Forside

ISBN-side

Preface

The present thesis is concerned with predictive control and identification. Special emphasis is laid on the optimal constrained predictive control, as an appealing procedure to handle the hard constraints, existing on a physical system, in a systematic way. A particular problem, which is of importance for predictive self tuning control, is the interplay between identification and predictive control. The discussed techniques are applied on a stochastic ship model.

This thesis has been prepared at the Department of Mathematical Modelling (IMM), Control Group, at the Technical University of Denmark (DTU), in partial fulfillment of the requirements for the degree of Ph.D. in engineering.

It is a pleasure for me to take the opportunity to thank everybody who enabled me to do the work presented in this thesis.

I would like first to thank Niels Kjølstad Poulsen for his willingness to be my supervisor and for his interest in the project.

I have been very pleased to be a member of the IMM, DTU, and I wish to express thanks for the good and stimulating atmosphere. I am also very grateful to the librarian of IMM, Finn Kuno Christensen, who is always ready to help you to find even the most obscure literature sources.

I would like to express my gratitude to Jan Holst for his helpful comments to parts of the manuscript. I also wish to thank Bjarke Rasmunssen for his suggestions.

I wish to thank Captain H. H. Nielsen and his crew on the Danish Environmental Ship *Gunnar Thorsen* for the interesting and fruitful practical discussions.

I would like to thank Anne Jepsen for her linguistic corrections to the manuscript.

A special acknowledgment is to my parents, Monica and Dumitru Petrov, whom I can never thank enough for their support and encouragement. I wish also thank my brother, Remus Petrov, for his sense of humour which helped me to relax and to get inspiration. I hope that Pluto will forgive me and understand why I could not "take a walk" all the times he wanted to!

Many authors of theses will permanently remember the special support of the person closest to them during the writing of it: I am deeply indebted to my husband, Lars Henrik Hansen, for his willingness to listen and read, for his insight and his patience. He has been my most loyal supporter and my harshest critic.

Nivå, July 1'st 1996.

Anca Daniela Hansen

Summary

The main objective of the present thesis is to enhance insight into predictive controller design and identification in connection with steering dynamics.

In Chapter 2, the dynamics of ship steering are reviewed. Models of different complexity, suitable for control systems design are presented. The influence of wind, waves and currents on the ship motions are also discussed.

Chapter 3 deals with the model reduction problem. Some basic concepts are explained, due to their role in the reduction of the dynamic models. Two model reductions techniques, based on singular values, are described. The theoretical properties of these methods are studied and their performance is examined via simulation on a stochastic linear Mariner Class Vessel model.

In Chapter 4, the attention is focused on the derivation of an extended GPC. This extended strategy implies a generalization of the model structure and of the loss function, which defines the optimality of the control. Some guidelines on how to choose the design parameters, depending on the type

of process to be controlled and on the required control performance, are presented. A predictive track keeping system for a Mariner Class Vessel is formulated based on the minimization of the mean squares prediction errors of the ship's deviation from the desired track.

Chapter 5 is concerned with constrained predictive control. The presented algorithm, which is based on Rosen's gradient projection method, minimizes a multi-step quadratic loss function, taking physical constraints systematically into account. The constraints may consist of amplitude constraints (signal level constraints) as well as rate constraints. The influences of the different parameters on the solution are illustrated via simulation experiments on a Mariner Class Vessel model. The results show that the proposed strategy leads to a significant better control than the ad-hoc control strategy.

Chapter 6 gives a survey on the so-called forgetting factor methods designed for tracking slowly drifting system parameters. The goal of this chapter is to formulate the identification framework in order to support the understanding of the connection between identification and control, analysed in Chapter 7.

Chapter 7 focuses on how to make the on-line identification for predictive control more robust towards unmodelled dynamics. The theory is verified via simulation studies on a Mariner Class Vessel. The effects and the need of a prefilter in the estimation are analysed and illustrated. Based on the idea that the control criterion must be dual to the estimation criterion, an iterative optimal prefilter is designed. This seems to be an appealing way to tune the model towards the objective for which the model is to be used.

Sammenfatning

Hovedformålet med nærværende Ph.D. rapport er at udvide indsigten i prediktiv kontrol design og identifikation i forbindelse med skibsstyringsdynamik.

I kapitel 2 gennemgås dynamikken for skibsstyring. Modeller af forskellig kompleksitet – egnet til kontrol system design – præsenteres. Påvirkninger fra vind, bølger og strøm på skibets bevægelser diskuteres også.

Kapitel 3 omhandler model reduktionsproblemet. Nogle grundbegreber forklares, på grund af den rolle de spiller i reduktionen af de dynamiske modeller. To model-reduktionsteknikker baseret på singulerværdier beskrives. Teoretiske egenskaber for disse metoder studeres og deres performance undersøges med simulation i en linear stokastisk model af et Mariner Class fartøj.

I kapitel 4 fokuseres opmærksomheden på udledningen af en udvidet GPC. Denne udvidede strategi medfører en generalisering af modelstrukturen og

af tabsfunktionen, som definerer optimaliteten af kontrolproblemet. Nogle retningslinier for, hvordan designparametrene skal vælges – afhængig af procestypen, som ønskes kontrolleret, og af den ønskede kontrol performance – præsenteres. Et prediktivt sporingssystem for et Mariner Class fartøj baseret på minimering af mindste kvadraters prediktionsfejlen af skibets afvigelse fra den ønskede rute formuleres.

Kapitel 5 beskæftiger sig med begrænsninger i den prediktive kontrol. Den viste algoritme, som er baseret på Rosens gradient projektionsmetode, minimerer en flertrins kvadratisk tabsfunktion, idet fysiske begrænsninger tages i betragtning. Begrænsningerne kan bestå både af amplitude begrænsninger (signal niveau begrænsninger), og af hastighedsbegrænsninger. Indvirkningen af de forskellige parametre på løsningen illustreres via simulationseksperimenter med modellen af et Mariner Class fartøj. Resultaterne viser at den foreslåede strategi fører til signifikant bedre kontrol samlignet med ad-hoc kontrolstrategien.

Kapitel 6 giver et overblik over de såkaldte glemselsfaktormetoder, som er designet til at følge langsomt varierende parametre. Formålet med dette kapitel er at formulere grundlaget for identifikationen og derved underbygge forståelsen af forbindelsen mellem identifikationen og kontrollen, som analyseres i kapitel 7.

Kapitel 7 fokuserer på, hvordan on-line identifikation gøres mere robust over for umodellerede dynamikker. Teorien verificeres via simulationsstudier på et Mariner Class fartøj. Effekten og behovet for et prefilter i estimationen analyseres og illustreres. Baseret på idéen om, at kontrolkriteriet skal være dualt til estimationskriteriet, designes et iterativt optimalt prefilter. Dette synes at være en atraktiv måde at fokusere modellen ind i dens brugsområde.

Parts of this thesis have previously been published in:

- A. D. Hansen, A. Christensen, N. K. Poulsen (1995): *Model Reduction and Gain scheduling Control*. Technical report no. 12/95, IMM, The Technical University of Denmark.
- A. D. Hansen, N. K. Poulsen (1995): *Level and Rate Constraints in Predictive Control*. Technical report no. 20/95, IMM, The Technical University of Denmark.
- A. D. Hansen, N. K. Poulsen (1996): *Tracking Keeping Systems with Predictive Control*. Technical report no. 4/96, IMM, The Technical University of Denmark.
- A. D. Hansen, N. K. Poulsen (1996): *Predictive control relevant prefiltering*. Technical report no. 10/95, IMM, The technical University of Denmark.

Contents

Preface **I**

Summary **V**

Sammenfatning **VII**

1 Introduction **1**

1.1 Background 1

1.2 Outline of the thesis 4

1.3 Glossary 6

2 Steering dynamics and disturbances' models **11**

2.1 Task and complexity in the ship dynamic modelling 12

2.2 Basic Newtonian equations 13

2.3 The forces' terms 18

2.4 Nonlinear models of the ship 20

2.5 Linear models of the ship 24

2.6 The steering machine 26

2.7 Modelling the disturbances 27

2.7.1 The principle of superposition 29

2.7.2 Wind 29

2.7.3 Waves 38

2.7.4 Currents 49

2.8 Summary 51

3 On model reduction **53**

3.1 Motivation 54

3.2 Prelude to model reduction 57
3.2.1 Controllability and observability matrices 57
3.2.2 Grammian matrices 60
3.2.3 Hankel singular values and Hankel matrix 64
3.2.4 Error norms 68
3.3 Reduction strategies 70
3.3.1 Balanced realizations 71
3.3.2 Balanced Truncation Approximation (BTA) 74
3.3.3 Hankel Norm Approximation (HNA) 77
3.4 On the optimality aspect 79
3.5 Testing the reduced order approximation 81
3.6 Model reduction and control of a ship's model 83
3.6.1 High order ship model 83
3.6.2 Model reduction 85
3.6.3 Control based on reduced design model 92

3.7 Summary 95

4 Unconstrained predictive control 97

4.1 The model and the cost function 100
4.2 Derivation of the control law 102
4.3 Choice of design parameters 116
4.4 Track keeping systems 119
4.4.1 Statement of the tracking problem 120
4.4.2 Maneuvering dynamics and kinematics 121
4.4.3 Tracking and coordinate systems 123
4.4.4 Route generator and the approach of a new trajectory section 126
4.4.5 Predictive tracking controller 128
4.4.6 Computer simulation 131
4.5 Summary 138

5 Optimal linear constrained predictive control 139

5.1 Statement of the control problem 141
5.2 Optimal searching strategy 145
5.3 Rosen's gradient projection method 148
5.4 Calculation of the gradient and the implementation 155
5.5 Level constraints 160
5.6 Rate constraints 175
5.7 Level and rate constraints 182
5.8 Summary 187

6 Parameter estimation algorithms 189

6.1 Some formal definitions 190
6.2 Recursive one step prediction error methods 193
6.3 Identification of the time-varying systems 198
6.3.1 Constant forgetting factor algorithm 199
6.3.2 Variable forgetting factor algorithm 205
6.3.3 Constant trace algorithm 208

6.3.4 SF1 algorithm 209
6.4 Summary 210

7 Predictive control relevant prefiltering 213

7.1 The basic theory elements and notations 215
7.1.1 Bias and variance 215
7.1.2 Prediction error method 219
7.1.3 Bias in open loop identification 222
7.2 Why prefilter ? 225
7.3 Bias distribution in closed loop identification 228
7.4 Derivation of predictive control relevant prefilter 232
7.5 An iterative design scheme 239
7.6 Related problems 240
7.7 Illustrative examples 241
7.7.1 Open loop simulations – analysis 242
7.7.2 Closed loop simulations – analysis 249

7.8 Summary 258

8 Conclusions **261**

Appendices **267**

A Mariner Class Vessel **269**

B Normalization **275**

C Model reduction – DATA **277**

D LQG controller design **281**

E Curve route generation **289**

F Optimal step for the unconstrained problem **293**

Bibliography **297**

This page is intentionally left empty

Chapter 1

Introduction

1.1 Background

The concept of predictive control originates from the late seventies and nowadays it has evolved to a mature level. Predictive controllers have remarkable features. They can be used to control a wide variety of processes, among which are non-minimum phase and unstable processes. Predictive controllers are also fairly robust with respect to unmodelled dynamics and furthermore, they are able to deal with pre-scheduled reference trajectory. However, the most attractive feature of predictive controller design is that constraints can be included in a straight forward manner.

In real situations most practical control schemes have to deal with constraints. These constraints may represent physical limitations of the system (e.g. limitations on the actuators) or they may originate in the process of modelling the physical system (i.e. the states are constrained to remain in a neighborhood of the operation points where the linear representation is sufficiently accurate). Control engineers have usually dealt with this class of problem by designing a controller, which ignores the constraints in the design and simply truncates the signal to the limits. This is known as the ad hoc approach. While appealing in its simplicity, the ad hoc approach does not guarantee an acceptable performance. The stability of the system can even disappear. Thus, the control system has to anticipate constraints violations and correct them in an appropriate way. Therefore the analysis of control systems, which takes the constraints systematically into account in the design, is both of theoretical and practical interest. Despite this importance, little attention was paid to the literature concerning the fact that the hard constraints should be taken into account in the controller design specifications. In this thesis, attention will be focused on the ability and the effectiveness of the predictive control to handle physical constraints in a natural and systematic way. The ability to cope with physical constraints on control signal indicates an improved robustness of the controller to the signal saturation.

It is well-known that predictive control belongs to the class of model-based controller design concepts. This means that, a mathematical model of the process is explicitly used in the design of the controller. The mathematical model plays an important issue in synthesis, analysis and implementation of the control system, due to the fact that it expresses the most important characteristics of the system and it gives insight into the behavior of the systems. There are two fundamental different ways in which the problem of formulating a mathematical model can be approached. The classical modelling involves an analytical approach, when the model is constructed on physical knowledge about the system. Because of computational and other practical limitations, the order of such a model (usually very high) should be reduced for synthesis, analysis and implementation of the control system. Thus robustness towards numerical problems, which possibly appear in the design of a controller, can be increased. Another way to create a mathematical model is an experimental approach. Some experiments are performed on the system and a model is then fitted to the recorded data by assigning suitable numerical values to its parameters. This procedure is known as system identification.

