \quad h_{wave}(s) \quad \frac{d}{dt}h_{wave}(s) \quad \beta_w^n]^\top, \end{aligned} \tag{9.21}$$

$$u = [\tau_1 \quad \psi \quad r]^\top \tag{9.22}$$

and

$$\dot{x} = \begin{bmatrix} x_2 \\ \frac{B_{44}}{I_x+A_{44}}x_2 - \frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi^3}}{I_x+A_{44}}x_1^3 \\ \frac{u_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}|x_3|x_3 \\ 0 \\ x_6 \\ -x_5\left(x_4 - \frac{x_4^2}{g}x_3\cos(x_9-u_2)\right)^2 \\ \ldots - \frac{\dot{x}_3 x_4 x_6}{(g-x_3x_4\cos(x_9-u_2))}(\cos(x_9-u_2)) \\ \ldots - \frac{\cos(x_1)x_3x_4x_6u_3}{(g-x_3x_4\cos(x_9-u_2))}(\sin(x_9-u_2)) \\ x_8 \\ -\omega_0^2 x_7 - 2\lambda\omega_0 x_8 + K_w w \\ 0 \end{bmatrix} \tag{9.23}$$

$$y = \begin{bmatrix} x_1 & x_2 & x_3 + x_8 \end{bmatrix}^\top \tag{9.24}$$

The equation for $\dot{x_6}$ has also been updated, and now has a coupling to state x_9, and measurements ψ and r, (input u_2 and u_3, respectively).

9.5 Implementation of EKF for Wave Direction Estimation

Since the system is observable, see Sec. 3.3.2 the state vector, $\mathbf{x}$, can be reconstructed recursively through the measurement vector, $\mathbf{y}$, and the control input vector, $\mathbf{u}$. Thus, we can make an observer for the system.

An extended Kalman filter were implemented for estimation of the direction of the incoming waves, and simulated. The Matlab code files are given in Appendix G. We get the parameter values that we need for the model by running the file INIT.M.

Box-Muller transformation is used for noise generation.

We implemented a speed and heading controller, see Chapter 4. The initial setpoint for the speed is $7.5[m/s]$.

For each time instant and system state, the function F_WAVE_DIR.M generates

the instantaneous value of $\dot{x}$ for the container ship, from the state space model in Sec. 9.4. Numerically integrating with a 4th-order Runge-Kutta method, with the fixed time-step $h = 0.1$, the states x are calculated for any given time instant.

The input vector model is given as in Eq. (9.22). The heading angle ψ, an yaw rate r were modeled with a nomoto model, see Eq.(9.12), and integrated using Runge-Kutta's 4th-order method.

As a consequence of augmenting the model, we needed to calculate a new Jacobian matrix. This was done in the mathematics software suite Maple, and the $[9 \times 9]$ Jacobian matrix is given in Appendix E.

9.6 Simulation Study for Wave Direction Estimation

The estimation of wave direction was simulated for three different cases,

Case 1: Head sea condition, i.e. $\beta_w^n = \pi$, starting with $\psi = 0$.

Case 2: Change in the wave direction after some time, i.e. β_w^n slowly time-varying.

Case 3: Course change, i.e. change the heading from 0 to 20 degrees after some time (and keep β_w^n constant).

Case 1
The ship is in head sea condition. The wave encounter angle β_w^n is equal π. Without loss of generality it is assumed that the ship is moving northwards and therefore $\psi = 0$.

In Figure 9.2 the direction of the incoming waves (the top plot) from the simulation (red line), and the estimation (blue line), as well as the difference between simulated and estimated value (the bottom plot), is shown. The actual wave direction β_w^n (red line) is constant, equal to π. The error plot, bottom part of Figure 9.2, denotes the ability of the observer to estimate the direction of the incoming waves.

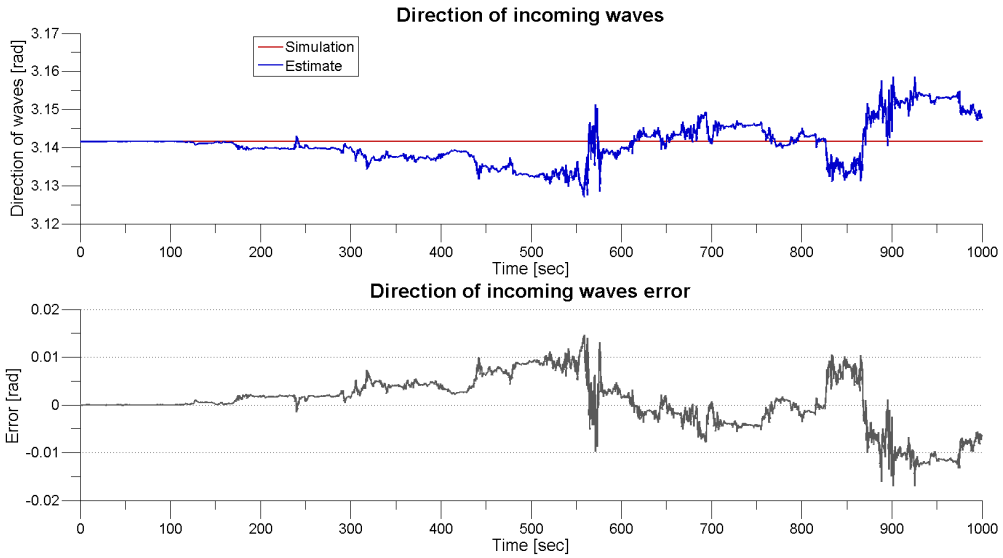

Figure 9.2: Direction of incoming waves, Case 1.

Case 2
We try with a slowly time-varying β_w^n. This would imply a change in the wave direction.

Figure 9.3 shows the direction of the incoming waves, and the error plot. The red line in the top plot in Figure 9.3 represents the simulated value. We can see that it varies slowly with time. The blue line in the top plot in Figure 9.3 is the estimate of the time-varying wave direction.

In the bottom plot we can see that the difference between actual wave direction, and the estimate is maximum about $0.05[rad] \approx 2.86[deg]$.

Case 3
The ship makes a course change. To simulate this we change the heading from 0 to 20 degrees after some time, while the β_w^n is kept constant.

Figure 9.4 denotes the estimation of the direction of the incoming waves from simulation of Case 3. In the top plot in Figure 9.4, the estimate (blue line) follows the "real" value (red line) with small deviations. The bottom plot in Figure 9.4 shows the error plot for Case 3. From this we can see that the EKF filter we have implemented have no problem estimating the direction of the incoming waves when we have a course change, and keep β_w^n constant.

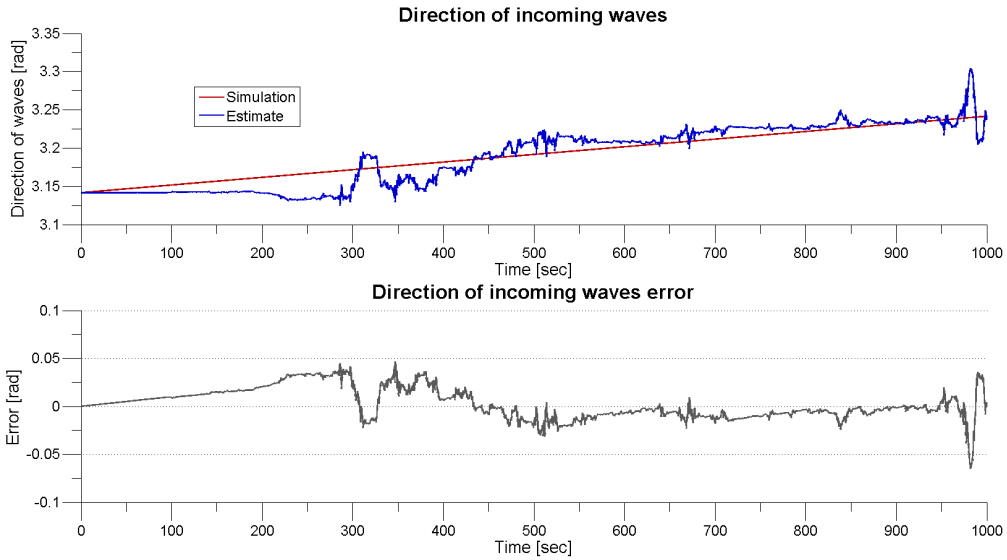

Figure 9.3: Direction of incoming waves, Case 2.

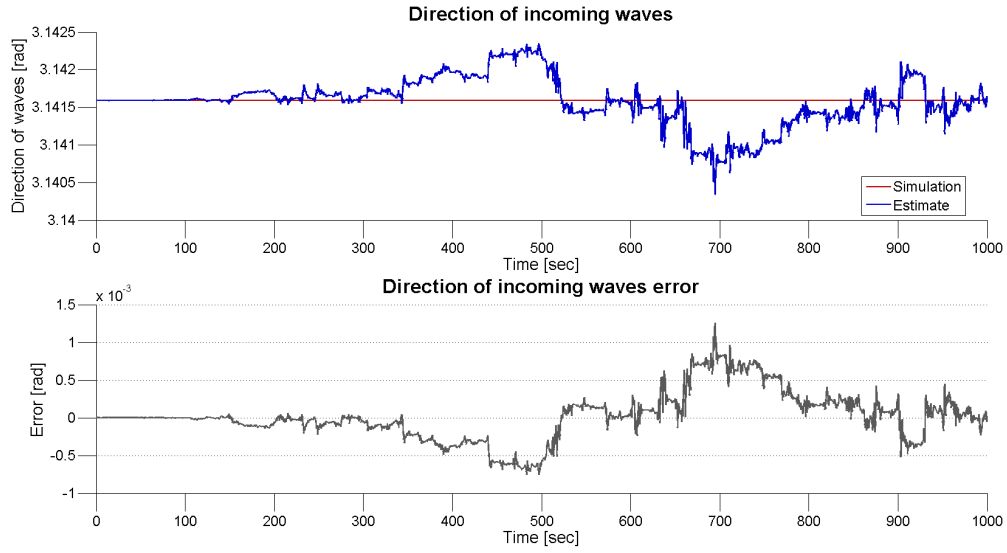

Figure 9.4: Direction of incoming waves, Case 3.

Part V

Closing Remarks

Chapter 10

Discussion/Summary

The subject for this thesis has been to investigate different nonlinear observers for estimation of wave encounter frequency, and in addition to estimate the direction of the incoming waves.

The mathematical model presented in Chapter 2 is a simplified model of a ship, for parametric rolling. The roll motion has been modeled as an uncoupled Mathieu type equation. A one degree of freedom roll motion equation gives us much faster computational speed, but single DoF models have too little complexity to describe an auto-parametric system, since the roll motion for a ship sailing in longitudinal seas represents only the secondary system. Another shortcoming of the simplistic model is the disadvantage that the parameters are computed only for certain fixed sea conditions, and forward speed. The parameters would therefore have to be re-computed if you change any of these. But, nevertheless, they are useful to obtain insight in the parametric roll resonance phenomenon.

The test of a system's observability is a necessary prerequisite to the estimation of states and parameters from the output of the system. In general stability and robustness conditions for nonlinear systems are weaker than those for linear systems. Properties for nonlinear systems tend to be local. In Chapter 3, we found an observability matrix, of full rank, for both the system with 8 states, and the augmented system with 9 states, making them locally observable.

The results from running the Simulink model, The_System.mdl, together with the M-files, Init_sim.m, compute_tau.m, and compute_x_dot.m, showed that the system equations do resonate in roll.

The input to the system, the propeller thrust, τ_1, is calculated through a feedback

linearization controller. Although feedback linearization provides a simple method to design controllers for nonlinear systems, the resulting controllers are sometimes very sensitive to measurement noise because it prevents a perfect cancellation of the nonlinearities. But we assume that the model is explicitly known, and that we have a structural property of our nonlinear system that allows us to perform such a cancellation.

Waves can be accurately approximated by 2nd order linear wave theory. 2nd order wave-induced forces are slowly varying forces. When a state observer is applied, estimates of the wave-induced forces can also be computed. The results from running the Simulink model, showed that the system equations resonates in roll when the state equations for the wave dynamics were added to the system.

From Figure 7.1 to Figure 7.8, we see that the EKF can produce almost noise-free estimates of the system states, even with the measurements exposed to noise, and the process contaminated by white noise.

From Figure 7.12 to Figure 7.19, it is seen that good state estimates are obtained from the UKF even though the measurement are highly noise-corrupted.

The proposed observer methods have shown very good capabilities in estimating wave encounter frequency, for a ship exhibiting parametric roll resonance, but it is important to point out that they worked only for constant modal wave frequency, and the experiments were run in head seas condition.

In Chapter 8, we made a comparison of the observers. To illustrate the different aspects of the two observers, theory, time complexity, method complexity and dynamic performance were evaluated.

To be able to estimate the direction of the incoming waves, we made an augmented state space model, and designed a heading controller. We implemented a new extended Kalman filter for estimation of the wave direction, and simulated it for three different cases. The observer was able to estimate the wave direction for all three cases.

Chapter 11

Conclusions and Future Work

This chapter lists the main conclusions drawn based on the thesis work. It also lists some possible avenues for future work.

11.1 Conclusions

This thesis investigated different observers - EKF and UKF - for estimation of the frequency of encounter for a ship exhibiting parametric roll. It also included estimation of the direction of the incoming waves.

A simplistic mathematical model, suitable for simulations, is presented for parametric roll resonance, in head seas.

By using Lie derivatives, we concluded that the nonlinear system equations are locally observable.

The state-space description of the ship model, with input τ_1 (the propeller thrust) computed through a feedback linearization controller, was successfully reproduced in Matlab/Simulink, using data from a specific 281 m container ship. We conclude that the simulation indicate that the model based on the simplified roll equation is adequate to describe the ships' dynamics in parametric roll resonance.

A design for a state observer for the system has been proposed, in the form of an extended Kalman filter. The filter included the effects of wave characteristics with a linear second order system. The extended Kalman filter is able to estimate both the encounter frequency, and a constant wave spectrum modal frequency. On the

whole the filter produce good estimates from the noise-contaminated states.

An unscented Kalman filter is implemented, for estimation of encounter frequency. Like the extended Kalman filter, the unscented Kalman filter is able to estimate both the encounter frequency, and a constant wave spectrum modal frequency. It produce almost noise-free estimates.

The two filters were compared against each other, and in conclusion the UKF has improved estimation performance compared to the EKF.

It is possible to estimate the direction of the incoming waves. From a simulation study with an augmented state-space model for wave direction estimation, we conclude that the nonlinear observer implemented, in the form of an EKF, make good estimates of the wave direction.

11.2 Future Work

The time one has to work an a master thesis is limited, and not all that could be done can be done in the time allotted. Thus, as always, there are several avenues that are open to further research. I list here some possibilities:

- Develop good techniques for tuning of filter parameters, and design matrices.
- There is a lot of room for development in the area of unscented filtering. These include UKF stability properties, unscented smoothing, robust unscented filtering, and others.
- Estimation of encounter frequency with other nonlinear filtering techniques.
- Future work includes additional characterization of performance benefits of different observers opposed to each other.
- Developing on-line detection schemes, to predict parametric roll resonance, with aid of nonlinear observers.
- Develop controllers or anti-rolling devices with observers based on the 1 DoF mathematical model for parametric roll.

Appendices

Appendix A

Main Characteristics of The Container Ship

Quantity	**Symbol**	**Value**	
Length between perpendiculars	L_{pp}	281	m
Beam amidships	B	32.36	m
Draught amidships	T	11.75	m
Displacement	∇	76468	m^3
Roll radius of gyration	r_x	12.23	m
Transverse metacentric height	GM_t	1.84	m

Table A.1: Main Characteristics of The Container Ship

Appendix B

Reference Frames

In this appendix the most common coordinate frames used in navigation is described. More details about coordinate frames, time frames and transformations can be found in [26].

Earth Centered Reference Frames

ECI
The origin of the Earth Centered Inertial (ECI) frame $\{i\} = (x_i, y_i, z_i)$ is coincident with the center of the Earth. The x-axis points towards the *vernal equinox*, the z-axis points along the Earth's rotation axis at some initial time, and the y-axis completes the right handed orthogonal coordinate system.

ECEF
The Earth Centered Earth Fixed (ECEF) frame $\{e\} = (x_e, y_e, z_e)$ also has its origin in the center of the Earth. The x-axis points towards the intersection of the 0° longitude (Greenwich meridian) and 0° latitude (Equator). The z-axis points along the Earth's rotation axis, and the y-axis completes the right handed orthogonal coordinate system. The ECEF frame rotates relative to the ECI frame with the Earth rotation rate, ω_e, see Figure. B.1.

Geographic Reference Frames

NED
The North East Down (NED) coordinate system $\{n\} = (x_n, y_n, z_n)$ is defined

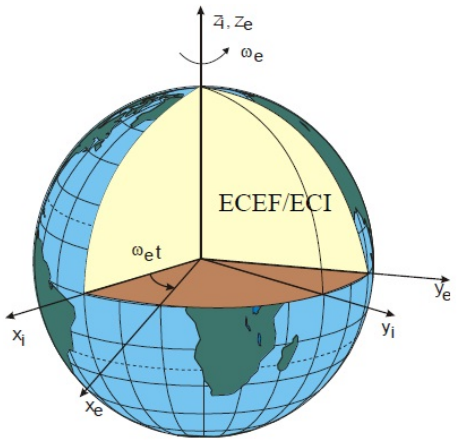

Figure B.1: The ECEF frame rotating with angular rate ω_e with respect to an ECI frame. Image from [56].

relative to the Earth's reference ellipsoid. The x-axis points towards true north. The z-axis points downward perpendicular to the tangent plane of the ellipsoid. The y-axis points towards east to complete the right handed orthogonal coordinate system.

Vehicle Reference Frames

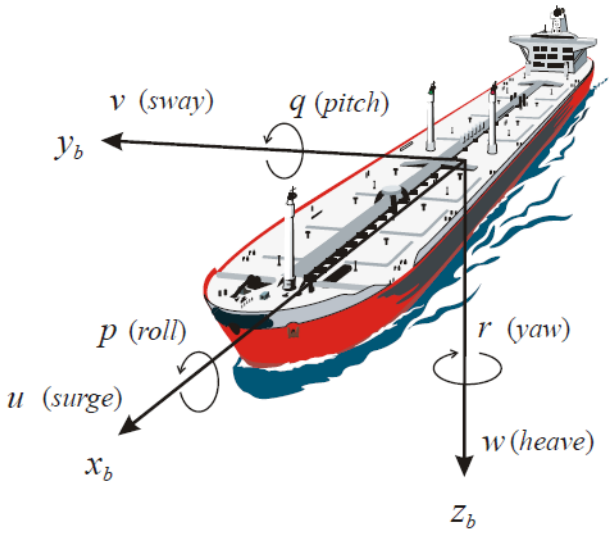

Figure B.2: The 6 DoF velocities u, v, w, p, q and r in the body-fixed reference frame $\{b\} = (x_b, y_b, z_b)$. Image from [17].

BODY

The body-fixed reference frame $\{b\} = (x_b, y_b, z_b)$ is a moving coordinate frame that is fixed to the craft. The origin of the BODY frame coincides with the origin of the NED frame, and the frame is moving and rotating with the vehicle, It is related to the NED frame through the Euler angles roll, pitch and yaw. For marine craft, 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. B.2):

- 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)

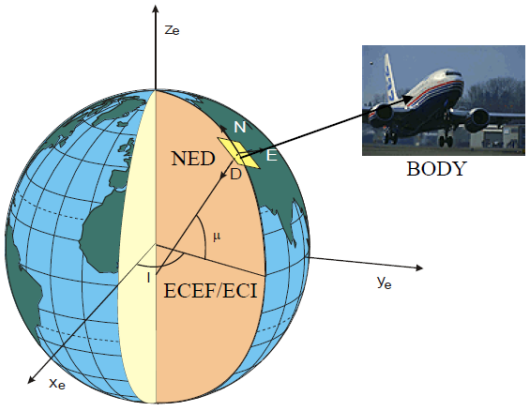

Figure B.3: The different frames relative to each other. Image from [17].