In general, the mathematical model of a system will not give a complete description of the behavior of the system, but however the model should characterize the physical system with sufficient fidelity, especially at those frequency ranges which are important for the controller. The inevitable model error should be adapted to the control objective. Therefore a number of modifications must be made to insure a robust adaptation and control. For example, in the case of predictive self tuning control, a prefiltering of data should be done before identification. In this way, the estimated model can supply the best possible output predictions over the prediction horizon of the controller. Thus, the controller becomes robust with respect to the mismatch between the system and the mathematical model.

The thesis contains a number of simulation examples which aim is to illustrate the practical behaviors of the studied methods. The attention is focus on ship steering. The reason for applying more sophisticated autopilots for ship steering is mainly due to fuel economy and performance improvement. The type of ship selected for study is of the Mariner Class Vessel, a 16622 m^3 displacement, 172 meters long cargo ship with a design speed of 15 knots (7.71 m/s). This class of vessel was designed in the mid-sixties and it is not in use anymore. The choice was made, however, because the hydrodynamic data available are one of the most complete set published for a particular vessel.

The methods which are to be described in this thesis are constructed for SISO processes (Single-Input Single-Output). It is, however, possible to extend the methods in order to make them applicable to MIMO processes (Multiple-Input Multiple-Output).

1.2 Outline of the thesis

In this section an outline of the thesis is given. Each chapter has its own introduction and summary, which may be consulted for additional information.

Chapter 2 contains a description of mathematical model of ship steering dynamics and disturbances, mainly intended for simulation and for control system design in combination with system identification.

Chapter 3 is devoted to model reduction problem. In a number of applications, the control engineer is faced with the control of a physical system for which, an analytical model can be derived in the form of a large number of coupled differential equations. Due to computational and practical limitations, the order of such a model must be reduced for synthesis, analysis and implementation of the control system. Two model reduction techniques, based on singular values, are described and illustrated on a linear model of the Mariner Class Vessel.

Chapter 4 focuses on the design of predictive control in the case of a general model structure and control criterion. In order to illustrate the ability of predictive control to deal with a pre-computed reference trajectory, the design of a route tracking system for the Mariner Class Vessel is performed. The synthesis of the ship's control system is based upon the minimization of the mean square prediction errors of the ship's deviation from the desired track.

Chapter 5 is concerned with the design and implementation of predictive controllers, when hard constraints on the controller output should be taken explicitly into account. The constraints may be on the amplitude as well as on the rate of e.g. the controller output. The constrained optimization problem must be solved at each sample, because an analytical solution is no available. The optimization algorithm used to solve the nonlinear constrained problem is based on Rosen's gradient projection method. The performance of the closed loop system and the numerical complexity involved in solving the constrained optimization problem are investigated on a stochastic linear model of the Mariner Class Vessel.

Chapter 6 focuses on the parameter estimation problem. Furthermore, attention is drawn to the case where the estimation has to be performed on-line, namely on recursive techniques. The chapter gives a quick survey of some variants of the ordinary RLS, namely forgetting factor methods.

They are based on the assumption that the variation of the parameters is so slow that they can be regarded as constant within a horizon with a reasonable length.

Chapter 7 deals with the analysis of the interconnection between identification and control in the case of predictive adaptive control. In order to obtain a robust system it is important to restrict the frequency range for which a good model is needed. This can be done by filtering the data, before they are used in the estimation of the model parameters. The design and analysis of this prefilter are done iteratively, by making the identification objective function dual to the control objective function. The identification strategy is "control-relevant" and thus it makes the overall adaptive control "optimal". An iterative scheme of identification and control, based on least squares identification and general predictive control design, is presented. The theoretical aspects are supported by some illustrative simulations on linear ship models.

Throughout the thesis, all units are SI-units unless specified.

1.3 Glossary

This list contains the most important symbols and abbreviations used in this thesis.

Notations

- A vector will be denoted by underlining and a matrix by bold face character.
- A positive definite matrix $\boldsymbol{A}$ will be denoted by $\boldsymbol{A} > 0$ and a positive semi-definite matrix $\boldsymbol{A}$ will be denoted by $\boldsymbol{A} \geq 0$.
- $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$ - represents a realization of the transfer function of the system of order n. The statements that the system is controllable or observable have the standard meaning.

Some of the most important symbols are listed below:

Symbols

T_s	Sampling period in seconds
J	Loss function alias criterion function
$tr\{\boldsymbol{A}\}$	Trace of matrix $\boldsymbol{A}$
$[\boldsymbol{A}]$	Dimension of matrix $\boldsymbol{A}$
η	Accuracy of solution found by DGP
H_p	Prediction horizon
H_m	Minimum cost horizon
H_c	Control horizon
ρ	Weighting factor
P_n, P_d, Q_n, Q_d	Polynomials in the control cost function

R, S, T	Controller polynomials
u^{min}	Lower bound level constraint
u^{max}	Upper bound level constraint
Δu^{min}	Lower bound rate constraint
Δu^{max}	Upper bound rate constraint
$\boldsymbol{H}$	Hessian matrix
$\underline{K}$	Kalman gain
$\boldsymbol{K}$	Matrix with the active constraints
$\boldsymbol{P}$	Projection matrix or covariance matrix
L	Prefilter polynomial

Abbreviations

SISO	Single-Input Single-Output
MIMO	Multiple-Input Multiple-Output
deg	Degrees
rad	Radians
LTI	Linear Time Invariant
ARX	Auto-Regressive with eXogenous input
ARMAX	Auto-Regressive Moving-Average with eXogenous input

LS	Least Squares Method
RLS	Recursive Least Squares Method
RELS	Recursive Extended Least Squares Method
RMS	Recursive Maximum Likelihood Method
BTA	Balanced Truncation Approximation
HNA	Hankel Norm Approximation
LQG	Linear Quadratic Gaussian
MV	Minimum Variance
GMV	Generalized Minimum Variance
GPC	Generalized Predictive Control
UPC	Unified Predictive Control
DGP	Dedicated Gradient Projection Method

Chapter 2

Steering dynamics and disturbances' models

This chapter reviews some aspects concerning the ship dynamics, the influence of the wind, waves and currents on the ship's motion.

Some nonlinear and linear models for the ship in deep, calm and unrestricted water are summarized. The models are confined to motions in the horizontal plane only.

2.1 Task and complexity in the ship dynamic modelling

The steering and maneuvering performance of ships have been treated in various papers. In the pioneer years, the investigation of the ship dynamics was based on the hydrodynamic theory and experiments. Work of this kind was presented in *[Nomoto, 1957]*, *[Abkowitz, 1964]*, *[Norrbin, 1970]* and *[Chislett and Tejsen, 1965]*. Later, the problem of determining ship dynamics has been developed from the point of view of system identification, as in *[Åström and Källström, 1976]*, *[Abkowitz, 1980]* and *[Amerongen, 1985]*.

Modelling of ship dynamics incorporates all real world conditions of operation and it is still a very complex problem, due to several approximations regarding the hydrodynamic derivatives, the perturbations of wind, waves and current.

The complexity and the correctness of different model structures of ship dynamics are chosen, taking the required application into account. A general classification of the mathematical models of the ship can be done, as follows:

- Mathematical models developed for prediction of the ships' steering performances and for training simulators. These models are generally required to be the most complex and precise models, able to be adapted to a variety of types of ship and applicable over a wide range of ship's speed and engine revolutions.

- Mathematical models developed for control purposes. These models are usually of reduced complexity. However, they must be able to describe rather precisely the structure of the system to be controlled, e.g. the dynamical responses of ship's motions caused by rudder deflections and the external perturbations (waves, wind and currents).

In the present chapter, some mathematical model of ship dynamics for control purposes are summarized.

2.2 Basic Newtonian equations

To analyze the dynamics of a ship, it is convenient to take two coordinate systems into account: an inertial system $X_oO_oY_oZ_o$ (Earth fixed) and a system $XOYZ$ (ship fixed), as it is indicated in the Figure 2.1.

The ship axes are usually defined as:

- X – longitudinal axis (directed from aft to fore).
- Y – transverse axis (directed to starboard).
- Z – normal axis (directed from top to bottom).

With the assumption of the ship as a rigid body, the motions of the ship can be totally described by its six degrees of freedom of motion, namely three rotations (roll, pitch and yaw) and three translations (surge, sway

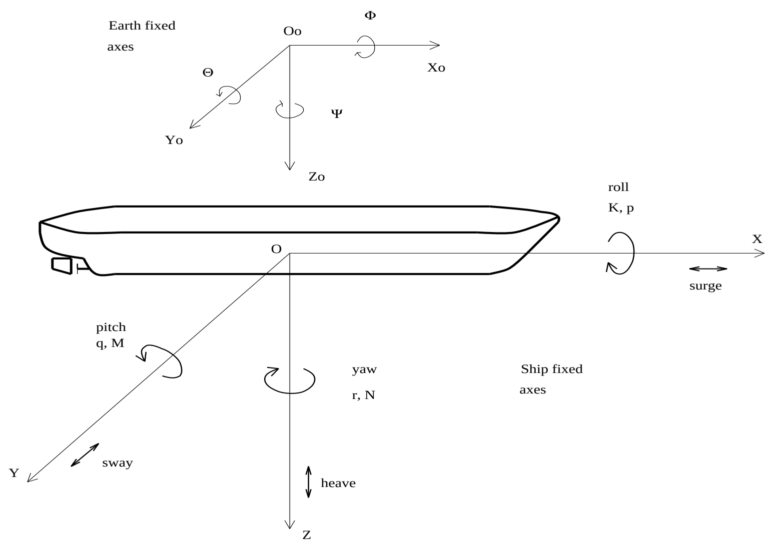

Figure 2.1. *Ship fixed and Earth fixed axes.*

and heave). The physical definitions and the nomenclature associated with states and motions are shown in Figure 2.1 and summarized in Table 2.1.

The equations of the ship's motion are usually expressed by using the ship fixed coordinate system. The hydrodynamic forces and moments acting on the ship are easy to describe in such a system because the symmetry of the hull can be exploited.

The dynamic behavior of a ship is described through Newton's laws, which express the conservation of linear and angular momentum:

$$\begin{bmatrix} m\,\underline{a}_{abs} \\ I\,\underline{\dot{\Omega}}_{abs} \end{bmatrix} = \begin{bmatrix} \underline{\mathcal{F}} \\ \underline{\mathcal{T}} \end{bmatrix} \tag{2.1}$$

axis	**x**	**y**	**z**
translation	surge	sway	heave
position	x	y	z
velocity	u	v	w
force	X	Y	Z
+ direction	forward	starboard	downward
rotation	roll	pitch	yaw
angle	Φ	θ	Ψ
rate	p	q	r
torque	K	M	N
+ direction	starboard	aft down	right turn

Table 2.1. *Motions nomenclature.*

where m denotes the mass of the ship, $\underline{a}_{abs} = \begin{bmatrix} \dot{u} & \dot{v} & \dot{w} \end{bmatrix}^T$ the acceleration of the centre mass of the ship, $\boldsymbol{I}$ the inertia tensor (see *[Blanke, 1981]*), $\underline{\Omega}_{abs} = \begin{bmatrix} p & q & r \end{bmatrix}^T$ is the angular acceleration referred to the inertial system, and $\underline{\mathcal{F}} = \begin{bmatrix} \mathcal{X} & \mathcal{Y} & \mathcal{Z} \end{bmatrix}^T$, and $\underline{\mathcal{T}} = \begin{bmatrix} \mathcal{K} & \mathcal{M} & \mathcal{N} \end{bmatrix}^T$ denote the forces and moments respectively, acting upon the ship's hull.

The terms on the left hand side of (2.1) describe the inertial responses and they are based on the symmetry of the ship's hull. They are derived as described in *[Abkowitz, 1964]*, *[Norrbin, 1970]* and *[Blanke, 1981]*:

$$m\underline{a}_{abs} = m \begin{bmatrix} \dot{u} - rv + qw - (q^2 + r^2)x_G + (pq - \dot{r})y_G + (rp + \dot{q})z_G \\ \dot{v} - pw + ru - (r^2 + p^2)y_G + (qr - \dot{p})z_G + (pq + \dot{r})x_G \\ \dot{w} - qu + pv - (p^2 + q^2)z_G + (rp - \dot{q})x_G + (qr + \dot{p})y_G \end{bmatrix}$$

$$\boldsymbol{I}\,\dot{\underline{\Omega}}_{abs} = \begin{bmatrix} I_x\dot{p} + (I_z - I_y)rq + m[y_G(\dot{w} + pv - qu) - z_G(\dot{v} + ru - pw)] \\ I_y\dot{q} + (I_x - I_z)rp + m(z_G(\dot{u} + qw - rv) - x_G(\dot{w} + pv - qu)] \\ I_z\dot{r} + (I_y - I_x)pq + m[x_G(\dot{v} + ru - pw) - y_G(\dot{u} + qw - rv)] \end{bmatrix} \quad (2.2)$$

Table 2.1 contains the notations used in the expressions (2.2). I_x, I_y, I_z are the moments of inertia of the ship about the X, Y, Z axes, respectively. (x_G, y_G, z_G) are the coordinates of the ship's centre of gravity. For convenience, the origin O coordinates of the ship's fixed system can be selected in the centre line of the ship. In this case $y_G = 0$ and $z_G = 0$. Using further the symmetry of actual hull forms, several terms in the complicated description in (2.2) are vanished.

An additional simplification can be used for large tankers or similar ships, of which motion can be acceptably restricted to the horizontal plane. For these types of ships, the heave, roll and pitch motions are assumed negligible, namely $w = \Phi = \theta = 0$, and thus, the yawing rate is defined as:

$$\dot{\psi} = r$$

Throughout thesis, only the horizontal motions surge, sway and yaw are considered, due to the fact that a model of a large ship is available in the simulations. The remaining motions such as roll, pitch, heave are assumed to be negligible.

The variables used to describe the horizontal motion of a ship are sketched in Figure 2.2. The components of the total ship's speed V on the X and Y axes are the surge velocity u and the sway velocity v. The yaw rate, the heading and the rudder angle are denoted by r, ψ, δ respectively.

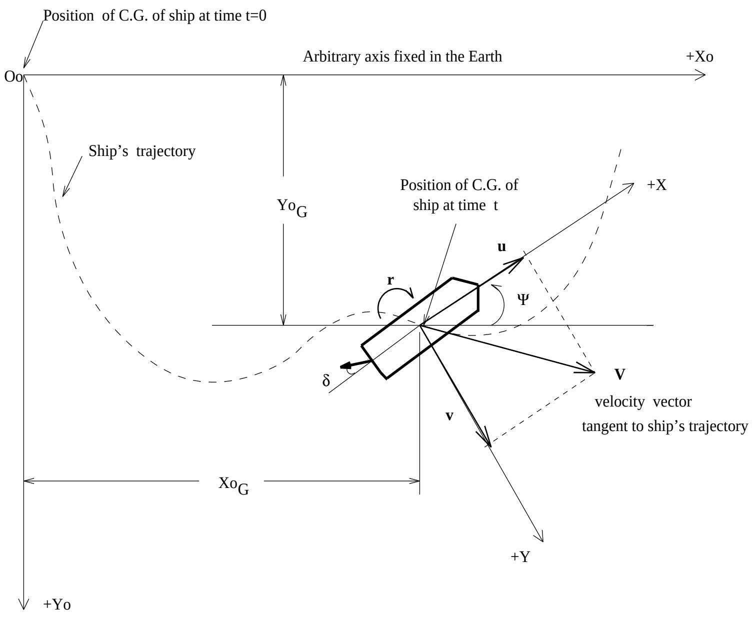

Figure 2.2. *Orientation of Earth's fixed axes and ship's fixed axes. Variables used to describe the motion in the horizontal plane.*

By using the simplifications described above, the number of the equations of motion (2.2) is reduced from 6 equations to 3 equations, as follows:

$$\begin{bmatrix} m\,a_{\mathrm{x}} \\ m\,a_{\mathrm{y}} \\ I_{\mathrm{z}}\,\dot{r} \end{bmatrix}_{\mathrm{abs}} = \begin{bmatrix} m(\dot{u} - rv - x_G r^2) \\ m(\dot{v} + ru + x_G \dot{r}) \\ I_z \dot{r} + m x_G(\dot{v} + ru) \end{bmatrix} \tag{2.3}$$

The expression (2.3) describes the term on the left hand side of (2.1) for horizontal motion. The problem left now is how to determine the right hand side vector term of (2.1).

2.3 The forces' terms

The right hand side vector term of (2.1) represents the forces and moments acting on the ship. They have the components on the three axes, denoted by:

$$\underline{\mathcal{F}} = \begin{bmatrix} \mathcal{X} \\ \mathcal{Y} \\ \mathcal{Z} \end{bmatrix}, \qquad \underline{\mathcal{T}} = \begin{bmatrix} \mathcal{K} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} \tag{2.4}$$

An analytical derivation of the hydrodynamic characteristics of an arbitrary hull shape is a difficult task, due to the fact that complicated functions of accelerations, velocities, propeller propulsion, rudder excitations and external disturbances must be handled.

In the present section, the attention is focused on the so-called internal forces and moments *[Christensen, 1992]*, which are nonlinear function of states and hydrodynamic coefficients.

By restricting the discussion to the horizontal plane, the equation of motion (2.1) based on (2.3) and (2.4) becomes:

$$\begin{bmatrix} m(\dot{u} - rv - x_G r^2) \\ m(\dot{v} + ru + x_G \dot{r}) \\ I_z \dot{r} + m x_G(\dot{v} + ru) \end{bmatrix} = \begin{bmatrix} \mathcal{X} \\ \mathcal{Y} \\ \mathcal{N} \end{bmatrix} \tag{2.5}$$

where $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{N}$ represent the significant components of the forces and moment in the horizontal plane motion. The main difficulty in the modelling of the ship dynamics is to find suitable expressions of the hydrodynamic forces and moment, namely for $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{N}$. A general functional form for these forces and moment was suggested by *[Abkowitz, 1964]*:

$$\begin{aligned} \mathcal{X} &= \mathcal{X}(u, v, r, \delta, \dot{u}, \dot{v}, \dot{r}, T) \\ \mathcal{Y} &= \mathcal{Y}(u, v, r, \delta, \dot{u}, \dot{v}, \dot{r}) \\ \mathcal{N} &= \mathcal{N}(u, v, r, \delta, \dot{u}, \dot{v}, \dot{r}) \end{aligned} \tag{2.6}$$

where δ is the rudder deflection and T is the propeller thrust, corresponding to one single-screw propeller *[Blanke, 1981]*. Actually, the available thrust for ship's propulsion is $(1-t)T$, where t is a trust deduction factor, $0 \leq t \leq 1$, an expression between propeller and hull. Details can be found in *[Blanke, 1981]*.

The functional expressions (2.6) can be reduced into a useful mathematical form, by expanding each of them to a Taylor expansion about the steady state condition $u = u_0$, $v_0 = r_0 = \delta_0 = 0$ and $\dot{u}_0 = \dot{v}_0 = \dot{r}_0 = 0$ (nominal values). Consequently:

$$u = u_0 + \Delta u, \qquad v = \Delta v, \qquad r = \Delta r, \qquad \delta = \Delta\delta \tag{2.7}$$

where Δu, Δv, Δr, $\Delta\delta$ are small perturbations from the nominal values. The assumption that the mean forward speed is constant, implies that the model is only valid for small rudder angles and that the surge mode (surge equation) can be decoupled from the sway and yaw modes (steering equations). The ship steering equations of motions include the state variables v, r, ψ and the control input δ.

The coefficients in the Taylor expansions:

$$d\mathcal{X} = \frac{\partial \mathcal{X}}{\partial u}u + \frac{\partial \mathcal{X}}{\partial \dot{u}}\dot{u} + \frac{\partial \mathcal{X}}{\partial v}v + \ldots \tag{2.8}$$

are defined as hydrodynamic coefficients, for instance: $\mathcal{X}_u = \frac{\partial \mathcal{X}}{\partial u}$, $\mathcal{X}_{uu} = \frac{\partial \mathcal{X}}{\partial uu}$, ... There exist different methods (Planar Motion Mechanism, Strip Theory, System Identification) which are used to determine the hydrodynamic coefficients *[Källström, 1979]* and *[Fossen, 1994]*.

Based on the simplified equations (2.5) of the ship's motion, a set of nonlinear or linear ship models can be created.

2.4 Nonlinear models of the ship

Several different expressions of nonlinear models of ship dynamics are available in the technical literature. In this section, some commonly used model structures are reviewed.

In the marine engineers' literature, the non-dimensional form of the models is commonly used *[Källström, 1979]*. The non-dimensional expressions are preferred because they give similar magnitudes for the individual variables and coefficients, independent of ship speed and size (Appendix B). The parameters only depend on ship loading, trim and water depth *[Fossen, 1994]*. However, for notational convenience, the ship models presented in this chapter are in dimensional form.

An exemplification of the forces $\mathcal{X}, \mathcal{Y}$ and of the moment $\mathcal{N}$ is given in *[Abkowitz, 1980]*, using a third order Taylor expansion:

$$
\begin{aligned}
\mathcal{X} = {} & \mathcal{X}_0 + \mathcal{X}_u \dot{u} + \mathcal{X}_u \Delta u + \mathcal{X}_{uu} \Delta u^2 + \mathcal{X}_{uuu} \Delta u^3 + \mathcal{X}_{vv} v^2 + \mathcal{X}_{rr} r^2 \\
& + \mathcal{X}_{\delta\delta} \delta^2 + \mathcal{X}_{rv} r v + \mathcal{X}_{r\delta} r \delta + \mathcal{X}_{v\delta} v \delta + \mathcal{X}_{vvu} v^2 \Delta u + \mathcal{X}_{rru} r^2 \Delta u \\
& + \mathcal{X}_{\delta\delta u} \delta^2 \Delta u + \mathcal{X}_{rvu} r v \Delta u + \mathcal{X}_{r\delta u} r + \delta \Delta u + \mathcal{X}_{v\delta u} v \Delta u \\
& + (1-t)T + \mathcal{X}_{ext} \\
\\
\mathcal{Y} = {} & \mathcal{Y}_0 + \mathcal{Y}_u \Delta u + \mathcal{Y}_{uu} \Delta u^2 + \mathcal{Y}_r r + \mathcal{Y}_v v + \mathcal{Y}_{\dot{r}} \dot{r} + \mathcal{Y}_{\dot{v}} \dot{v} + \mathcal{Y}_\delta \delta \\
& + \mathcal{Y}_{rrr} r^3 + \mathcal{Y}_{vvv} v^3 + \mathcal{Y}_{\delta\delta\delta} \delta^3 + \mathcal{Y}_{rr\delta} r^2 \delta + \mathcal{Y}_{\delta\delta r} \delta^2 r + \mathcal{Y}_{rrv} r^2 v \\
& + \mathcal{Y}_{vvr} v^2 r + \mathcal{Y}_{\delta\delta v} \delta^2 v + \mathcal{Y}_{vv\delta} v^2 \delta + \mathcal{Y}_{\delta vr} \delta v r + \mathcal{Y}_{vu} v \Delta u + \mathcal{Y}_{ru} r \Delta u \\
& + \mathcal{Y}_{vuu} v \Delta u^2 + \mathcal{Y}_{ruu} r \Delta u^2 + \mathcal{Y}_{\delta u} \delta \Delta u + \mathcal{Y}_{\delta uu} \delta \Delta u^2 + \mathcal{Y}_{ext} \\
\\
\mathcal{N} = {} & \mathcal{N}_0 + \mathcal{N}_u \Delta u + \mathcal{N}_{uu} \Delta u^2 + \mathcal{N}_r r + \mathcal{N}_v v + \mathcal{N}_{\dot{r}} \dot{r} + \mathcal{N}_{\dot{v}} \dot{v} + N_\delta \delta \\
& + \mathcal{N}_{rrr} r^3 + \mathcal{N}_{vvv} v^3 + \mathcal{N}_{\delta\delta\delta} \delta^3 + \mathcal{N}_{rr\delta} r^2 \delta + \mathcal{N}_{\delta\delta r} \delta^2 r + \mathcal{N}_{rrv} r^2 v \\
& + \mathcal{N}_{vvr} v^2 r + \mathcal{N}_{\delta\delta v} \delta^2 v + \mathcal{N}_{vv\delta} v^2 \delta + \mathcal{N}_{\delta vr} \delta v r + \mathcal{N}_{vu} v \Delta u + \mathcal{N}_{ru} r \Delta u \\
& + \mathcal{N}_{vuu} v \Delta u^2 + \mathcal{N}_{ruu} r \Delta u^2 + \mathcal{N}_{\delta u} \delta \Delta u + \mathcal{N}_{\delta uu} \delta \Delta u^2 + \mathcal{N}_{ext}
\end{aligned}
\tag{2.9}
$$

where $\mathcal{X}_{ext}$, $\mathcal{Y}_{ext}$, $\mathcal{N}_{ext}$ denote the external disturbances on X, Y, Z axes. They are presented in details in Section 2.7 of this chapter.

Obviously this model is quite large, with more than 60 parameters and therefore it is inadequate for use with regards to the identification and control scope.

[Norrbin, 1970] has proposed a nonlinear mathematical model for ship maneuvering using the viscous cross flow concept. The original form of this model used the 'bis' - normalization system (Appendix B). In the following, the dimensional form of Norrbin's model is given:

$$
\begin{aligned}
\mathcal{X} &= \mathcal{X}_{uu} u^2 + \mathcal{X}_{uuuu} u^4 + \mathcal{X}_{vr} v r + \mathcal{X}_{rr} r^2 + \mathcal{X}_{u|v|vv} u|v|v^2 \\
& \quad + \mathcal{X}_{\delta\delta} \delta^2 + (1-t)T \\
\mathcal{Y} &= \mathcal{Y}_{\dot{v}} \dot{v} + \mathcal{Y}_{\dot{r}} \dot{r} + \mathcal{Y}_{ur} u r + \mathcal{Y}_{uur} u^2 r + \mathcal{Y}_{uv} u v + \mathcal{Y}_{uuv} u^2 v \\
& \quad + \mathcal{Y}_\delta \delta + \mathcal{Y}_{|v|v} |v| v + \mathcal{Y}_{|r|r} |r| r + \mathcal{Y}_{|v|r} |v| r + K_y T \\
\mathcal{N} &= \mathcal{N}_{\dot{v}} \dot{v} + \mathcal{N}_{\dot{r}} \dot{r} + \mathcal{N}_{uv} u v + \mathcal{N}_{ur} u r + \mathcal{N}_{uur} u^2 r + \mathcal{N}_{uuv} u^2 v + \mathcal{N}_\delta \delta \\
& \quad + \mathcal{N}_{|v|v} |v| v + \mathcal{N}_{|r|r} |r| r + \mathcal{N}_{|v|r} |v| r + \mathcal{N}_{vxs|r|} + K_\delta T
\end{aligned}
\tag{2.10}
$$

The terms $Y_{vv}|v|v$, $Y_{vr}|v|r$, $N_{vv}|v|v$, $N_{vr}|v|r$ represent the viscous effects in the steering equations.