Figure. B.3 shows the different reference frames, described in this appendix, in relation to each other.

When dealing with several reference and coordinate frames it is necessary to know which frame vectors are decomposed into. Table. B.1, sums the notation used in the different coordinate systems described above.

Index	**Coordinate system**	**Components**
i	ECI	x_i, y_i, z_i
e	ECEF Cartesian	x_e, y_e, z_e
n	NED	x_n, y_n, z_n
b	BODY	x_b, y_b, z_b

Table B.1: Coordinate System Indexes

Appendix C

Wave Spectra

In 1968-1969 the measurement program Joint North Sea Wave Project (JONSWAP), was carried out. The results have been adopted as an ITTC standard.

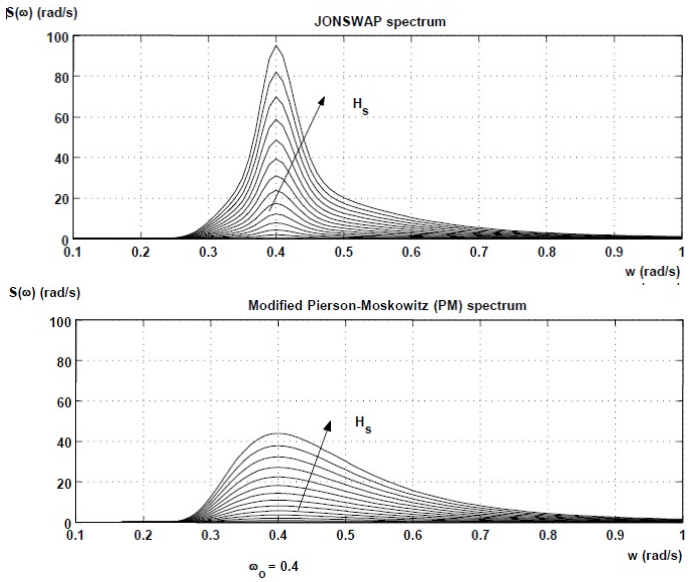

Figure C.1: Wave Spectra, [17].

The JONSWAP spectrum, top graph in Figure C.1, is an empirical relationship that defines the distribution of energy with frequency within the ocean. It is a fetch-limited version of the Pierson-Moskowitz spectrum, bottom graph in Figure C.1, except that the wave spectrum is never fully developed and may continue to develop due to non-linear wave-wave interactions for a very long time.

Appendix D

Calculations in Observability Check

Calculation of the Lie Derivatives, up to an order of 2

$$L_f^0(h_x) = h_x$$

$$L_f^1(h_x) = \sum_{k=1}^{n} f(x)\frac{\partial h(x)}{\partial x_k} = \nabla\langle h(x), f(x)\rangle$$

$$L_f^2(h_x) = L_f L_f h(x) = \sum_{k=1}^{n} f(x)\frac{\partial (L_f h_x)}{\partial x_k} f_k(x)$$

$$\begin{aligned} L_f^0(h_1) &= h_1 = x_1 \\ L_f^0(h_2) &= h_2 = x_2 \\ L_f^0(h_3) &= h_3 = x_3 \end{aligned}$$

$$\begin{aligned} L_f^1(h_1) = f(1)\frac{\partial h(1)}{\partial x_1} + f(2)\frac{\partial h(1)}{\partial x_2} + f(3)\frac{\partial h(1)}{\partial x_3} + f(4)\frac{\partial h(1)}{\partial x_4} + f(5)\frac{\partial h(1)}{\partial x_5} \\ + f(6)\frac{\partial h(1)}{\partial x_6} + f(7)\frac{\partial h(1)}{\partial x_7} + f(8)\frac{\partial h(1)}{\partial x_8} = \dot{x_1} \end{aligned}$$

$$L_f^1(h_2) = f(1)\frac{\partial h(2)}{\partial x_1} + f(2)\frac{\partial h(2)}{\partial x_2} + f(3)\frac{\partial h(2)}{\partial x_3} + f(4)\frac{\partial h(2)}{\partial x_4} + f(5)\frac{\partial h(2)}{\partial x_5} + f(6)\frac{\partial h(2)}{\partial x_6} + f(7)\frac{\partial h(2)}{\partial x_7} + f(8)\frac{\partial h(2)}{\partial x_8} = \dot{x_2}$$

$$L_f^1(h_3) = f(1)\frac{\partial h(3)}{\partial x_1} + f(2)\frac{\partial h(3)}{\partial x_2} + f(3)\frac{\partial h(3)}{\partial x_3} + f(4)\frac{\partial h(3)}{\partial x_4} + f(5)\frac{\partial h(3)}{\partial x_5} + f(6)\frac{\partial h(3)}{\partial x_6} + f(7)\frac{\partial h(3)}{\partial x_7} + f(8)\frac{\partial h(3)}{\partial x_8} = \dot{x_3}$$

$$L_f^2(h_1) = \sum_{k=1}^{8} f(x)\frac{\partial(L_f h_1)}{\partial x_k} f_k(x) = \dot{x_2} = \frac{B_{44}}{I_x + A_{44}}x_2 - \frac{C_{44}}{I_x + A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{K_{\phi^3}}{I_x + A_{44}}x_1^3$$

$$L_f^2(h_2) = \sum_{k=1}^{8} f(x)\frac{\partial(L_f h_2)}{\partial x_k} f_k(x) = -\frac{C_{44}}{I_x + A_{44}}(\overline{GM}_m + \overline{GM}_a x_5)x_2 + \frac{3K_{\phi^3}}{I_x + A_{44}}x_1^2 x_2 + \left(\frac{B_{44}}{I_x + A_{44}}\right)^2 x_2 \ldots - \frac{C_{44}B_{44}}{(I_x + A_{44})^2}(\overline{GM}_m + \overline{GM}_a x_5)x_1 + \frac{B_{44}K_{\phi^3}x_1^3}{(I_x + A_{44})^2} - \frac{C_{44}\overline{GM}_a x_1}{I_x + A_{44}}x_6$$

$$L_f^2(h_3) = \sum_{k=1}^{8} f(x)\frac{\partial(L_f h_3)}{\partial x_k} f_k(x) = -\frac{B_{11}}{m + A_{11}}(x_3 sign(x_3) + |x_3|)\left(\frac{\tau_1}{m + A_{11}} - \frac{B_{11}}{m + A_{11}}x_3|x_3|\right)$$

Calcultaion of the $\mathcal{O}$ matrix, with the gradient of the l vector

$$\frac{\partial L_f^0(h_1)}{\partial x} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^0(h_2)}{\partial x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^0(h_3)}{\partial x} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^1(h_1)}{\partial x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^1(h_2)}{\partial x} = \begin{bmatrix} -\frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5) + \frac{3K_{\phi^3}}{I_x+A_{44}}x_1^2 & \frac{B_{44}}{I_x+A_{44}} & 0 & 0 & -\frac{C_{44}}{I_x+A_{44}}\overline{GM}_a x_1 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^1(h_3)}{\partial x} = \begin{bmatrix} 0 & 0 & -\left(\frac{B_{11}}{m+A_{11}}\right)(x_3 sign(x_3) + |x_3|) & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial L_f^2(h_1)}{\partial x} = \begin{bmatrix} -\frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5) + \frac{3K_{\phi^3}}{I_x+A_{44}}x_1^2 & \frac{B_{44}}{I_x+A_{44}} & 0 & 0 & -\frac{C_{44}}{I_x+A_{44}}\overline{GM}_a x_1 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{aligned}\frac{\partial L_f^2(h_2)}{\partial x} = [&\frac{6K_{\phi^3}x_1x_2}{I_x+A_{44}} - \frac{C_{44}B_{44}}{(I_x+A_{44})^2}(\overline{GM}_m + \overline{GM}_a x_5) + \frac{3B_{44}K_{\phi^3}x_1^2}{(I_x+A_{44})^2} - \frac{C_{44}\overline{GM}_a x_6}{I_x+A_{44}},\\ &-\frac{C_{44}}{I_x+A_{44}}(\overline{GM}_m + \overline{GM}_a x_5) + \frac{3K_{\phi^3}x_1^2}{I_x+A_{44}} + \left(\frac{B_{44}}{I_x+A_{44}}\right)^2, 0, 0,\\ &-\frac{C_{44}\overline{GM}_a x_2}{I_x+A_{44}} - \frac{C_{44}B_{44}\overline{GM}_a x_1}{(I_x+A_{44})^2}, -\frac{C_{44}\overline{GM}_a x_1}{I_x+A_{44}}, 0, 0]\end{aligned}$$

$$\begin{aligned}\frac{\partial L_f^2(h_3)}{\partial x} = [&0, 0,\\ &\dots \quad -\frac{B_{11}}{m+A_{11}}(2 \times sign(x_3))\left(\frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}x_3|x_3|\right)\\ &\dots + \left(\frac{B_{11}}{m+A_{11}}\right)^2\left(\frac{\tau_1}{m+A_{11}} - \frac{B_{11}}{m+A_{11}}x_3|x_3|\right)^2,\\ &\dots \quad 0, 0, 0, 0, 0, 0, 0]\end{aligned}$$

Command lines used in Maple

Calculation of Lie Derivatives

$$
\begin{aligned}
h_1 &:= x_1 \\
L_{11} &:= h_1 \\
L_{12} &:= Gradient(L_{11}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{13} &:= Gradient(L_{12}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{14} &:= Gradient(L_{13}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{15} &:= Gradient(L_{14}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{16} &:= Gradient(L_{15}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{17} &:= Gradient(L_{16}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{18} &:= Gradient(L_{17}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f \\
L_{19} &:= Gradient(L_{18}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])^+.f
\end{aligned}
$$