A simplified form of Norrbin's nonlinear model, which retains the most important terms for steering and propulsion loss assessment, has been proposed in *[Blanke, 1981]* as follows:

$$
\begin{aligned}
\mathcal{X} &= \mathcal{X}_{uu} u^2 + \mathcal{X}_{vr} v r + \mathcal{X}_{rr} r^2 + \mathcal{X}_{\delta\delta} \delta^2 + (1-t)T + \mathcal{X}_{ext} \\
\mathcal{Y} &= \mathcal{Y}_{\dot{v}} \dot{v} + \mathcal{Y}_{\dot{r}} \dot{r} + \mathcal{Y}_{ur} u r + \mathcal{Y}_{uv} u v + \mathcal{Y}_{|v|v} |v| v + \mathcal{Y}_{|v|r} |v| r \\
& \quad + \mathcal{Y}_\delta \delta + \mathcal{Y}_{ext} \\
\mathcal{N} &= \mathcal{N}_{\dot{v}} \dot{v} + \mathcal{N}_{\dot{r}} \dot{r} + \mathcal{N}_{uv} u v + \mathcal{N}_{ur} u r + \mathcal{N}_{|v|v} |v| v + \mathcal{N}_{|v|r} |v| r \\
& \quad + \mathcal{N}_\delta \delta + \mathcal{N}_{ext}
\end{aligned}
\tag{2.11}
$$

It should be noted that all nonlinear models discussed so far in this chapter are based on the assumption that the ship's motion is restricted to the horizontal plane. Models which also deal with the roll motion can be found in *[Fossen, 1994]*, *[Son and Nomoto, 1981]* and *[Christensen and Blanke, 1986]*.

The present thesis will focus on the nonlinear model (2.11), described in *[Blanke, 1981]*. It is attractive for control purposes, due to the fact that it is simple and it retains the most important terms for steering. Taken (2.3), (2.4) and $u = u_0$ into account, the steering equations of the model (2.11) can be written as:

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \boldsymbol{A}\, u_o \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \boldsymbol{B}\, u_0^2\, \delta + \boldsymbol{C} \begin{bmatrix} v^3 \\ rv^2 \end{bmatrix} + \boldsymbol{D} \begin{bmatrix} Y_{ext} \\ N_{ext} \end{bmatrix} \tag{2.12}$$

where:

$$\begin{gathered} \boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 1 & 0 \end{bmatrix}; \qquad \boldsymbol{B} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix} \\ \boldsymbol{C} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ 0 & 0 \end{bmatrix}; \qquad \boldsymbol{D} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ 0 & 0 \end{bmatrix} \end{gathered} \tag{2.13}$$

where the coefficients are defined as in Appendix A.

The simulations described in the present work are based on the nonlinear model (2.11) for the Mariner Class Vessel[1]: a 16622 m^3 displacement, a 160 meters long ship with a design speed of 7.71 m/s (15 knots).

2.5 Linear models of the ship

For practical reasons (e.g. in order to establish the design parameters for control systems) one often makes certain assumptions and approximations about the physical system so that the linear system theory can be applied.

A linear approximation of the model of the ship is obtained if the higher order terms in the Taylor expansion (2.8) are neglected.

It is well-known that the surge equation of ship's dynamic model is inherently nonlinear *[Blanke, 1981]*, while linear terms are presented in the sway and yaw equations. Linearizations about the stationary solution $\dot{v} = \dot{r} = v = r = \delta = 0$ of nonlinear models will now be considered. The standard linear model is given by:

$$\begin{bmatrix} m - Y_{\dot{v}} & mx_G - Y_{\dot{r}} & 0 \\ mx_G - N_{\dot{v}} & I_z - N_{\dot{r}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} \tag{2.14}$$

[1] Ship's data are summarized in Appendix A.

$$= \begin{bmatrix} Y_v & Y_r - m u_0 & 0 \\ N_v & N_r - m x_G u_0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} Y_\delta \\ N_\delta \\ 0 \end{bmatrix} \delta$$

A criterion for asymptotic stability of the linear model (2.14) is provided in *[Abkowitz, 1964]*:

$$Y_v\,(N_r - m\,x_G\,u_0) - N_v\,(Y_r - m\,u_0) > 0 \tag{2.15}$$

Straight algebraic calculations on (2.14) leads to the standard differential state space model:

$$\dot{\underline{x}} = \boldsymbol{A}_l\,\underline{x} + \boldsymbol{B}_l\,\delta \tag{2.16}$$

where $\underline{x} = \begin{bmatrix} v & r & \psi \end{bmatrix}^T$, $\boldsymbol{A}_l = \boldsymbol{A}\,u_0$, $\boldsymbol{B}_l = \boldsymbol{B}\,u_0^2$. $\boldsymbol{A}$ and $\boldsymbol{B}$ are defined as described in (2.13). The values for the undimensional versions of the coefficients (A.3) for different ships are presented in *[Källström, 1979]*.

Based on (2.16), the input-output relation between the rudder angle δ and the heading angle ψ can be represented by the transfer function:

$$G_{\psi\,\delta} = \frac{K(1 + s\,T_3)}{s\,(1 + s\,T_1)(1 - s\,T_2)} \tag{2.17}$$

where K is the gain constant and T_i $(i = 1, 2, 3)$ are three time constants. The dependence between the parameters of the transfer function (2.17) and the parameters of the state space model (2.16) can be found e.g. in

[Fossen, 1994]. The transfer function (2.17) is commonly used in the design of autopilots. An approximation of the transfer function (2.17), valid only for low frequencies, has been proposed by *[Nomoto, 1957]*:

$$G_{\psi\,\delta} = \frac{K}{s\,(1 + s\,T)} \tag{2.18}$$

where the effective time constant T is equal to $T_1 + T_2 - T_3$. The coefficients of Nomoto model (2.18) can be determined from experimental data.

2.6 The steering machine

The rudder is actuated by means of the hydraulic steering machine (ship's actuator) which has nonlinear dynamics. A simplified representation of the steering machine can be found in *[Amerongen, 1982]*, see Figure 2.3.

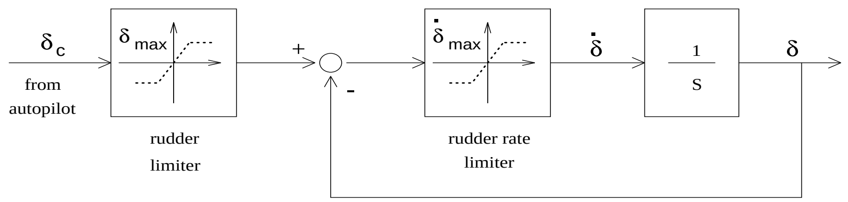

Figure 2.3. *Simplified diagram of the steering machine.*

Generally, the rudder angle and rudder rate limiters in Figure 2.3 are typically in the ranges:

$$\delta_{max} = 35 \ \ [\text{deg}], \qquad \dot{\delta}_{max} = 2\!-\!7 \ \ [\text{deg/s}] \tag{2.19}$$

For example, for the Mariner Class Vessel, the maximum deflection rudder angles must not be larger than $\delta_{max} = 35$ deg, while the maximum accepted rudder speed should not exceed $\dot{\delta} = 3.5$ deg/s.

The steering machine is usually faster than the ship and it can be modelled as a first order system as *[Amerongen, 1982]*:

$$\tau\,\dot{\delta} + \delta = \delta_c$$

where δ_c is rudder command and τ is the time constant of the steering machine. In control design, the dynamics in the steering machine may be disregarded ($\delta \approx \delta_c$).

2.7 Modelling the disturbances

So far the internal forces and moments, as complicated functions of accelerations, velocities, propeller propulsion, rudder excitations, have been discussed. In this section, the external forces which act on a ship, are described.

According to *[Van der Klugt, 1987]*, the external disturbances can be classified as follows:

- Additive disturbances – are external forces and moments which act additively on the ship. By including their effect, the total description

  of the ship model is extended by additional states (e.g. current, wind, waves).

- Multiplicative disturbances – affect the dynamics of the system (e.g. the depth of water, load condition, trim, speed changes). These disturbances can be regarded as time variant.

- Measurement disturbances – are due to incorrect measurements (e.g. noise on the position measurements).

Throughout the thesis, only the additive disturbances will be taken into account. From the class of disturbances causing additive effects in the system, the current, wind and waves will be considered.

The controllability of a ship is known to be significantly affected by the immediate environment *[Fossen, 1994]*. Many previous attempts on the design of a steering control system have suffered from inadequate modelling of the disturbances effects on the ship. The reason is that the modeling of the external disturbances, especially of the waves and wind, is rather complicated.

A general assumption *[Källström, 1979]* is that the disturbances can be modelled by forces and moments which are added to the calm sea ship models, outlined in the previous sections. This assumption, the expression of the principle of superposition, is a good approximation for most maritime control applications.

2.7.1 The principle of superposition

The responses of the ocean vehicle to the external disturbances are usually computed by applying the principle of superposition. The principle of superposition can be used in order to derive the linear and nonlinear equations of the ship's motion, in terms of environmental disturbances. The total external forces and moment, which are significantly for the horizontal plane motion, are:

$$\begin{aligned} \mathcal{X}_{ext} &= \mathcal{X}_{wind} + \mathcal{X}_{wave} + \mathcal{X}_{current} \\ \mathcal{Y}_{ext} &= \mathcal{Y}_{wind} + \mathcal{Y}_{wave} + \mathcal{Y}_{current} \\ \mathcal{N}_{ext} &= \mathcal{N}_{wind} + \mathcal{N}_{wave} + \mathcal{N}_{current} \end{aligned} \tag{2.20}$$

It should be noted that the superposition principle is not valid for large maneuvers *[Källström, 1979]*.

2.7.2 Wind

It is known that the solar radiation produces temperature and pressure differences in the atmosphere. Under the influence of these pressure differences, the air is accelerated and thus wind is produced *[Blanke, 1981]*.

Wind forces and moments acting on a ship can usually be expressed in terms of a mean wind speed $\overline{V}$ in combination with a wind spectrum, describing the variation of the turbulence of the wind. The mean wind is a constant homogeneous air current, which exposes the ship to a steady force over periods much longer than the effective time constants of the ship dynamics

[Blanke, 1981]. The turbulence is caused by the stochastic fluctuations of the air flow and it is therefore random in magnitude and direction.

Wind Spectrum

The atmospheric turbulence can quantitatively be analyzed using the power spectrum of the wind. The power spectrum is a representation of the distribution of the energy of the fluctuations in the wind with respect to the frequency.

One of the most frequently used spectral formulations for wind gusts is the empirical spectrum suggested in *[Davenport, 1971]*:

$$S(\omega) = \frac{8\pi k}{\omega}\overline{V}_z^2 \left(\frac{L\omega}{2\pi\overline{V}_{10}}\right)^2 \left[1 + (\frac{L\omega}{2\pi\overline{V}_{10}})^2\right]^{-4/3}, \quad \omega > 0 \tag{2.21}$$

where:

k	the surface drag coefficient equal with 0.001 for open sea *[Davenport, 1971]*.
L	a scale parameter, $L = 1200$ [m] *[Davenport, 1971]*.
ω	the frequency of the wind oscillations [rad/s].
$\overline{V}_{10}$	the average wind speed [m/s] at a level of 10 [m] above the water surface .
$\overline{V}_z$	is the local velocity z [m] above the sea surface (see *[Blanke, 1981]*):

$$\overline{V}_z = \overline{V}_{10}\,(z/10)^{1/7}$$

It should be noted that in (2.21), $S(\omega)$ represents the energy per unit frequency interval (or power spectral density) at frequency ω. Other useful spectral formulations can be found e.g. in *[Fossen, 1994]*.

The spectral density (2.21) appears to be approximately proportional to the square of the mean wind speed $\overline{V}_z$. The term "mean wind" refers to an average of the speed carried out over a period of a 10 minute duration to one hour *[Davenport, 1971]*.

The variance of the turbulence, i.e. the variance of the stochastic fluctuations of the air flow, is given by:

$$\sigma^2 = \int_0^{\infty} S(\omega)d\omega \tag{2.22}$$

Straight computation of the integral (2.22) yields that the variance of the turbulence is $\sigma^2 = (0.2\overline{V}_z)^2$ (see *[Källström, 1979]*).

A level of 10 meters above the water surface will be considered. In order to simplify the notation, the mean wind speed $\overline{V}_{10}$ will simply be denoted by $\overline{V}$.

Linear approximation of the wind spectrum

The turbulent wind on the ship motion can be regarded as a linear realization of a white noise, by writing the output of the turbulence model as:

$$\zeta(s) = H(s)e(t)$$

where $e(t)$ is a zero mean Gaussian white noise process with power spectrum:

$$\Phi_{ee}(\omega) = 1$$

and $H(s)$ is the transfer function to be determined. The one-side spectrum density of the output $\zeta(s)$ can be computed as:

$$\Phi_{\zeta\zeta} = 2\,|H(j\omega)|^2\, \Phi_{ee}(\omega) = 2\,|H(j\omega)|^2, \quad \omega > 0 \tag{2.23}$$

The main idea is to design a rational approximation $\Phi_{\zeta\zeta}$ for $S(\omega)$ such that, $\Phi_{\zeta\zeta}$ reflects the energy distribution of $S(\omega)$ in the frequency range ($\omega > 0$).

Linear approximations of the disturbances are usually preferred due to their simplicity and applicability. *[Fossen, 1994]* presents a linear 1st-order approximation for the turbulent wind. However, it is an empirical fact that the turbulent wind is usually properly approximated by a second order model:

$$H(s) = \frac{k_\zeta s}{(s+a_1)(s+a_2)} \tag{2.24}$$

Transforming this expression (2.24) to the time domain, the following linear state space model is obtained:

$$\begin{aligned} \begin{pmatrix} \dot{x}_1{}^{wind} \\ \dot{x}_2{}^{wind} \end{pmatrix} &= \begin{pmatrix} \theta_1 & \theta_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^{wind} \\ x_2^{wind} \end{pmatrix} + \begin{pmatrix} k_\zeta \\ 0 \end{pmatrix} e(t) \\ \zeta(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^{wind} \\ x_2^{wind} \end{pmatrix} \end{aligned} \tag{2.25}$$

where $\theta_1 = -(a_1 + a_2)$, $\theta_2 = -a_1 a_2$ and $e(t) \in \mathrm{N}(0,1)$. The states of the model x_1^{wind} and x_2^{wind} represent wind gust speed and acceleration, respectively.

Taking expressions (2.23) and (2.24) into account, the one-side spectrum is obtained as:

$$\Phi_{\zeta\zeta}(\omega) = \frac{2k_\zeta^2\omega^2}{(\omega^2 + a_1^2)(\omega^2 + a_2^2)}, \qquad \omega > 0 \tag{2.26}$$

The parameters of the linear approximation (2.25) can be determined in an analytical way *[Jensen, 1991]*, using the following assumptions:

- The centre frequencies of the true $S(\omega)$ and the approximated $\Phi_{\zeta\zeta}$ spectrums are identical:
  - for the true spectrum: $\frac{d(S(\omega))}{d\omega} = 0 \Rightarrow \omega_c = 0.06368^2\,\overline{V} \quad [s^{-1}]$
  - for the approximated spectrum: $\frac{d(\Phi_{\zeta\zeta})}{d\omega} = 0 \Rightarrow \omega_c = (a_1 a_2)^{1/2}$
- The power at the centre frequencies for the true and approximated spectrums is identical:
  - for the true spectrum: $S(\omega_c) = 1.9865\,\overline{V} \quad [m^2 s^{-1}]$
  - for the approximated spectrum: $\Phi_{\zeta\zeta}(\omega_c) = \frac{2k_\zeta^2}{(a_1+a_2)^2}$
- The variances obtained from both expressions of the spectrum are identical:
  - for the true spectrum: $\sigma^2 = \int_0^\infty S(\omega)d\omega = (0.2\overline{V})^2 \quad [m^2\,s^{-2}]$
  - for the approximated spectrum: $\sigma^2 = \int_0^\infty \Phi_{\zeta\zeta}(\omega) = \frac{k_\zeta^2\pi}{(a_1+a_2)}$

Thus the amplitude of the wind gusts can be modelled by the model (2.25), where the parameters are determined as :

$$\begin{aligned} \theta_1 &= -(a_1 + a_2) &&= -0.012819\,\overline{V} \\ \theta_2 &= -a_1 a_2 &&= -1.648 \times 10^{-5}\,\overline{V}^2 \\ k_\zeta &= 0.0127756\,\overline{V}^{3/2} \end{aligned} \tag{2.27}$$

Note that the realization of $\zeta(t)$ only depends on $\overline{V}$ and $e(t)$. The empirical (2.21) and the approximated (2.26) spectrum are plotted in Figure 2.4, for different mean wind speeds.

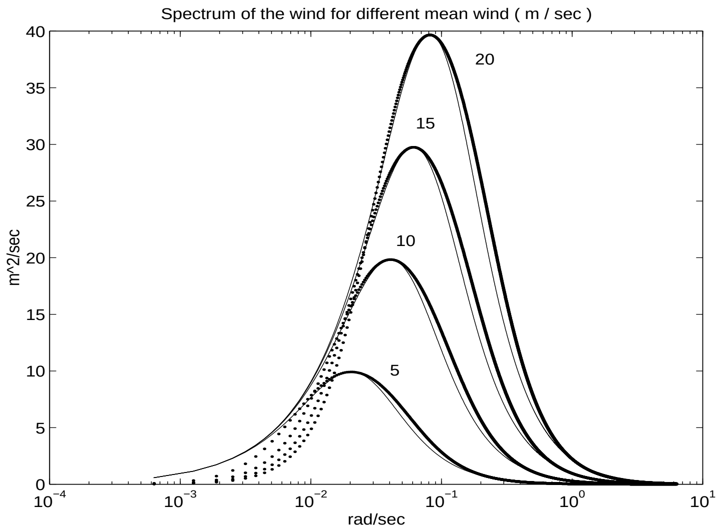

Figure 2.4. *The empirical and the approximated spectrum for different mean wind speeds* $\overline{V} = [5; 10; 15; 20]$.

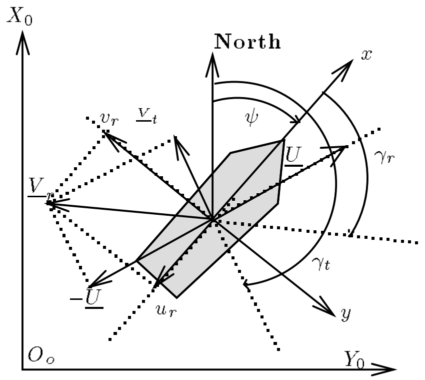

Figure 2.5. *Relative wind speed V_r and direction γ_r.*

The wind gusts cannot be compensated by the control system, since the dynamic of the ship is usually too slow compared with the gusts. However, slowly varying wind forces can be counteracted by a controller, when the average wind speed $\overline{V}$, the direction γ_t of the true wind (see Figure 2.5), and the size and the shape of the ship are known. This requires the wind forces and the wind moment coefficients to be known with sufficient accuracy.

Wind forces and moments

The total wind speed V_t (the true wind) contains a slow component $\overline{V}$ (mean wind speed) and a high-frequency component ζ (wind gust modelled by (2.25)):

$$V_t = \overline{V} + \zeta$$

The resulting wind forces and moment acting on a ship are usually defined in terms of relative wind speed V_r m/s and angle γ_r rad. According to Figure 2.5:

$$\begin{aligned} V_r &= \sqrt{u_r^2 + v_r^2} \\ \gamma_r &= \tan^{-1}(v_r/u_r) + \pi \qquad\qquad\qquad\quad if \;\; u_r \geq 0 \\ \gamma_r &= \tan^{-1}(v_r/u_r) \;\; (+2\pi \;\; if \;\; v_r > 0) \;\; if \;\; u_r < 0 \end{aligned} \tag{2.28}$$

where the components of V_r on the x and y directions are:

$$\begin{aligned} u_r &= V_t \cos(\gamma_t - \psi - \pi) - (u + u_c) \\ v_r &= V_t \sin(\gamma_t - \psi - \pi) - (v + v_c) \end{aligned} \tag{2.29}$$

where (u, v) and (u_c, v_c) are the ship and current velocity components *[Källström, 1979]*, respectively. V_t is the speed of the true wind $V_t = \overline{V} + \zeta$, namely it is the result of adding the mean wind $\overline{V}$ and the turbulent component ζ according to (2.25).

In the technical literature (see *[Isherhood, 1972]* and *[Berkelom et al., 1974]*), different methods for computation of the wind force and moment vector can be found. The forces induced by the mean wind have been investigated by wind tunnel tests on different ships. According to *[Berkelom et al., 1974]*, the wind forces (surge and sway) and moment (yaw) can be expressed as:

$$
\boxed{\begin{aligned}
X_{wind} &= \tfrac{1}{2}\rho_a V_r^2 C_x(\gamma_r) A_f \qquad [N] \\
Y_{wind} &= \tfrac{1}{2}\rho_a V_r^2 C_y(\gamma_r) A_s \qquad [N] \\
N_{wind} &= \tfrac{1}{2}\rho_a V_r^2 C_n(\gamma_r) A_s L_{oa} \quad [Nm]
\end{aligned} \qquad (2.30)}
$$

where:

V_r	relative wind speed.
γ_r	relative wind angle to the ship.
ρ_a	density of the air above the water surface ($\rho_a = 1.23$ [kg/m^3]).
A_f	projected front area of the ship above the waterline [m^2].
A_s	projected side area of the ship above the waterline [m^2].
L_{oa}	overall length of the ship [m].
C_x, C_y, C_n	wind coefficients.

For those ships that are not too asymmetrical with respect to the xz and yz planes, *[Källström, 1979]* suggests some approximation forms for the wind coefficients:

$$
\begin{aligned}
C_x(\gamma_r) &\approx \hat{C}_x \cos(\gamma_r) \\
C_y(\gamma_r) &\approx \hat{C}_y \sin(\gamma_r) \\
C_n(\gamma_r) &\approx \hat{C}_n \sin(2\gamma_r)
\end{aligned} \tag{2.31}
$$

It is observed that C_x, C_y, C_n are given as functions of γ_r. The approximated values for the wind coefficients are based on wind tunnel tests.

Empirical variations of these coefficients are presented in *[Åge, 1971]* for different types of ships. For instance, the variation of the wind coefficients for a cargo ship is shown in Figure 2.6. Based on this figure the wind

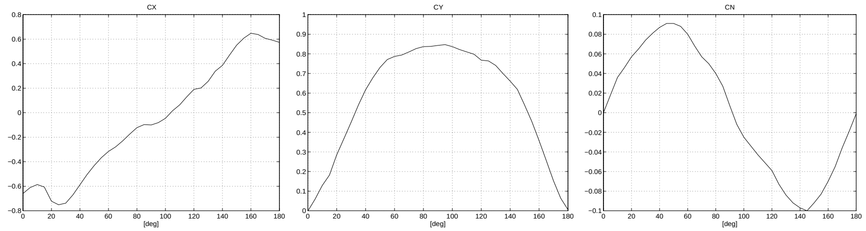

Figure 2.6. *Variation of the wind coefficients C_x, C_y, C_n with respect to the relative angle for a cargo ship.*

coefficients can be approximated by:

$$
\begin{aligned}
C_x(\gamma_r) &= -0.64 \cdot \cos(\gamma_r) - 0.07 \\
C_y(\gamma_r) &= 0.85 \cdot \sin(\gamma_r) \\
C_n(\gamma_r) &= 0.09 \cdot \sin(2\gamma_r)
\end{aligned} \tag{2.32}
$$

2.7.3 Waves

The generation of the waves is rather complex. A proper characterization of the real sea surface is difficult to obtain. With respect to the design of a control system, it is sufficient to assume a simplified description of the waves by considering only unidirectional linear waves. Based on this assumption and on the superposition principle, some simplified models of the waves forces on the ship can be determined. Concerning the modelling of the waves, there exist two typical approaches:

- Regular waves – where the sea motion is seen as a simple two-dimensional sinusoidal wave train over an infinite water surface with infinite depth. This interpretation is based on the empirical observation, that the motions generated by the waves have a strong periodic component (for details see e.g. *[Källström, 1979]*). The characterization of the sea level as a train of regular waves, is however, an approximation.

- Irregular waves – where the stochastic nature of the waves is taken into account. According to this approach, the sea level can be modelled as 1) a superposition of a large number of regular waves of different amplitudes, frequencies, phase angles and directions of propagations or as 2) a stochastic process. In the case when the sea level is regarded as a stochastic process, the spectral density should be available. Thus the model for the variation of the wave elevation can be determined from the spectral density of the waves. The irregular waves are so with respect to amplitude and distance between crests. In *[Blanke, 1981]*, the long crested seaway is described by a one-dimensional amplitude spectrum.

Since the waves usually appear due to the wind, the frequency and the steady state amplitude of the waves depend on the mean value of the wind speed $\overline{V}$. In the following the stochastic nature of the waves is considered.

The empirical wave spectrum

The profile and the slope of the waves can be modelled as stochastic processes. An internationally accepted wave spectral formulation is the Pierson-Moskowitz power spectra density function *[Fossen, 1994]*. This spectrum is

empirical and it is given for the fully developed wind generated sea, as follows:

$$S(\omega) \quad = \quad A_s\ \omega^{-5} e^{-B_s \omega^{-4}} \qquad [m\ ^2\mathrm{s}] \tag{2.33}$$

where $A_s = 0.78\ \mathrm{m}^2/\mathrm{s}^4$, $B_s = 3.11 h^{-2}\ 1/\mathrm{s}^{-4}$, h denotes the significant wave height and ω the wave frequency. The significant wave height h is defined as the average distance from top to the bottom of the largest third of the waves *[Christensen, 1992]*.