Gradients of the l vector

$$
\begin{aligned}
Od_1 &:= Gradient(L_{11}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_2 &:= Gradient(L_{12}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_3 &:= Gradient(L_{13}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_4 &:= Gradient(L_{14}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_5 &:= Gradient(L_{15}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_6 &:= Gradient(L_{16}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_7 &:= Gradient(L_{17}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_8 &:= Gradient(L_{18}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]) \\
Od_9 &:= Gradient(L_{19}, [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8])
\end{aligned}
$$

The term for $x[4]$ in Observability Analysis for Wave Direction Estimation

$$
\left(-\frac{1}{I_x + A_{44}}\left(C_{44}\,GM_a\left(-2x_5\left(x_4 + \frac{x_4^2 x_3\cos(-x_9+u_2)}{g}\right)\left(1+\frac{2x_4x_3\cos(-x_9+u_2)}{g}\right)\right.\right.\right.
$$

$$
-\frac{x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^2x_3}{(g-x_3x_4\cos(-x_9+u_2))^2}
$$

$$
\left.\left.+\frac{\cos(x_1)x_3x_6u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)x_3^2x_4x_6u_3\sin(-x_9+u_2)\cos(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}\right)\right)
$$

$$
\left.+\frac{C_{44}\,GM_a x_1\sin(x_1)x_3x_6u_3\sin(-x_9+u_2)}{(I_x+A_{44})(g-x_3x_4\cos(-x_9+u_2))}+\frac{C_{44}\,GM_a x_1\sin(x_1)x_3^2x_4x_6u_3\sin(-x_9+u_2)\cos(-x_9+u_2)}{(I_x+A_{44})(g-x_3x_4\cos(-x_9+u_2))^2}\right)x_2
$$

$$
-\frac{1}{I_x+A_{44}}\left(C_{44}\,GM_a x_1\left(-\frac{2x_5\left(1+\frac{2x_4x_3\cos(-x_9+u_2)}{g}\right)x_4^2\cos(-x_9+u_2)}{g}\right.\right.
$$

$$
-\frac{4x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)x_4\cos(-x_9+u_2)}{g}
$$

$$
-\frac{x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3\,\mathrm{abs}(1,x_3)}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}
$$

$$
\left.-\frac{x_4x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3\,\mathrm{abs}(1,x_3)}{m+A_{11}}\right)\cos(-x_9+u_2)^2x_3}{(g-x_3x_4\cos(-x_9+u_2))^2}\right.
$$

$$
-\frac{2x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^2}-\frac{2x_4^2x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^3x_3}{(g-x_3x_4\cos(-x_9+u_2))^3}
$$

$$
+\frac{\cos(x_1)x_6u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{3\cos(x_1)x_4x_6u_3\sin(-x_9+u_2)x_3\cos(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}
$$

$$
\left.+\frac{2\cos(x_1)x_3^2x_4^2x_6u_3\sin(-x_9+u_2)\cos(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^3}\right)\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\Bigg)
$$

$$
+\frac{2C_{44}\,GM_a x_1\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)\left(1+\frac{2x_4x_3\cos(-x_9+u_2)}{g}\right)x_6}{I_x+A_{44}}-\frac{1}{I_x+A_{44}}\left(C_{44}\,GM_a x_1\left(\right.\right.
$$

$$-\frac{\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}-\frac{x_4\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^2x_3}{(g-x_3x_4\cos(-x_9+u_2))^2}$$

$$+\frac{\cos(x_1)x_3u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)x_3^2x_4u_3\sin(-x_9+u_2)\cos(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}\Bigg)\Bigg(-x_5\Bigg(x_4$$

$$+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\Bigg)^2-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$

$$+\frac{\cos(x_1)x_3x_4x_6u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}\Bigg)\Bigg)+\Bigg(-\frac{2C_{44}GM_ax_2}{I_x+A_{44}}-\frac{B_{44}C_{44}GM_ax_1}{(I_x+A_{44})^2}$$

$$-\frac{C_{44}GM_ax_1\left(-\frac{x_4\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)x_3x_4u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}\right)}{I_x+A_{44}}\Bigg)\Bigg($$

$$-2x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)\left(1+\frac{2x_4x_3\cos(-x_9+u_2)}{g}\right)$$

$$-\frac{x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^2x_3}{(g-x_3x_4\cos(-x_9+u_2))^2}$$

$$+\frac{\cos(x_1)x_3x_6u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)x_3^2x_4x_6u_3\sin(-x_9+u_2)\cos(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}\Bigg)$$

The term for $x[9]$ in Observability Analysis for Wave Direction Estimation

$$\left(-\frac{1}{I_x + A_{44}}\left(C_{44}\,GM_a\left(-\frac{2x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)x_4^2x_3\sin(-x_9+u_2)}{g}\right.\right.\right.$$
$$-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$
$$-\frac{x_4^2x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)x_3\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}-\frac{\cos(x_1)x_3x_4x_6u_3\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$
$$\left.\left.+\frac{\cos(x_1)x_3^2x_4^2x_6u_3\sin(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^2}\right)\right)-\frac{C_{44}\,GM_a\,x_1\sin(x_1)x_3x_4x_6u_3\cos(-x_9+u_2)}{(I_x+A_{44})(g-x_3x_4\cos(-x_9+u_2))}$$
$$\left.+\frac{C_{44}\,GM_a\,x_1\sin(x_1)x_3^2x_4^2x_6u_3\sin(-x_9+u_2)^2}{(I_x+A_{44})(g-x_3x_4\cos(-x_9+u_2))^2}\right)x_2-\frac{1}{I_x+A_{44}}\left(C_{44}\,GM_a\,x_1\left(\right.\right.$$
$$-\frac{2x_5x_4^4x_3\sin(-x_9+u_2)\cos(-x_9+u_2)}{g^2}-\frac{2x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)x_4^2\sin(-x_9+u_2)}{g}$$
$$-\frac{x_4x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3\,\mathrm{abs}(1,x_3)}{m+A_{11}}\right)\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$
$$-\frac{x_4^2x_6\left(-\frac{B_{11}|x_3|}{m+A_{11}}-\frac{B_{11}x_3\,\mathrm{abs}(1,x_3)}{m+A_{11}}\right)\cos(-x_9+u_2)x_3\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}$$
$$-\frac{2x_4^2x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}$$
$$-\frac{2x_4^3x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)^2x_3\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^3}-\frac{\cos(x_1)x_4x_6u_3\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$
$$+\frac{2\cos(x_1)x_4^2x_6u_3\sin(-x_9+u_2)^2x_3}{(g-x_3x_4\cos(-x_9+u_2))^2}-\frac{\cos(x_1)x_3x_4^2x_6u_3\cos(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^2}$$

$$+\frac{2\cos(x_1)\,x_3^2x_4^3x_6\,u_3\sin(-x_9+u_2)^2\cos(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^3}\Bigg)\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\Bigg)$$

$$+\frac{2\,C_{44}\,GM_a x_1\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)x_4^2x_3\sin(-x_9+u_2)\,x_6}{(I_x+A_{44})\,g}-\frac{1}{I_x+A_{44}}\Bigg(C_{44}\,GM_a x_1\Bigg($$

$$-\frac{x_4\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$

$$-\frac{x_4^2\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)\,x_3\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}-\frac{\cos(x_1)\,x_3x_4\,u_3\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$

$$+\frac{\cos(x_1)\,x_3^2x_4^2u_3\sin(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^2}\Bigg)\left(-x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)^2\right.$$

$$\left.-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)\,x_3x_4x_6\,u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}\right)\Bigg)+\Bigg(-\frac{2\,C_{44}\,GM_a x_2}{I_x+A_{44}}$$

$$-\frac{B_{44}\,C_{44}\,GM_a x_1}{(I_x+A_{44})^2}$$

$$-\frac{C_{44}\,GM_a x_1\left(-\frac{x_4\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}+\frac{\cos(x_1)\,x_3x_4\,u_3\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}\right)}{I_x+A_{44}}\Bigg)\Bigg(\Bigg($$

$$-\frac{2x_5\left(x_4+\frac{x_4^2x_3\cos(-x_9+u_2)}{g}\right)x_4^2x_3\sin(-x_9+u_2)}{g}-\frac{x_4x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\sin(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$

$$-\frac{x_4^2x_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}x_3|x_3|}{m+A_{11}}\right)\cos(-x_9+u_2)\,x_3\sin(-x_9+u_2)}{(g-x_3x_4\cos(-x_9+u_2))^2}-\frac{\cos(x_1)\,x_3x_4x_6\,u_3\cos(-x_9+u_2)}{g-x_3x_4\cos(-x_9+u_2)}$$

$$+\frac{\cos(x_1)\,x_3^2x_4^2x_6\,u_3\sin(-x_9+u_2)^2}{(g-x_3x_4\cos(-x_9+u_2))^2}\Bigg)$$

Appendix E

Jacobian Matrix

In this appendix the 9×9 Jacobian Matrix for the EKF for estimation of the direction of the incoming waves is given.