The modal frequency ω_0 for the spectrum is found by requiring that:

$$\left(\frac{dS(\omega)}{d\omega}\right)_{\omega=\omega_0} = 0 \tag{2.34}$$

Straightforwardly this implies that:

$$\begin{aligned} \omega_0 &= \sqrt[4]{\tfrac{4\,B_s}{5}} &= 1.26\,h^{-1/2} \qquad &[\mathrm{rad/s}] \\ T_0 &= 2\pi\,\sqrt[4]{\tfrac{5}{4\,B_s}} &= 5\,h^{1/2} \qquad &[\mathrm{s}] \end{aligned} \tag{2.35}$$

where T_0 is the modal period (dominating wave period). The maximum value of the spectrum $S(\omega)$ is then:

$$\max\{S(\omega)\} = S(\omega_0) = \frac{5\,A_s}{4\,B_s\,\omega_0}\ \exp\left(-\frac{5}{4}\right)$$

In Figure 2.7, the empirical power spectrum defined by (2.33), is illustrated with respect to the mean wind $\overline{V}$ (continuous line). Figure 2.7 illustrates

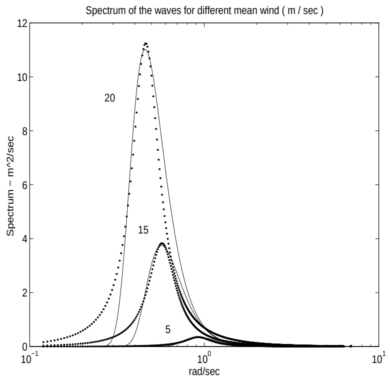

Figure 2.7. *Empirical and approximated waves' spectrums for mean winds $\overline{V} = 5, 15, 20$ [m/s].*

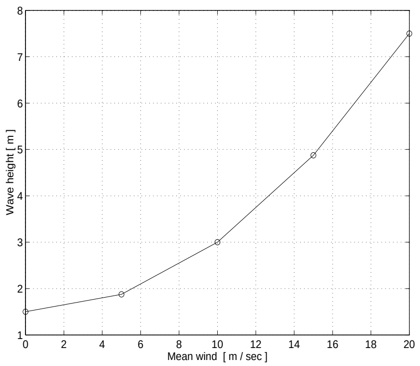

Figure 2.8. *Mean wind and significant wave height.*

that as the mean wind increases, the bulk of energy in the spectrum is shifted towards lower frequencies. Typical values for the peak frequency are 0.35-1.2 rad/s. Based on the empirical observation, an approximative relationship between mean wind $\overline{V}$ and significant wave height h has been proposed by *[Price and Bishop, 1974]*:

$$h(\overline{V}) = 0.015\ \overline{V}^{\,2} + 1.5 \tag{2.36}$$

This relation is represented in Figure 2.8. It is observed that the significant wave height increases with the mean speed. The interpretation of Figures 2.7 and 2.8 leads to the explanation that, high amplitude waves have long wavelength (low frequency). The large waves have more energy than the small waves.

Similar to (2.36), an approximative relationship between mean wind speed $\overline{V}$ m/s and wave period T_w seconds can be derived:

$$T_w(\overline{V}) = -\,0.0014\ \overline{V}^{\,3} + 0.042\,\overline{V}^{\,2} + 5.6 \tag{2.37}$$

It should be noticed that the empirical expressions (2.36) and (2.37) are valid for $0 < \overline{V} \leq 20$ m/s *[Källström, 1979]*.

Based on (2.33), the variance of the wave amplitude is given by:

$$\sigma^2 = \int_0^\infty S(\omega) d\omega = \frac{A_s}{4\,B_s} = \frac{0.78}{12.44}\,h^2$$

Linear approximation of the wave spectrum

Linear wave model approximations are usually preferred by the system engineers of ship control, owing to their simplicity and applicability.

Based on the representational theorem *[Poulsen, 1995]*, a linear approximation to the Pierson-Moskowitz spectral density function $S(\omega)$ can be found by writing the output $\xi(s)$ from the wave model as:

$$\xi(s) = H(s)e(s)$$

where $e(s)$ is a zero-mean Gaussian white noise process with unity power spectrum and $H(s)$ is a transfer function to be determined. This transfer function is estimated by minimizing the error between the empirical spectrum $S(\omega)$ and the approximated spectrum $\Phi_{\xi\xi}(\omega)$:

$$\Phi_{\xi\xi}(\omega) = |H(j\omega)|^2 \Phi_{ee}$$

A damping term is usually introduced in the wave model *[Fossen, 1994]* to better fit the shape of the Pierson-Moskowitz spectrum. This model is written as:

$$H(s) = \frac{k_w\, s}{s^2 + 2\,\zeta\,\omega_0\, s + \omega_0^2} \tag{2.38}$$

where k_w is the gain constant, ζ is the damping coefficient while ω_0 is the dominating (modal) wave frequency, defined in (2.35). Using the notation $\theta_1 = 2\,\zeta\,\omega_0$ and $\theta_2 = \omega_0^2$, it implies the following state space form of (2.38):

$$\begin{aligned} \begin{pmatrix} \dot{x_1}^{wave} \\ \dot{x_2}^{wave} \end{pmatrix} &= \begin{pmatrix} -\theta_1 & -\theta_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^{wave} \\ x_2^{wave} \end{pmatrix} + \begin{pmatrix} k_w \\ 0 \end{pmatrix} e(t) \\ \xi(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^{wave} \\ x_2^{wave} \end{pmatrix} \end{aligned} \tag{2.39}$$

where $\xi(t)$ represents the wave profile (the height), while $s(t) = k_w \dot{\xi}(t)$ is the slope of the wave. The resulting spectrum may then be written as:

$$\Phi_{\xi\xi}(\omega) = \frac{2\,k^2 \omega^{\,2}}{\omega^{\,4} + ({\theta_1}^{\,2} - 2\,\theta_2)\omega^{\,2} + {\theta_2}^{\,2}}\,\Phi_{ee}(\omega)$$

In order to determine the parameters for the model (2.39) it is necessary to perform a linear regression, by the minimization of the cost function:

$$J = \int [S(\omega) - \Phi_{\xi\xi}(\omega)]^2 d\omega$$

However, in the case of the low order models, an analytical approach can be applied. For the second order model (2.39), straightforward analytical computations yield the following parameters:

$$\begin{aligned} \theta_1 &= \tfrac{2\,\omega_0}{5\,\pi}\ exp(\tfrac{5}{4}) \\ \theta_2 &= \omega_0^2 \\ k_w &= \tfrac{1}{\pi}\sqrt{\tfrac{A_s}{10\,B_s}\,\omega_0}\ exp(\tfrac{5}{8}) \end{aligned} \tag{2.40}$$

It is noted that the parameters of the approximation (2.39) depend on the modal frequency ω_0, defined in (2.34), and thus on the significant wave height h. Taking the relations (2.33) and (2.35) into account, the parameters of (2.39) can be further expressed as:

$$\begin{aligned} \theta_1 &= 0.5599\, h^{-1/2} \\ \theta_2 &= 1.5876\, h^{-1} \\ k_w &= 0.1057\, h^{3/4} \end{aligned} \tag{2.41}$$

Note also that the realization of the profile of the waves $\xi(t)$ only depends on the significant wave height h and the white noise signal $e(t)$.

Frequency of encounter

The spectrum described by formula (2.33), however, does not represent the actual waves spectrum encountered by a ship moving with the speed U and the heading angle relative to the propagation of the sea waves β. The definition of the encounter angle β is shown in Figure 2.9. Thus:

$$\beta = \pi - (\gamma - \psi)$$

where γ is the direction of the true wave propagation and ψ is the heading angle of the ship.

The effect of the waves on the ship depends on the ship's speed, the direction of waves and the heading angle of the ship. Thus, for a ship moving with forward speed U, the modal frequency ω_0 from (2.35) should be modified *[Blanke, 1981]* according to:

$$\begin{aligned} \omega_e &= \omega_0 - k_w\ u\ \cos\beta + k_w\ v\ \sin\beta \\ &\approx \omega_0 - \frac{\omega_0^2}{g}\ U\cos\beta \end{aligned} \tag{2.42}$$

where (assuming deep water) $\omega_0 = \frac{2\pi}{T_w} = \sqrt{g\ k_w} = \sqrt{\frac{2\ \pi\ g}{\lambda}}$, $k_w = \frac{2\pi}{\lambda} = \frac{4\ \pi^2}{g\ T_w^2}$ (the wave number) and:

- ω_e encounter frequency [rad/s].
- λ wave length (distance between consecutive crests of the wave.

- g acceleration of gravity [m/s^2].
- U total speed of ship [m/s] while (u,v) are the ship's velocity components.
- β the angle between the direction of the waves and the heading of the ship.

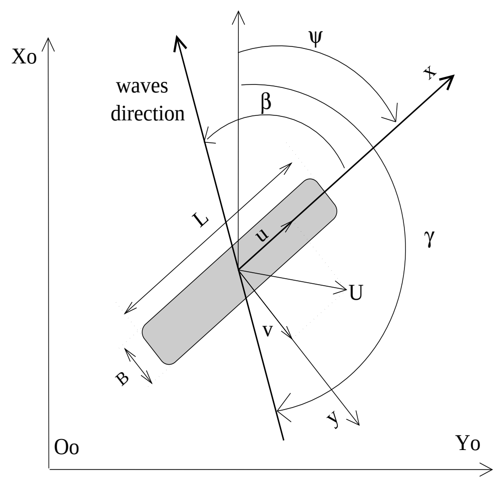

Figure 2.9. *The rectangular parallelepiped in a regular sea.*

Depending on the encounter angle β, a certain situation, which describes the encountered waves relative to the ship, has a specific name, as is illustrated in Figure 2.10 (adopted from *[Reid and Moore, 1982]*). Thus, the case when the encounter angle $\beta = 0$ deg, namely when the direction of the ship and the direction of the wave are the same, is called "following seas". This means that the encounter frequency ω_e in (2.42) is smaller than the modal frequency of the wave ω_0 (the ship runs away).

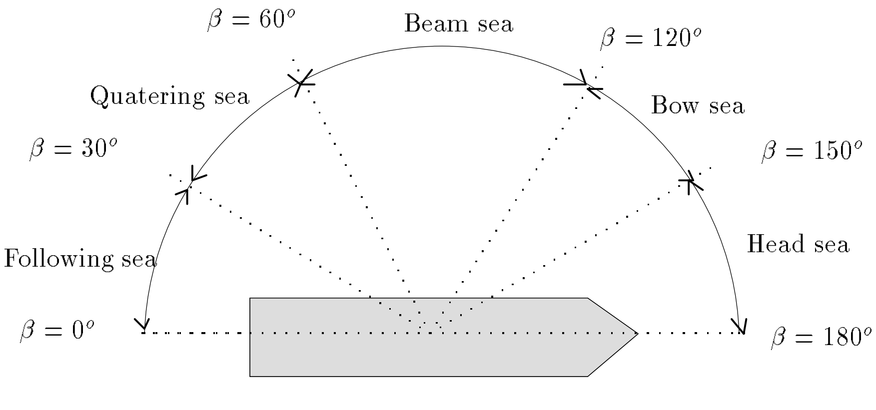

Figure 2.10. *The ship and the encounter angle.*

From (2.42) it is observed that the encounter frequency can be negative for large values of U. Negative values for ω_e can appear in the case of seas aft of the beam ($\beta < 90$ deg), depending on the ship's speed. If $\beta = 90$ deg the encounter frequency ω_e is the same as the frequency of the waves ω_0 (beam seas). For seas in front of the beam ($\beta > 90$ deg), the spectrum is shifted towards the high frequencies and is slightly attenuated. $\beta = 180$ deg represents head seas, the case when the ship and waves directions are opposite.

It should be noted that the wave spectrum for a heading controlled ship moving at speed $U > 0$ should incorporate the encounter frequency ω_e. For instance, the approximation model (2.38) can be rewritten as:

$$H(s) = \frac{k_w\, s}{s^2 + 2\,\zeta\,\omega_e\, s + \omega_e^2} \tag{2.43}$$

where ω_0 was replaced by ω_e. Similarly, the parameters θ_1, θ_2, k_w from (2.40) can analytically be determined. More details concerning the seaway encountered by the shipcan be found in *[Reid and Parent, 1982]* and *[Blanke, 1981]*.

Wave forces and moments

To derive the expressions for the forces and moments induced by the waves, a set of assumptions *[Källström, 1979]* should be specified: (1) the forces and moments only result from water pressure acting on the wetted surface, (2) the wave field is not disturbed by the ship and (3) the wave pressure appears as a fluctuation of the pressure distribution below the water surface, whereas the water surface itself is assumed to be undisturbed.

For a ship where the wetted part is a rectangular parallelepiped with length L, breadth B and draught T (see Figure 2.9), the following expressions for the forces and moment induced by the waves *[Källström, 1979]* can be defined:

$$\boxed{\begin{aligned} X_{wave} &= 2\,a\,B\,\tfrac{\sin b \sin c}{c}\,s(t) && [N] \\ Y_{wave} &= -2\,a\,L\,\tfrac{\sin b \sin c}{b}\,s(t) && [N] \\ N_{wave} &= a\,k_w\,M\;\zeta(t) && [Nm] \end{aligned}} \tag{2.44}$$

where:

$$\begin{array}{lcl} a & = & \rho_w\, g\,(1 - e^{-k_w\,T})/k_w\,^2 \\ b & = & k_w\, L/2\, \cos\beta \\ c & = & k_w\, B/2\, \sin\beta \\ M & = & \left[B^2 \sin b\, \frac{c\cos c - \sin c}{c^2} - L^2 \sin c\, \frac{b\cos b - \sin b}{b^2}\right] \end{array} \qquad (2.45)$$

where ρ_w is the density of the water and g is the gravitational acceleration. For regular waves, the profile and slope can be calculated as:

$$\xi(t) \;=\; \frac{h}{2}\,\cos(\omega_e\, t), \qquad s(t) \;=\; \frac{k_w\, h}{2}\,\sin(\omega_e\, t) \qquad (2.46)$$

where the significant wave height h is determined with the empirical relation (2.36). Otherwise, for the irregular stochastic waves, the wave profile is realized as (2.39) and the wave slope as $s(t) = k_w\dot{\xi}(t)$. The expressions (2.44) show the dependence of the forces and the moment on the characteristic of the seaway environment (wavelength and angle of attack from waves) and on the geometry of the ship's hull.