$$A = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}) =$$

$$\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{df}{dx}[2,1] & \frac{B_{44}}{I_x+A_{44}} & 0 & 0 & -\frac{C_{44}GM_a x_1}{i_x+A_{44}} & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{B_{11}|\hat{x}_3|+sign(\hat{x}_3)}{m+A_{11}} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\frac{df}{dx}[6,1] & 0 & \frac{df}{dx}[6,3] & \frac{df}{dx}[6,4] & \frac{df}{dx}[6,5] & \frac{df}{dx}[6,6] & 0 & 0 & \frac{df}{dx}[6,9] \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -{\omega_0}^2 & -2\,\lambda\,\omega_0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \tag{E.1}$$

where

$$\frac{df}{dx}[2,1] = -\frac{C_{44}\left(GM_m + GM_a\hat{x}_5\right)}{I_x + A_{44}} + 3\,\frac{K_{\phi^3}\hat{x}_1^2}{I_x + A_{44}} \tag{E.2}$$

$$\frac{df}{dx}[6,1] = -\frac{\sin(x_1)\hat{x}_3\hat{x}_4\hat{x}_6u_3\sin(\hat{x}_9-u_2)}{g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)} \tag{E.3}$$

$$\begin{aligned}\frac{df}{dx}[6,3] = &-\frac{2\,\hat{x}_5\left(\hat{x}_4+\frac{\hat{x}_4^2\hat{x}_3\cos(\hat{x}_9-u_2)}{g}\right)\hat{x}_4^2\cos(\hat{x}_9-u_2)}{g}\\ &-\frac{\hat{x}_4\hat{x}_6\left(-\frac{B_{11}|\hat{x}_3|+sign(\hat{x}_3)}{m+A_{11}}\right)\cos(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))}\\ &-\frac{\hat{x}_4^2\hat{x}_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)(\cos(\hat{x}_9-u_2))^2}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}+\frac{\cos(\hat{x}_1)\hat{x}_4\hat{x}_6u_3\sin(\hat{x}_9-u_2)}{g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)}\\ &+\frac{\cos(\hat{x}_1)\hat{x}_3\hat{x}_4^2\hat{x}_6u_3\sin(\hat{x}_9-u_2)\cos(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}\end{aligned} \tag{E.4}$$

$$\begin{aligned}\frac{df}{dx}[6,4] = &-2\,\hat{x}_5\left(\hat{x}_4+\frac{\hat{x}_4^2\hat{x}_3\cos(\hat{x}_9-u_2)}{g}\right)\left(1+2\,\frac{\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)}{g}\right)\\ &-\frac{\hat{x}_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)\cos(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))}\\ &-\frac{\hat{x}_4\hat{x}_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)(\cos(\hat{x}_9-u_2))^2\,\hat{x}_3}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}+\frac{\cos(\hat{x}_1)\hat{x}_3\hat{x}_6u_3\sin(\hat{x}_9-u_2)}{g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)}\\ &+\frac{\cos(\hat{x}_1)\hat{x}_3^2\hat{x}_4\hat{x}_6u_3\sin(\hat{x}_9-u_2)\cos(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}\end{aligned} \tag{E.5}$$

$$\frac{df}{dx}[6,5] = -\left(\hat{x}_4+\frac{\hat{x}_4^2\hat{x}_3\cos(\hat{x}_9-u_2)}{g}\right)^2 \tag{E.6}$$

$$\begin{aligned}\frac{df}{dx}[6,6] = &-\frac{\hat{x}_4\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)\cos(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))}+\frac{\cos(\hat{x}_1)\hat{x}_3\hat{x}_4u_3\sin(\hat{x}_9-u_2)}{g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)}\\ \frac{df}{dx}[6,9] = &-\frac{2\,\hat{x}_5\left(\hat{x}_4+\frac{\hat{x}_4^2\hat{x}_3\cos(\hat{x}_9-u_2)}{g}\right)\hat{x}_4^2\hat{x}_3\sin(\hat{x}_9-u_2)}{g}\\ &-\frac{\hat{x}_4\hat{x}_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)\sin(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))}\\ &-\frac{\hat{x}_4^2\hat{x}_6\left(\frac{u_1}{m+A_{11}}-\frac{B_{11}\hat{x}_3|\hat{x}_3|}{m+A_{11}}\right)\cos(\hat{x}_9-u_2)\hat{x}_3\sin(\hat{x}_9-u_2)}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}\\ &-\frac{\cos(\hat{x}_1)\hat{x}_3\hat{x}_4\hat{x}_6u_3\cos(\hat{x}_9-u_2)}{g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2)}+\frac{\cos(\hat{x}_1)\hat{x}_3^2\hat{x}_4^2\hat{x}_6u_3\left(\sin(\hat{x}_9-u_2)\right)^2}{(g-\hat{x}_3\hat{x}_4\cos(\hat{x}_9-u_2))^2}\end{aligned} \tag{E.7}$$

Appendix F

Matlab/Simulink Model

To check that the system equations do resonate for constant speed, the system was implemented in Matlab. The Simulink model is given in Figure F.1, and it depicts the overall model of the system, implemented in Matlab/Simulink. The M-files, used for simulation of the system, are given below.

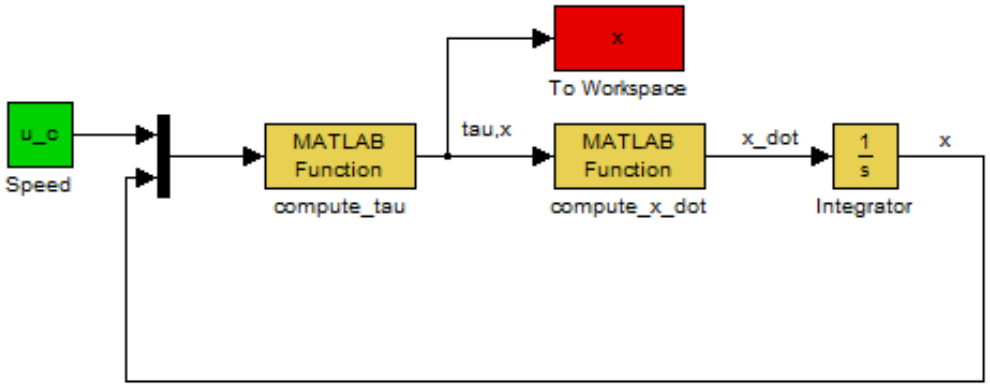

Figure F.1: Simulink model of the system, with feedback linearization controller.

The Simulink model does in turn call the functions in M-files compute_tau.m and compute_x_dot.m, in order to run the complete model. COMPUTE_TAU and COMPUTE_X_DOT are executed through Interpreted MATLAB function blocks, which can be seen in the Simulink model in Figure F.1. Values of the states, x, are saved to the workspace, and the roll angle is plotted at the end of the simulation.

The solver in the Simulink model was set to ode3.

M-files that complete the system model in Matlab/Simulink

- INIT_SIM.M initializes the parameters used in the model, starts the simulation of TheSystem.mdl, and plots the roll angle.
- COMPUTE_TAU.M computes the input to the model, the thrust τ_1.
- COMPUTE_X_DOT.M computes the derivation of the complete system dynamic equations.

INIT_SIM.M

```
% INIT_SIM simulates parametric resonance for a
%  container ship in head seas, and plots the roll angle.
%
% See also: COMPUTE_TAU, COMPUTE_X_DOT, THE_SYSTEM.MDL

clear all; close all; clc;

% Conversion Parameters
D2R = pi/180; % deg to rad
R2D = 180/pi; % rad to deg

% Ship Model Parameters
const.I_x    = 1.4014*(10^10); const.A_44    = 2.17*(10^9);
const.B_44   = -3.20*(10^8);   const.nabla   = 76468;
const.GM_m   = 1.91;           const.GM_a    = 0.84;
const.w_0    = 0.4764;         const.omega_0 = 0.4784;
const.K_phi3 = -2.974*(10^9);  const.lambda  = 0.1018;
const.sigma  = 1.2871;         const.m       = 7.6654*(10^7);
const.A_11   = 7.746*(10^6);   const.rho     = 1025;
const.S      = 6600;           const.g       = 9.81;
const.L_pp   = 281;            const.v       = 1.519*(10^-6);
const.k      = 0.1;            const.u       = 7;
const.k_p    = -10;            const.u_c     = 7;
const.tau    = 2.6805e+005;    u_c           = 7;
const.R_n    = const.u*const.L_pp/const.v;
const.B_11   = 0.5*const.rho*const.S*(1+const.k)...
               *0.075/((log(const.R_n)/log(10)-2)^2);
const.Ix_A44 = const.I_x + const.A_44;
```

```
const.m_A11  = const.m + const.A_11;
const.C_44   = const.rho*const.g*const.nabla;
const.K_w    = 2*const.lambda*const.omega_0*const.sigma;
x_initial    = [0.05*D2R,0,7.5,const.w_0,1,0,0,0];

%% Start Simulation
Tstop = 1000; sim('The_System',Tstop);
%% Plot ...
```

COMPUTE_TAU.M

```
function [ out ] = compute_tau( u_c,x,const )
tau = const.B_11*abs(x(3))*x(3)...
    + const.k_p*(x(3)-u_c)*const.m_A11;
out(1:8) = x;
out(9) = tau;
end
```

COMPUTE_X_DOT.M

```
function [ x_dot ] = compute_x_dot( tau,x,const )
x_dot(1) = x(2);
x_dot(2) = ((const.B_44*x(2))...
           - ((const.C_44*x(1))...
             *(const.GM_m+const.GM_a*x(5)))...
           + (const.K_phi3*(x(1)^3)))/(const.Ix_A44);
x_dot(3) = (tau-const.B_11*abs(x(3))*x(3))/(const.m_A11);
x_dot(4) = 0;
x_dot(5) = x(6);
x_dot(6) = -x(5)*((x(4)+(x(3)*x(4)^2)/const.g)^2)...
           + (x(4)*x(6)/(x(3)*x(4)+const.g))...
             *((tau-const.B_11*x(3))/(const.m_A11));
x_dot(7) = x(8);
x_dot(8) = -(const.omega_0^2*x(7))...
           - (2*const.lambda*const.omega_0*x(8))...
           + (const.K_w*0.1*randn(1));
end
```

Appendix G

Matlab Files to Generate Results

- INIT.M initializes all the parameters for the container vessel.
- COMPUTE_F.M returns the $f = \dot{x}$ for a container vessel.
- COMPUTE_INPUT.M computes the input to the model, thrust τ_1.
- RK4.M integration using Runge-Kutta's 4th-order method.
- BoxMuller.M returns your signal with normally distributed random number.
- StdDev.M, ElapsedTime.M and FFT_EncFreq.M is used for calculation of time complexity, and performance of the filters.
- EKF.M is the extended Kalman filter M-file.
- SIGMAS.M returns the sigma points, X, around a reference point.
- UT_WEIGHTS.M computes unscented transformation weights.
- UKF.M is the M-file for the unscented Kalman filter.
- F_WAVE_DIR.M returns the $f = \dot{x}$ for a container vessel.
- COMPUTE_RUDDER_ANGLE.M calculates the rudder angle.
- NOMOTO.M is a first order nomoto model.
- WaveDirectionEstimation.M implements an EKF for estimation of the direction of the incoming waves.