2.7.4 Currents

Currents are mainly generated by the atmospheric wind system over the sea surface, by the heat exchange at the sea surface. As in *[Källström, 1979]*, in the present thesis the current is assumed to be constant and homogeneous. Therefore, the current is modelled as a speed vector which has to be added to the speed vector of the ship. Thus the hydrodynamic forces and moments

on the ship are not affected by the current. Only the inertial speed of the ship is influenced.

A current is generally characterized by its speed V_c and by its direction γ_c. The velocity components of the current on the x and y axis respectively, are:

$$\begin{array}{lcl} u_c & = & V_c\, \cos(\gamma_c - \psi - \pi) \\ v_c & = & V_c\, \sin(\gamma_c - \psi - \pi) \end{array} \qquad (2.47)$$

They are sketched in Figure 2.11.

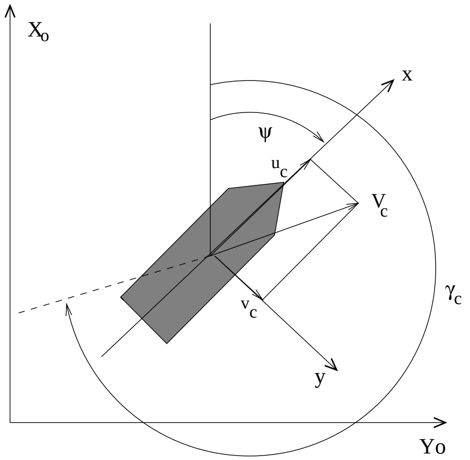

Figure 2.11. *Definition of the velocity V_c and the direction γ_c of the current.*

The forces and moment of the currents on the ship are:

$$
\begin{aligned}
X_{current} &= m\, v_c\, r && [N] \\
Y_{current} &= -m\, u_c\, r && [N] \\
N_{current} &= -m\, x_G\, u_c\, r && [Nm]
\end{aligned}
\tag{2.48}
$$

Once the expressions (2.30), (2.44), (2.48) for the disturbances' forces and moments are determined, the principle of superposition (2.20) can be applied. Thus the obtained external forces are added to the internal forces in order to model the total forces which act on a ship, namely the right hand side vector term of (2.5). Then, the expression (2.5) constitutes a suitable model for computer simulation of the steering of a large ship.

2.8 Summary

The ship model structures presented in this chapter are mainly intended for computer simulation and for control system design in combination with system identification. In the case when an accurate prediction of the sea-keeping properties is required, more advanced modelling techniques should be used. The described models are confined to motions in the horizontal plane only, by assuming that the coupling effects of roll, pitch and heave motions are negligible.

The disturbances acting on the ship are modelled by forces and moments which, based on the principle of superposition, are added to the calm sea ship models. Simple models for wind, waves and current are discussed. Spectral formulations are used in the wind and the waves' modelling. The current is considered homogeneous and constant.

Some standard references on ship models, wave, wind and current with application to ship control are *[Källström, 1979]*, *[Fossen, 1994]* and *[Blanke, 1981]*.

Chapter 3

On model reduction

This chapter is mainly concerned with some techniques on how to obtain a reduced order mathematical model of a given large order model, for the purposes of synthesis, analysis and implementation of control systems. Exemplification of the presented model reduction techniques is done on a model of the Mariner Class Vessel, described in Chapter 2. As the main goal of the model reduction is to generate a reduced order controller, in this chapter, for exemplification, a an LQG controller is designed on the obtained simplified linear model.

Over the years, many approaches to the model reduction problem has been proposed *[Adamjan et al., 1978]*, *[Moore, 1981]*, *[Glover, 1984]* and *[Enns, 1984b]*. However, despite the strength of the theory, there does not exist a universal model reduction algorithm, because the relative importance

of various plant characteristics is highly dependent upon the application. The best one can hope for is a good set of tools and some reliable guidelines for using them.

Two model reduction techniques will be discussed throughout this chapter:

- Balanced Truncation Approximation (BTA) – low order models are simply obtained by discarding those states in a balanced realization that correspond to small Hankel singular values.

- Hankel Norm Approximation (HNA) – low order models are obtained by a straightforward minimization of the Hankel norm of the error.

3.1 Motivation

In many practical situations, for the analysis and the implementation of a controller, it is often desired to approximate the high order model by a lower order model in such a way that the control relevant dynamics are preserved in the low order model. This is mathematically done by minimization of a suitable norm. The reduced order model must characterize the physical system with sufficient fidelity in such a way that the performance objectives for the controlled physical system can be also met by the controlled reduced order model. The control design is then based on the low order model and thus, the obtained reduced order controller is computationally less demanding and possibly numerically more reliable.

It must be stressed that the model reduction and the control problem are not independent of each other. During the last years, the interconnection between the way a (reduced) model is generated and the design of the controller became a very important research activity. This interconnection is emphasized and analyzed in Chapter 7, where an inductive way in obtaining a reduced order model, namely system identification approach, is used in connection with the controller design.

Even though there is a strong connection between reduction and control, in the literature two different concepts exist : model reduction and controller reduction. However, in connection with the control theory, they point towards the same result (a low order controller), even though they are based on different strategies, as it is shown in Figure 3.1:

- Model reduction is based on open loop considerations. The high order model is reduced. The reduced order model is then used in the design of a low order controller.

- Controller reduction is related more or less to the closed loop aspects, i.e. it should e.g. preserve the close loop stability and performances.

Both strategies have advantages and disadvantages which commute from one to another. Model reduction has e.g. the advantage that the computation of the controller is simplified, while the controller reduction has the advantage that the reduction is preserved, starting from closed loop considerations.

In practice, the main reasons for using the model/controller reduction philosophies consist in the following:

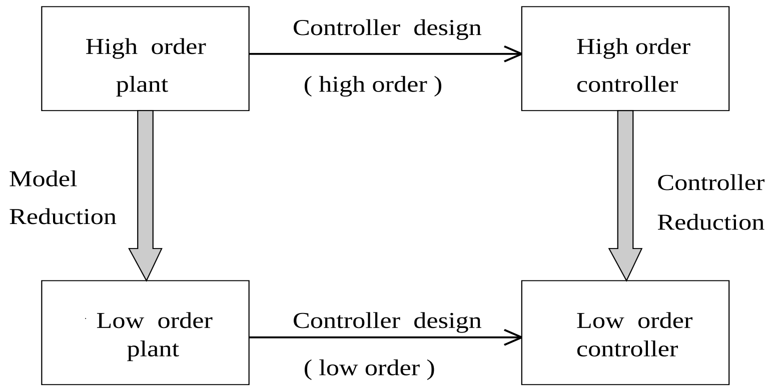

Figure 3.1. *Low order design strategies.*

- Low-order models simplify the understanding of a system, in the sense that they focus on the most significant modes of the system. An understanding of the minimum number of states involved in the behavior of a system is important in order to estimate the significant minimal model order, e.g. especially in the case of adaptive control.

- Reduced computational efforts are implied in the simulation problems and the numerical drawbacks can to a certain extent be avoided. This is e.g. the case of stiff systems, where very fast dynamics are mixed with slow dynamics and where, therefore, numerical problems typically exist.

- The design of the controller is numerically more efficient.

- Control laws are more simple and hardware requirements are reduced.

The present chapter is focused on the model reduction problem, which represents an analytical approach to reduce the order of a model, based on physical knowledge about the system.

3.2 Prelude to model reduction

In order to analyze the afore-mentioned model reduction techniques, some basic concepts (controllability and observability matrices, grammian matrices, Hankel singular values, Hankel matrix, error norms), used in connection with the model reduction, are firstly reviewed, due to their role in the reduction of the dynamic models.

3.2.1 Controllability and observability matrices

Consider a linear time invariant (LTI) system given by:

$$\begin{aligned} \underline{\dot{x}} &= \boldsymbol{A}\ \underline{x} + \boldsymbol{B}\ \underline{u}, \quad \underline{x}_{t_0} = \underline{x}_0 \\ \underline{y} &= \boldsymbol{C}\ \underline{x} + \boldsymbol{D}\ \underline{u} \end{aligned} \tag{3.1}$$

where $\underline{x} \in R^n$, $\underline{y} \in R^m$ and $\underline{u} \in R^p$, and $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$ are time invariant matrices of compatible sizes. n, m, p are the order, the number of outputs and the number of inputs of the system (3.1), respectively. Description (3.1) might be a linearization of a nonlinear system, approximation which is only reasonable in a specific region around the operating point.

It is well known that stability, controllability and observability are fundamental concepts of the state space description of the dynamic systems.

The controllability and observability matrices can be expressed in a compact way, as in *[Poulsen, 1995]*:

$$\boldsymbol{W}_c(\tau) \triangleq \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \cdots & \boldsymbol{A}^{\tau-1}\boldsymbol{B} \end{bmatrix} \tag{3.2}$$

$$\boldsymbol{W}_o(\tau) \triangleq \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\ \boldsymbol{A}^{\tau-1} \end{bmatrix} \tag{3.3}$$

where τ represents the length of the horizon on which the controllability and the observability of a system, respectively, are checked:

- $\tau = n$ corresponds to the controllability, the observability matrices of the system, denoted by $\overline{\boldsymbol{W}}_c$, $\overline{\boldsymbol{W}}_o$.

- $\tau > n$ is the case of the finite controllability, respectively the observability matrices, denoted by $\boldsymbol{W}_c$, $\boldsymbol{W}_o$.

- $\tau = \infty$ is the case of infinite controllability, respectively the observability matrices, denoted by $\mathcal{W}_c$, $\mathcal{W}_o$.

The realization $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D})_n$ is said to be minimal if it is completely controllable and observable *[Kailath, 1980]*. Otherwise, it is said to be non-minimal, case when, for SISO systems[1], the uncontrollable or unobservable states emerge as common zeros and poles in the transfer function:

$$G(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} \tag{3.4}$$

This means that the transfer function of a dynamic system is equal to its controllable and observable part $(\boldsymbol{A}_{co},\boldsymbol{B}_{co},\boldsymbol{C}_{co},\boldsymbol{D})_n$:

$$G(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} = \boldsymbol{C}_{co}(s\boldsymbol{I} - \boldsymbol{A}_{co})^{-1}\boldsymbol{B}_{co} + \boldsymbol{D} \tag{3.5}$$

even though their initial behavior is very different.

It should be pointed out that in the remaining part of the chapter, the representation $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D})_n$ will be considered to be stable, controllable and observable.

Besides the complete uncontrollable/unobservable states, there could exist some weak controllable/observable states. The existence of these states can be outlined by a bad conditional number of matrices $\boldsymbol{W}_c$, respectively $\boldsymbol{W}_o$. The result of this aspect, can be viewed by the fact that it can still be very difficult to control a linearized form of a nonlinear system, even though all uncontrollable and unobservable states were previously removed.

[1] In MIMO case, the controllability and observability aspects are more complicated – see *[Kailath, 1980]* for details.

3.2.2 Grammian matrices

Usually, the computation of the rank of a matrix is very fragile from a numerical point of view, especially in the case when the matrix has a tendency to become highly ill-conditioned, as e.g. is the case of controllability $\boldsymbol{W}_c$ and observability W_o matrices. A robust way to analyze the controllability and observability of a system is to look at the size of the singular values of the grammian matrices. For example, the number of the small singular values of the controllability grammian, will indicate the number of weak controllable states in the system. Some important properties of the grammians are pointed out in the following.

Assuming that the system is described by the linearized time invariant representation $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D})_n$, as in equation (3.1), the controllability grammian $\boldsymbol{P}$ is defined as:

$$\boldsymbol{P} \triangleq \int_0^\infty e^{\boldsymbol{A}t}\,\boldsymbol{B}\boldsymbol{B}^T\,e^{\boldsymbol{A}^T t}dt \tag{3.6}$$

and the observability grammian $\boldsymbol{M}$ as:

$$\boldsymbol{M} \triangleq \int_0^\infty e^{\boldsymbol{A}^T t}\,\boldsymbol{C}^T\boldsymbol{C}\,e^{\boldsymbol{A}t}dt \tag{3.7}$$

It is well known that the grammians $\boldsymbol{P}$ and $\boldsymbol{M}$ can be computed from the Lyapunov equations[1]:

[1] The Lyapunov equation has a unique solution if the system is asymptotically stable *[Kailath, 1980]*.

$$\begin{aligned} \boldsymbol{AP} + \boldsymbol{PA}^T + \boldsymbol{BB}^T &= 0 \\ \boldsymbol{A}^T\boldsymbol{M} + \boldsymbol{MA} + \boldsymbol{C}^T\boldsymbol{C} &= 0 \end{aligned} \tag{3.8}$$

It can be shown, see *[Glover, 1984]*, that the positive definite character of the grammians involves some important aspects:

- $\boldsymbol{P} > 0 \Longleftrightarrow (\boldsymbol{A}, \boldsymbol{B})$ is completely controllable
- $\boldsymbol{Q} > 0 \Longleftrightarrow (\boldsymbol{A}, \boldsymbol{C})$ is completely observable

These implications actually provide a more reliable way (i.e. based on the singular values of the grammians) to test the controllability and observability of the system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$. A zero eigenvalue of the controllability grammian $\boldsymbol{P}$ implies $(\boldsymbol{A}, \boldsymbol{B})$ is uncontrollable, while a zero eigenvalue of the observability grammian implies $(\boldsymbol{A}, \boldsymbol{C})$ is unobservable.