M-files Used in All The Observers

INIT.M

```
% INIT returns parameters in base SI units (kg-m-s).
clear all; close all; clc;

% Conversion
D2R = pi/180; R2D = 180/pi;

% Ship Model Parameters
const.I_x = 1.4014*(10^10);%Rigid-body inertia (in roll)
const.A_44 = 2.17*(10^9);%Added mass
const.Ix_A44 = const.I_x + const.A_44;
const.B_44 = -3.20*(10^8);%Linear damping coefficient
const.GM_m = 1.91; %MCH
const.GM_a = 0.84; %Amplitude of MCH change
const.omega_0 = 0.4684; %Resonance frequency
const.K_phi3 = -2.974*(10^9);%Restoring force
const.lambda = 0.1018;%Relative damping factor
const.sigma = 1.2871;%Sigma gain
const.K_w = 2*const.lambda*const.omega_0*const.sigma;%Gain h_w(s)
const.m = 7.6654*(10^7);%Mass of the ship
const.A_11 = 7.746*(10^6);%Added mass
const.m_A11 = const.m + const.A_11;
const.L_pp = 281;  %Ship length between perpendiculars
const.v = 1.519*(10^-6);%Kinematic viscosity
const.u = 7;       %Ship speed
const.R_n = const.u*const.L_pp/const.v; %Reynolds number
const.rho = 1025;  %Water density
const.S = 6600;    %Wetted surface
const.k = 0.1;     %Factor
const.B_11 = 0.5*const.rho*const.S... %Hydrodynamic resistance
             *(1+const.k)*0.075...
             /(((log(const.R_n)/log(10))-2)^2);
const.g = 9.81;    %Gravitational acceleration
const.nabla = 76468;%Displaced water volume
const.C_44 = const.rho*const.g*const.nabla;%Restoring moment factor
const.k_p = -1e1;  %Feedback linearization controller gain
const.w_0 = 0.4764;%Modal wave frequency
```

```
const.u_c = 7;      %Control speed
u0 = 7.5;           %Initial speed

%Initial states
x_init = [0.1*D2R,0,u0,const.w_0,1,0,0,0]';
x_init_wave_dir = [0.1*D2R,0,u0,const.w_0,1,0,0,0,pi]';

%Constants for the Nomoto model (mariner class cargo ship)
const.T1 = 118;  const.T2 = 7.8; %Time constants
const.T3 = 18.5; const.T  = const.T1+const.T2-const.T3;
const.K  = 0.185;   %Static yaw rate gain

%Rudder Angle Controller Parameters
const.lambda_1 = 15000;
const.lambda_2 = 8.0;  %Weighting factor
const.K_p = sqrt(1/const.lambda_2);  %Proportional gain
const.K_d = (const.L_pp/const.u)*... %Derivative gain
   ((sqrt(1 + 2*const.K_p*const.K*const.T ...
   + const.K^2*(const.u/const.L_pp)^2 ...
   *(const.lambda_1/const.lambda_2))-1)/const.K);
const.psi_d = 10*D2R;   %Desired heading (deg)
```

COMPUTE_F.M

```
function [ x_dot ] = compute_f( X,U,const )
% COMPUTE_F returns the f = x_dot for a container vessel
%  during parametric resonance, for a given time instant.
  x_dot(1,1) = X(2);
  x_dot(2,1) = ( ( const.B_44*X(2) )...
    - ( const.C_44*X(1)*( const.GM_m+const.GM_a*X(5) ) )...
    + ( const.K_phi3*(X(1)^3) ) ) / const.Ix_A44;
  x_dot(3,1) = ( U-(const.B_11*(abs(X(3)))*X(3)) )/const.m_A11;
  x_dot(4,1) = 0;
  x_dot(5,1) = X(6);
  x_dot(6,1) = ( -X(5)*(( X(4)+(X(3)*(X(4)^2)/const.g) )^2) )...
    + ( (X(4)*X(6)/(X(3)*X(4)+const.g))*x_dot(3,1) );
  x_dot(7,1) = X(8);
  x_dot(8,1) = - (const.omega_0^2)*X(7)...
    - ( 2*const.lambda*const.omega_0*X(8) )...
    + ( const.K_w*BoxMuller(0.1,const.K_w) );
end
```

COMPUTE_INPUT.M

```
function [ tau_1 ] = compute_input( X,const )
%Feedback Linearization Controller
% COMPUTE_INPUT retuns the input u = tau_1
  tau_1 = ( const.B_11*(abs(X(3)))*X(3) )...
        + ( const.k_p*( X(3)-const.u_c )*const.m_A11 );
end
```

RK4.M

```
function xnext = rk4( f,x,u,h )
%RK4 Integrate a system using Runge-Kutta's 4th-order method.
% Copyright (C) 2008 Thor I. Fossen and Tristan Perez
 xo = x;
    k1 = h*feval(f,xo,u);
    x  = xo+0.5*k1;
    k2 = h*feval(f,x,u);
    x  = xo+0.5*k2;
    k3 = h*feval(f,x,u);
    x  = xo+k3;
    k4 = h*feval(f,x,u);
 xnext = xo + (k1+2*(k2+k3)+k4)/6;
end
```

BoxMuller.M

```
function [ Z ] = BoxMuller( stdv,mean )
%Box-Muller Transformation
 s = size(mean,1);
 U = rand(s,2);
 y = zeros(s,1);
  for i=1:s,
     y(i,1) = (sqrt( -2*log(U(i,1) ))...
              *cos( 2*pi*U(i,2) ))*stdv...
              + mean(i,1);
  end
 Z = y;
end
```

StdDev.M

```
%Standard Deviation calculates standard deviation of
% estimation errors for the EKF or the UKF.

%% Choose one filter (comment out the other one)
% EKF;
UKF;
close all; % close plots

%% Standard Deviation of Estimation Errors
N = size(simdata(:,2:9), 2);
x_sim = simdata(:,2:9);
x_est = simdata(:,10:17);

%% Calculate the Std Dev
EstErr = zeros(8);
for i = 1:8,
 EstErr(i) = sqrt(norm(x_sim(:,i)-x_est(:,i))^2/N);
end
%% Display the result ...
```

ElapsedTime.M

```
%Elapsed_Time calculates min, max,
% and average elapsed time for the EKF/UKF.
init;
REPS = 100; % 100 repetitions
for j=1:REPS
   tic;   % starts a stopwatch timer
   UKF;   % run the filter you want to investigate
%    EKF; % (comment out the other one)
   tElapsed(j) = toc; % stops the timer
end
 minTime = min(tElapsed); % min elapsed time
 maxTime = max(tElapsed); % max elapsed time
 averageTime = sum(tElapsed)/REPS; % average time

 %% Display the min, max and average elapsed time...
```

FFT_EncFreq.M

```
%FFT_EncFreq calculates the FFT of estimated state 5,
% calculates and displays the value of the encounter
% frequency, and plots a single-sided amplitude spectrum.

UKF; % Run one filter (comment out the other one)
% EKF;

%% Fast Fourier Transform
g = simdata(:,14);      % estimate of state 5
NFFT = 2^nextpow2(N+1); % next power of 2 from length of g
G = fft(g,NFFT)/(N+1);  % discrete Fourier transform of g
freq = (1/(2*h))*linspace(0,1,NFFT/2+1); % x-axis

%% Calculate the wave encounter frequency from the FFT
Enc_Freq = ((find(abs(G) == max(abs(G)),1)-1)/...
           (NFFT/2+1))*(1/(2*h))*2*pi;

%% Display the value of the calculated encounter freq...
%% Plot Single-Sided Ampl. Spectrum of Enc. Freq...
```

M-files For The Extended Kalman Filter

EKF.M

```
%EKF implements an Extended Kalman Filter for estimation
% for a container ship, during parametric resonance,
% and plots the reults.
%
% See also: INIT, COMPUTE_F, COMPUTE_INPUT, RK4, BoxMuller

init;       % runs the init.m file
h = 0.1;    % sampling time
N = 10000; % no. of iterations
n = 8;      % number of states
m = 3;      % number of measurements
q = 0.1;    % std of process
r = 0.1;    % std of measurement

```

```
% Design matrices
Q = q^0*eye(n);     % covariance of process
R = r^-1.7*eye(m); % covariance of measurement noise
% Initial conditions
x     = x_init;     % initial state
P_bar = eye(n);     % initial state covariance
x_hat = x;          % initial estimate
x_bar = zeros(n,1); % pre-allocating space
P_hat = zeros(n,n); % pre-allocating space
% Memory allocation
simdata = zeros(N+1,17);

%% MAIN LOOP %%
for i = 1:N+1,
 t = (i-1)*h; % time (sec)

 % Measurement equation with noise
 a = [x(1) x(2) x(3)+x(8)]';
 z = BoxMuller(r,a);

%% Partial derivative Matrices
 % Jacobian
 A = [0,1,0...
 % Process noise matrix
 E = [1 0 0 0 0 0 0 0
      0 1 0 0 0 0 0 0
      0 0 1 0 0 0 0 0
      0 0 0 0 0 0 0 0
      0 0 0 0 1 0 0 0
      0 0 0 0 0 1 0 0
      0 0 0 0 0 0 1 0
      0 0 0 0 0 0 0 1]';

%% Discrete-time matrices: PHI, GAMMA, H
 [PHI,GAMMA] = c2d(A,E,h);
 H = [1 0 0 0 0 0 0 0
      0 1 0 0 0 0 0 0
      0 0 1 0 0 0 0 1];

%% PREDICT
```

```
 % Discrete-time extended KF model
 u   = compute_input(x,const);
 f_k = rk4(@(X_HAT,U) compute_f(X_HAT,U,const),x_hat,u,h);

 % Time update, predictor, (k+1)
 x_bar = f_k; % state estimate propagation
 P_bar = PHI*P_hat*PHI' + GAMMA*Q*GAMMA'; % error cov. prop.
 x     = rk4(@(x,u) compute_f(x,u,const),x,u,h); % integration

%% UPDATE
 % Estimation error (residual)
 z_bar = [x_bar(1) x_bar(2) x_bar(3)+x_bar(8)]';
 eps   = z - z_bar;

 % Measurement update, corrector
 K = ( P_bar*H' )/( H*P_bar*H' + R ); % EKF Kalman gain
 x_hat = x_bar + K*eps; % state estimate update
 IKH   = eye(n) - K*H;
 P_hat = IKH*P_bar*IKH' + K*R*K'; % error cov. update

%% Store simulation data in a table
 simdata(i,:) = [t x' x_hat'];
end

%% Simulation values
t         = simdata(:,1);
x_sim_EKF = simdata(:,2:9);
x_est_EKF = simdata(:,10:17);
%% Plots ...
```

M-files For The Unscented Kalman Filter

SIGMAS.M

```
function [ X_sigma ] = sigmas( x,P,c )
% SIGMAS returns the sigma points,
%   X_sigma, around a reference point.
 A = c*(chol(P)');
 Y = x(:,ones(1,numel(x)));
 X_sigma = [x (Y+A) (Y-A)];
```

```
end
```

UT_WEIGHTS.M

```
function [ Wm,Wc,c ] = ut_weights( L,alpha,kappa,beta )
% UT_WEIGHTS computes unscented transformation weights.
 lambda = (( alpha^2 )*( L+kappa )) - L;
 c =  L + lambda;
 Wm = [lambda/c (0.5/c)+zeros(1,2*L)];
 Wc = Wm;
 Wc(1,1) = Wc(1,1) + (1 - (alpha^2) + beta);
end
```

UKF.M

```
%UKF implements an Unscented Kalman Filter for
% for a container ship, during parametric resonance,
% and plots the reults.
%
% See also: INIT, COMPUTE_F, COMPUTE_INPUT,
%  RK4, BoxMuller, SIGMAS, UT_WEIGHTS

init;       % runs the init.m file
h = 0.1;    % sampling time
N = 10000; % no. of iterations
n = 8;      % number of states
m = 3;      % number of measurements
q = 0.1;    % std of process
r = 0.1;    % std of measurement

% Design matrices
Q = q^3*eye(n); % covariance of process
R = r^1*eye(m); % covariance of measurement
% Initial conditions
x = x_init;          % initial state
x_hat_pluss_k = x;  % initial estimate
P_pluss_k = eye(n); % initial state covariance
% Unscented transformation weights
alpha = 1e-4; % transformation parameter
kappa = 0;    % transformation parameter
beta  = 2;    % transformation parameter
```

```
[Wm,Wc,c] = ut_weights(n,alpha,kappa,beta); % weights
c = sqrt(c);
% Memory allocation
simdata = zeros(N+1,17);

%% MAIN LOOP %%
for i = 1:N+1,
 t = (i-1)*h; % time (sec)

 % Measurement equation with noise
 a = [x(1) x(2) x(3)+x(8)]';
 z = BoxMuller(r,a);

%% PREDICT
 % Sigma points around x_hat_pluss_k
 x_hat_k_1 = sigmas(x_hat_pluss_k,P_pluss_k,c);

 %Time update
 u = compute_input(x,const); % system input
 L = size(x_hat_k_1,2);
 x_hat_k     = zeros(n,L); % pre-allocating space
 x_hat_bar_k = zeros(n,1); % pre-allocating space
  for k = 1:L
    % Sigma points propagated through transition funct. f.
    x_hat_k(:,k) = compute_f( x_hat_k_1(:,k),u,const );
    % Combine vectors to obtain a priori state estimate
    x_hat_bar_k = x_hat_bar_k + ( Wm(k)*x_hat_k(:,k) );
  end

 % Estimate a priori error covariance
 P_bar_k = zeros(n,n);
  for k = 1:L
    x_hat_dev = x_hat_k(:,k)-x_hat_bar_k;
    P_bar_k = P_bar_k...
      + ( Wc(k) * (x_hat_dev * x_hat_dev') );
  end
 % add Q, to take process noise into account
 P_bar_k = (P_bar_k + Q);

 % Runge-Kutta 4th order integration
```

```
67  x_hat_bar_k = ...
68      rk4(@(X_Hat,U) compute_f(X_Hat,U,const),x_hat_pluss_k,u,h);
69  x = rk4(@(X,U) compute_f(X,U,const),x,u,h);
70
71 %% UPDATE
72  % Sigma points around x_hat_bar_k
73  x_hat_k = sigmas(x_hat_bar_k,P_bar_k,c);
74
75  % Measurement update
76  y_hat_k     = zeros(m,L); % pre-allocating space
77  y_hat_k_sum = zeros(m,1); % pre-allocating space
78   for k = 1:L
79     % Sigma points projected through observation funct. h.
80     y_hat_k(1,k) = x_hat_k(1,k);
81     y_hat_k(2,k) = x_hat_k(2,k);
82     y_hat_k(3,k) = x_hat_k(3,k) + x_hat_k(8,k);
83     % Combine vectors to obtain predicted measurement
84     y_hat_k_sum = y_hat_k_sum + (Wm(k) * y_hat_k(:,k));
85   end
86
87  % Covariance of predicted measurement
88  P_y = zeros(m,m);
89   for k = 1:L
90     y_hat_dev = y_hat_k(:,k) - y_hat_k_sum;
91     P_y = P_y + ( Wc(k)*( y_hat_dev*y_hat_dev' ) );
92   end
93  P_y = (P_y + R); % add R; take measurement noise into account
94
95  % Cross-covariance of state and measurement
96  P_xy = zeros(n,m);
97   for k = 1:L
98     x_hat_dev = x_hat_k(:,k)-x_hat_bar_k;
99     y_hat_dev = y_hat_k(:,k) - y_hat_k_sum;
100    P_xy = P_xy + ( Wc(k)*( x_hat_dev*y_hat_dev' ) );
101  end
102
103 % Measurement update
104 K_k = P_xy / P_y; % UKF Kalman gain
105 x_hat_pluss_k = x_hat_bar_k... % state estimate update
106                 + K_k * (z - y_hat_k_sum);
```

```
 P_pluss_k = P_bar_k - K_k * P_y * K_k'; % cov. update

%% Store simulation data in a table
 simdata(i,:) = [t x' x_hat_pluss_k'];
end

%% Simulation values
t         = simdata(:,1);
x_sim_UKF = simdata(:,2:9);
x_est_UKF = simdata(:,10:17);
%% Plots ...
```

M-files For The Wave Direction Estimation

F_WAVE_DIR.M

```
function [ x_dot ] = f_wave_dir( X,U,const )
% F_WAVE_DIR returns the f = x_dot for a container vessel
%  during parametric resonance, for a given time instant.
  x_dot(1,1) = X(2);
  x_dot(2,1) = ( ( const.B_44*X(2) )...
    - ( const.C_44*X(1)*( const.GM_m+const.GM_a*X(5) ) )...
    + ( const.K_phi3*(X(1)^3) ) ) / const.Ix_A44;
  x_dot(3,1) = ( U(1)-(const.B_11*(abs(X(3)))*X(3)) )/const.m_A11;
  x_dot(4,1) = 0;
  x_dot(5,1) = X(6);
  x_dot(6,1) = ( -X(5)*...
      ( (X(4)+((X(3)*(X(4)^2)*cos(X(9)-U(2)))/const.g))^2) )...
    - ( (X(4)*X(6)/(const.g - X(3)*X(4) *cos(X(9)-U(2))) )*...
       x_dot(3,1)*cos( X(9)-U(2) ) )...
    - ( (cos(X(1))*X(3)*X(4)*X(6)*U(3)...
       / (const.g - X(3)*X(4) *cos( X(9)-U(2) )) )*...
       sin(X(9)-U(3)) );
  x_dot(7,1) = X(8);
  x_dot(8,1) = - (const.omega_0^2)*X(7)...
    - ( 2*const.lambda*const.omega_0*X(8) )...
    + ( const.K_w*BoxMuller(0.1,const.K_w) );
  x_dot(9,1) = 0;
end
```

COMPUTE_RUDDER_ANGLE.M

```
function [ delta_c ] = compute_rudder_angle( U,const )
%Heading Autopilot
% COMPUTE_RUDDER_ANGLE retuns rudder angle delta_c
 delta_c = - const.K_p *( U(2)-const.psi_d )...
           - const.K_d * U(3);
end
```

NOMOTO.M

```
function [ U_dot ] = nomoto( U,DELTA,const )
%First order nomoto model
% NOMOTO returns [tau_1_dot,psi_dot,r_dot]';
 U_dot(1,1) = 0;
 U_dot(2,1) = U(3);
 U_dot(3,1) = (const.K/const.T)*DELTA...
              - (1/const.T)*U(3);
end
```

WaveDirectionEstimation.M

```
%Wave Direction Estimation implements an EKF for
% estimation of the direction of the incoming waves
% for a container ship during parametric roll resonance.
%
% See also: INIT, COMPUTE_INPUT, F_WAVE_DIR, RK4,
%   BoxMuller, COMPUTE_RUDDER_ANGLE, NOMOTO

init;       % runs the init.m file
h = 0.1;    % sampling time
N = 10000; % no. of iterations
n = 9;      % number of states
m = 3;      % number of measurements
q = 0.1;    % std of process
r = 0.1;    % std of measurement

% Design matrices
Q = q^0*eye(n);     % covariance of process
R = r^-1.7*eye(m); % covariance of measurement noise
% Initial conditions
```

```
x = x_init_wave_dir; % initial state
P_bar = eye(n);      % initial state covariance
x_hat = x;           % initial estimate
u     = [0 0 5]';    % initial input vector
x_bar = zeros(n,1);  % pre-allocating space
P_hat = zeros(n,n);  % pre-allocating space
% Memory allocation
simdata = zeros(N+1,19);

%% MAIN LOOP
for i = 1:N+1,
 t = (i-1)*h; % time (sec)

 % Measurement equation with noise
 a = [x(1) x(2) x(3)+x(8)]';
 z = BoxMuller(r,a);
 % Rudder angle
 delta = compute_rudder_angle(u,const);
 % System input vector
 tau_1 = compute_input(x,const);
 u_temp = rk4(@(u,delta) nomoto(u,delta,const),u,delta,h);
 psi = BoxMuller(r,u_temp(2,1));
 u = [tau_1 psi u_temp(3,1)]';

%% Partial derivative Matrices
 % Jacobian
 A = [0,1,0,...
 % Process noise matrix
 E = [1 0 0 0 0 0 0 0 0
      0 1 0 0 0 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 1 0 0 0 0
      0 0 0 0 0 1 0 0 0
      0 0 0 0 0 0 1 0 0
      0 0 0 0 0 0 0 1 0
      0 0 0 0 0 0 0 0 1]';

%% Discrete-time matrices: PHI, GAMMA, H
 [PHI,GAMMA] = c2d(A,E,h);
```

```
 H = [1 0 0 0 0 0 0 0 0
      0 1 0 0 0 0 0 0 0
      0 0 1 0 0 0 0 1 0];

%% PREDICT
 % Time update, predictor
 x_bar = ... % state estimate propagation
         rk4(@(X_HAT,U) f_wave_dir(X_HAT,U,const),x_hat,u,h);
 P_bar = PHI*P_hat*PHI' + GAMMA*Q*GAMMA'; % error covariance prop.
 x     = rk4(@(x,u) f_wave_dir(x,u,const),x,u,h); % integration

%% UPDATE
 % Estimation error (residual)
 z_bar = [x_bar(1) x_bar(2) x_bar(3)+x_bar(8)]';
 eps   = z - z_bar;

 % Measurement update, corrector
 K     = ( P_bar*H' )/( H*P_bar*H' + R ); % EKF Kalman gain
 x_hat = x_bar + K*eps; % state estimate update
 IKH   = eye(n) - K*H;
 P_hat = IKH*P_bar*IKH' + K*R*K'; % error covariance update

%% Store simulation data in a table
 simdata(i,:) = [t x' x_hat'];
end

%% Simulation values
t         = simdata(:,1);
x_sim_EKF = simdata(:,2:10);
x_est_EKF = simdata(:,11:19);
%% Plot of wave direction estimation ...
```

References

[1] ABS. Guide for the assessment of parametric roll resonance in the design of container carriers. Technical report, American Bureau of Shipping, Houston, Texas, 2004.

[2] T. B. S. S. I. Association. Undergraduate lecture notes on nonlinear control. *RISK WATCH*, 2003.

[3] M. Athans, R. P. Wishner, and A. Bertolini. Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements. TAC-13(6):504, October 1968.

[4] Balchen, Andresen, and Foss. *Reguleringsteknikk*. Institutt for teknisk kybernetikk, NTNU, Trondheim, Norway, 2003.

[5] G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, 29(2):610–611, 1958.

[6] M. Breivik. *Topics in Guided Motion Control of Marine Vehicles*. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway, June 2010.

[7] M. Breivik and T. I. Fossen. Path following of straight lines and circles for marine surface vessels. 2004. Proceedings of the 6th IFAC CAMS.

[8] D. Breu and T. I. Fossen. Extremum seeking speed and heading control applied to parametric roll resonance. Technical report, 2010.

[9] D. A. Breu, C. Holden, and T. I. Fossen. *Ship Model for Parametric Roll Incorporating the Effects of Time-Varying Speed*, chapter 9, In "Parametric Resonance in Dynamical Systems". Springer, 2012.

[10] R. Brown and P. Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. Wiley, third edition, 1997.

[11] G. Bulian. *Development of analytical nonlinear models for parametric roll and hydrostatic restoring variations in regular and irregular waves*. PhD thesis, 2006.

[12] G. Bulian, A. Francescutto, and C. Lugni. On the nonlinear modeling of parametric rolling in regular and irregular waves. *International shipbuilding progress*, 51(2):173–203, 2004.

[13] D. Clarke. The foundations of steering and manoeuvring. pages 10–25, 2003. in Proc. IFAC Conf. Control Applications.

[14] M. Faraday. *On a peculiar class of acoustical figures; and on certain forms assumed by groups of particles upon vibrating elastic surfaces*, volume 121. Philosophical Transactions of the Royal Society of London, 1831.

[15] T. I. Fossen. *Guidance and control of ocean vehicles*. Wiley, New York, 1994.

[16] T. I. Fossen. *Marine Control Systems Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics, Trondheim, Norway, 2002.

[17] T. I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.

[18] T. I. Fossen and T. Perez. Kalman filtering for positioning and heading control of ships and offshore rigs. 2009.

[19] T. I. Fossen and T. Perez. Mss. marine systems simulator. http://www.marinecontrol.org, 2010. Viewed 01.01.2011.

[20] W. France, M. Levadou, T. Treakle, J. Paulling, R. Michel, and C. Moore. An investigation of head-sea parametric rolling and its influence on container lashing systems. 2001.

[21] R. Galeazzi, J. Vidic-Perunnivic, M. Blanke, and J. J. Jensen. Stability analysis of the parametric roll resonance under non-constant ship speed. July 7-9 2008.

[22] S. Ginsberg. Lawsuits rock apl's boat. *San Fransisco Business Times*, November 1998.

[23] J. K. Hedrick and A. Girard. *Control of Nonlinear Dynamic Systems: Theory and Aplications*. 2005.

[24] R. Hermann and A. Krener. Nonlinear controllability and observability. AC-22(5):728–740, October 1977.

[25] J. P. Hespanha. Undergraduate lecture notes on nonlinear control. 2007.

[26] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Wien New York, 1994.

[27] C. Holden and T. I. Fossen. Active control of u-tanks. Technical report, European Control Conference, 2009.

[28] C. Holden and T. I. Fossen. *Parametric Resonance in Dynamical Systems*. Springer-Verlag, 2011.

[29] C. Holden, R. Galeazzi, C. Rodríguez, T. Perez, T. I. Fossen, M. Blanke, and M. d. A. S. Neves. Nonlinear container ship model for the study of parametric roll resonance. *Modeling, Identification and Control*, 28(4):87–103, 2007.

[30] IMO. Revised guidance to the master for avoiding dangerous situations in adverse weather and sea conditions. January 2007.

[31] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. 1997.

[32] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, third edition, 2002.

[33] T. Koyama. *On the Optmium Automatic Steering System of Ships at Sea*, volume 122. J.S.N.A., 1967.

[34] S. Kwon and W. Chung. Combined synthesis of state estimator and perturbation observer. *ASME Journal of Dynamic Systems, Measurement, and Control*, 125(1):19–26, March 2003.

[35] E. L. Mathieu. Mémoire sur le mouvement vibratoire d'une membrane de forme elliptique. *Journal des Mathématiques Pures et Appliquées*, pages 137–203, 1868.

[36] R. v. d. Merwe. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. PhD thesis, Oregon Health and Science University, April 2004.

[37] A. H. Nayfeh and D. T. Mook. *Nonlinear Oscillations*. WILEY-VCH, 2004.

[38] M. Neves. On the excitation of combination modes associated with parametric resonance in waves. 2002.

[39] M. Neves and C. Rodríguez. A coupled third order model of roll parametric resonance. Technical report, LabOceano/COPPE, Federal University of Rio de Janeiro, Brazil, 2005.

[40] K. Nomoto, H. Taguchi, K. Honda, and S. Hirano. On the steering qualities of ships. Technical report, International Shipbuilding Progress, 1957.

[41] N. H. Norrbin. On the added resistance due to steering on a straight course. 1972. In: Proceedings of the 13th ITTC.

[42] T. S. of Naval Architects & Marine Engineers. http://www.sname.org/home/.

[43] J. R. Paulling. The transverse stability of a ship in a longitudinal seaway. *Journal of Ship Research*, 4(4):37–49, 1961.

[44] J. R. Paulling and R. M. Rosenberg. On unstable ship motions resulting from nonlinear coupling. *Journal of Ship Research*, 3(1):36–46, 1959.

[45] T. Perez and T. I. Fossen. Kinematic models for manoeuvring and seakeeping of marine vessels. 28, 2007. Bulletin of the Norwegian Society of Automatic Control.

[46] K. Salahshoor and M. R. Jafari. On-line identification of non-linear systems using adaptive rbf-based neural networks. *International Journal of Information Science and Technology*, 2007.

[47] L. Schwartz and E. Stear. A computational comparison of several nonlinear filters. 13(1):83–86, 1968.

[48] Y. Shin, V. Belenky, J. Paulling, K. Weems, and W. Lin. Criteria for parametric roll of large container ships in longitudinal seas. (112):117–147, 2004. Transactions of SNAME.

[49] D. Simon. *Optimal State Estimation*. John Wiley & Sons, June 2006.

[50] P. A. Tipler and G. Mosca. *Physics For Scientists and Engineers*. W. H. Freeman and Company, New York, sixth edition edition, 2008.

[51] J. K. Uhlmann. Simultaneous map building and localization for real time applications. Transfer thesis, University of Oxford, 1994.

[52] J. van Amerongen. *Adaptive Steering of Ships; A Model Reference Approach to Improved Maneuvering and Economical Course-Keeping*. PhD thesis, Delft University of Technology, 1982.

[53] J. Van Amerongen and H. R. Van Nauta Lemke. Optmium steering of ships with an adaptive autopilot. 1978. In: Proceedings of the 5th Ship Control Systems Symposium.

[54] J. Van Amerongen and H. R. Van Nauta Lemke. Criteria for optimum steering of ships. 1978. In: Proceedings of Symposium on Ship Steering Automatic Control.

[55] J. Vidic-Perunovic and J. Juncher Jensen. Parametric roll due to hull instantaneous volumetric changes and speed variations. (36):891–899, 2009.

[56] B. vik. Integrated satellite and inertial navigation systems. Trondheim, Norway, 2011.

[57] E. A. Wan and R. v. d. Merwe. The unscented kalman filter for nonlinear estimation. 2000. Adaptive Systems for Signal Processing, Communication and Control.

[58] G. Welch and G. Bishop. An introduction to the kalman filter. July 2006.