A physical interpretation of the grammians, based on energy considerations, is presented in *[Glover, 1984]*. In order to assure that the energy quantities are finite, it is assumed that the signals $u(t)$ and $y(t)$ are square–integrable. Thus, according to *[Glover, 1984]*, the controllability grammian $\boldsymbol{P}$ represents a measure of the energy necessary to transport a system from a state $\underline{x}_0$ to $\underline{x}_1$:

$$\min\left\{\int_{-\infty}^{0} \underline{u}^T(t)\underline{u}(t)dt\right\} = (\underline{x}_1 - \underline{x}_0)^T \boldsymbol{P}^{-1}(\underline{x}_1 - \underline{x}_0) \tag{3.9}$$

An eigenvalue of $\boldsymbol{P}$ is an expression of the controllability of the system in the direction of the corresponding eigenvector. A large eigenvalue of $\boldsymbol{P}$

indicates that in the direction of the corresponding eigenvector, little energy is needed to transport the system in that direction. Hence, in the case when $\boldsymbol{P}^{-1}$ has some large eigenvalues[1] (i.e. $\boldsymbol{P}$ has small eigenvalues), some states will only be reached if a large input energy is used.

On the another hand, the observability grammian $\boldsymbol{M}$ is related to the energy transmitted to the output, through the time interval $[\,0\;\infty[$ (when $u(t) = 0$, $t > 0$):

$$\int_0^\infty \underline{y}^T(t)\underline{y}(t)dt = \underline{x}_0^T \boldsymbol{M}\underline{x}_0 \tag{3.10}$$

where $\underline{x}_0$ is the initial state. An eigenvalue of $\boldsymbol{M}$ is an expression of the observability of the system in the direction of the corresponding eigenvector. A large eigenvalue of $\boldsymbol{M}$ suggests that the corresponding state is strongly observable. Otherwise, a small eigenvalue of $\boldsymbol{M}$ implies that some initial conditions will have little effect on the output and consequently they will be difficult to observe.

Consider now two realizations of the same transfer function $G(s)$:

$$(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n \text{ and } (\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \tilde{\boldsymbol{D}})_n$$

Between these two realizations, a singular transformation exists, defined by a nonsingular matrix $\boldsymbol{T}$ (see *[Kailath, 1980]*) such that $\underline{z} = \boldsymbol{T}\underline{x}$ and:

$$\tilde{\boldsymbol{A}} = \boldsymbol{TAT}^{-1}, \quad \tilde{\boldsymbol{B}} = \boldsymbol{TB}, \quad \tilde{\boldsymbol{C}} = \boldsymbol{CT}^{-1}, \quad \tilde{\boldsymbol{D}} = \boldsymbol{D} \tag{3.11}$$

[1]The eigenvectors of the $\boldsymbol{P}$ and $\boldsymbol{P}^{-1}$ are the same, while their eigenvalues are reciprocal.

The controllability and observability matrices are related by:

$$\tilde{\boldsymbol{W}}_c = \boldsymbol{T}\,\boldsymbol{W}_c, \quad \tilde{\boldsymbol{W}}_o = \boldsymbol{W}_o\,\boldsymbol{T}^{-1} \tag{3.12}$$

The similar transformation $\underline{z} = \boldsymbol{T}\underline{x}$ has no qualitatively influence on the controllability and observability properties of the system, since the eigenvalues remain unchanged. But quantitatively, the properties are modified due to the transformation of the singular values. Using the transformation $\boldsymbol{T}$ the quantitative properties of the system (3.1) can actually be changed (improved or unimproved). Thus, the conditional number and the determinant of the controllability and observability matrices are changed (improved or not).

An interesting fact is that, even though the grammians matrices $\boldsymbol{P}$ and $\boldsymbol{M}$ depend strongly on the state space coordinates, the eigenvalues of the product $(\boldsymbol{PM})$ are invariant with respect to basis changes, since:

$$\tilde{\boldsymbol{P}} = \boldsymbol{TPT}^T, \quad \tilde{\boldsymbol{M}} = \boldsymbol{T}^{-T}\boldsymbol{MT}^{-1}, \quad \tilde{\boldsymbol{P}}\tilde{\boldsymbol{M}} = \boldsymbol{T}(\boldsymbol{PM})\boldsymbol{T}^{-1} \tag{3.13}$$

Thus:

$$\lambda_i(\boldsymbol{PM}) = \lambda_i(\tilde{\boldsymbol{P}}\tilde{\boldsymbol{M}}) \tag{3.14}$$

the eigenvalues of the product of the grammians do not depend on the realization.

3.2.3 Hankel singular values and Hankel matrix

The Hankel singular values of a transfer function $G(s)$ are defined as the square roots of the eigenvalues λ_i of the product of controllability and observability grammians $(\boldsymbol{PM})$:

$$\sigma_i\{G(s)\} \triangleq \sqrt{\lambda_i(\boldsymbol{PM})}, \quad i = 1, \ldots, n \tag{3.15}$$

According to Section 3.2.2, for a stable minimal representation $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$, $\boldsymbol{P}$ and $\boldsymbol{M}$ are positive definite matrices, and therefore the eigenvalues of $(\boldsymbol{PM})$ are strictly positive, i.e. $\sigma_i > 0$. They are usually ordered as $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$. The name arises from the fact that, the Hankel singular values of a transfer function are equal to the singular values of the corresponding Hankel matrix. This aspect will be shown later. Due to definition (3.15), it is observed that the Hankel singular values give a measure of the importance of the various states in the overall system transfer function.

Consider now $\{h_i, i = 0, \cdots, \infty\}$ to be the Markov parameters of the transfer function:

$$G(s) = \sum_{i=0}^{\infty} h_i s^{-i}$$

where:

$$\begin{aligned} h_i &= \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}, \quad i > 0 \\ h_0 &= 0, \qquad\qquad\quad i = 0 \end{aligned} \tag{3.16}$$

The Markov parameters[1] represent weights for the basic functions and they are invariant under similar transformations *[Kailath, 1980]*.

Using the Markov parameters, the infinite Hankel matrix can be defined as follows:

$$\mathcal{H} \triangleq \begin{bmatrix} h_1 & h_2 & h_3 & \cdots \\ h_2 & h_3 & h_4 & \cdots \\ h_3 & h_4 & h_5 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.17}$$

It is observed that the Hankel matrix consists of elements, which describe the relationship between input and output. This matrix plays an important role in the model reduction problem. Based on (3.16), the Hankel matrix (3.17) can be rewritten as:

$$\mathcal{H} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \vdots \\ \boldsymbol{CA}^{n-1} \\ \vdots \end{bmatrix} \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \cdots & \boldsymbol{A}^{n-1}\boldsymbol{B} & \cdots \end{bmatrix} = \mathcal{W}_o \mathcal{W}_c \tag{3.18}$$

[1]For MIMO case, the Markov parameters are not anymore scalars, but matrices.

The dimension of the matrix $\mathcal{H}$ is infinite *[Glover, 1984]*, while its rank is finite and equal to the order of the transfer function without common factors:

$$\operatorname{rank}(\mathcal{H}) \leq n \tag{3.19}$$

The rank of the Hankel matrix is thus connected with the concept of the minimal realization. When the representation $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$ is minimal, the rank of Hankel matrix is n *[Kailath, 1980]*.

From (3.12) and (3.18), it can be observed that the Hankel matrix is not influenced by a similar transformation.

In the following analysis, the finite Hankel matrix with dimension $(\tau \times \tau)$, where $\tau > n$, will be considered:

$$\boldsymbol{H} = \boldsymbol{W}_o \boldsymbol{W}_c$$

According to *[Glover, 1984]*, the singular values of $\boldsymbol{H}$ can be evaluated as:

$$\begin{aligned} \sigma_i^2 &= \lambda_i(\boldsymbol{H}^T \boldsymbol{H}) = \lambda_i\{\boldsymbol{W}_c^T \boldsymbol{W}_o^T \boldsymbol{W}_o \boldsymbol{W}_c\} \\ &= \lambda_i\{\boldsymbol{W}_c \boldsymbol{W}_c^T \boldsymbol{W}_o^T \boldsymbol{W}_o\} = \lambda_i\{\boldsymbol{PM}\} \end{aligned} \tag{3.20}$$

This means that the singular values of Hankel matrix can be directly determined as the eigenvalues of the matrix $(\boldsymbol{PM})$, and that they are the same with the Hankel singular values defined in (3.15).

Properties of the Hankel singular values

The Hankel singular values σ_i can be used to get some ideas about the dynamic system, as follows:

- The sum of the Hankel singular values gives half of the absolute value of DC – gain of the system:

$$\sum_{i=1}^{n} \sigma_i = \frac{|DC|}{2} \tag{3.21}$$

  and it can be understood as the total energy of the system *[Fernando and Nicholson, 1983]*.

- The sum of the squares of Hankel singular values is equal to the variance (AC power) of the output of the system with the impulse response transfer function $G(j\omega)$, driven by a sequence of white noise with mean zero and variance 1 *[Moore, 1981]*:

$$\sum_{i=1}^{n} \sigma_i^2 = \int_{-\infty}^{\infty} |G(j\omega)|^2 d\omega$$

Most theory of reduced order approximations has been developed for the continuous time case, but the use of a zero order hold, see *[Glover, 1984]*, often gives a useful link to discrete time. It can be choose to do the model reduction in discrete time or in continuous time. Note that it is not get the same reduced order approximations. However, some interesting aspects about the connection between continuous and discrete time domain, are pointed out in *[Glover, 1984]*. Thus, if $F(z)$ is the representation of the transfer function $G(s)$ in the z – domain (assuming a zero order hold and a sampling period T_s), then:

- the grammians of the transformed discrete time system equal the grammians of the continuous time system *[Wahlberg, 1987]*.

- the Hankel singular values of the continuous time $G(s)$ and of the discrete $F(z)$ systems are identical.

- minimal degrees of F and G are the same.

- suppose that $\tilde{G}_k(s)$ represents a k order approximation of the continuous transfer function $G(s)$. Then, if $\tilde{F}_k(z)$ is the discrete transfer function of the reduced k order transfer function $\tilde{G}_k(s)$, the following relation will be satisfied:

$$\sigma_i\{F(z) - \tilde{F}_k(z)\} = \sigma_i\{G(s) - \tilde{G}_k(s)\}, \quad i = 1, \ldots, n \tag{3.22}$$

  Namely, a solution to the equivalent continuous time problem, generates a solution to the discrete time problem in the sense that, the Hankel singular values of the errors will be the same. This means that the approximation error of the reduced model is the same both for the continuous and discrete situations.

3.2.4 Error norms

When a stable high order dynamic system with transfer function $G(s)$ must be approximated by a low order transfer function $\tilde{G}_k(s)$, inevitably there will be a difference between the two descriptions. Let the error transfer function, denoted by:

$$E(s) = G(s) - \tilde{G}_k(s)$$

$E(s)$ has the inverse Laplace transform $e(t)$.

Some of the most usual error norms used in the control theory are:

- L_2 norm:

$$||e(t)||_2^2 = \int_0^\infty e(t)^T e(t) dt$$

- H_∞ norm:

$$||E(s)||_\infty \triangleq \sup_\omega \overline{\sigma}\{E(j\omega)\}$$

  where $\overline{\sigma}$ denotes the largest Hankel singular value.

- Hankel norm:

$$||E(s)||_H \triangleq \overline{\sigma}\{E(s)\}$$

  where $\overline{\sigma}$ denotes the largest Hankel singular value.

The smallest error[1] is achieved by the Hankel norm (see *[Glover, 1987]*):

$$||E(s)||_H \leq ||E(s)||_\infty$$

[1] It can be shown *[Glover, 1987]*, that $||E(s)||_H$ and $||E(s)||_\infty$ are invariant under a rescaling of the frequency axis ($\omega \rightarrow \alpha\omega$), except $||e(t)||^2$ which will be scaled by $\alpha^{-1/2}$.

3.3 Reduction strategies

The considered model approximation problem is oriented towards the application rather than the description, namely only the input-output behavior is relevant and not the internal structure or physical interpretation of the states. The states from the reduced model do not have any kind of physical interpretation. Thus, the problem may be formulated as follows: the aim of the model reduction is, for a given set of inputs, that the original system and the reduced system produce as similar outputs as possible.

A review of two reduction techniques, Balanced Truncation Approximation (BTA) and Hankel Norm Approximation (HNA), will be given. The reader is referred to *[Moore, 1981]* or *[Pernebo and Silvermann, 1979]* for full details on the BTA, and to *[Glover, 1984]* for a complete study of the HNA. An extension of BTA with frequency weighting can be found in *[Enns, 1984b]*.

Both approaches, BTA and HNA, based on the singular value decomposition, have as a starting point, the balanced realization. This realization is characterized by certain symmetries between the controllability (input property) and observability (output property). The reduced order model is then obtained by discarding those states having little effect on the input/output behavior of a system.

It should be pointed out that the balance realization is acquired for a stable minimal representation. There will always exist a nonsingular balancing state transformation $\boldsymbol{T} \in R^{n \times n}$, which takes a given minimal realization $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$ onto a balanced realization $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \tilde{\boldsymbol{D}})_n$, as defined in (3.11).

3.3.1 Balanced realizations

A balanced realization $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \tilde{\boldsymbol{D}})_n$ is defined as a realization where the controllability $\tilde{\boldsymbol{P}}$ and observability $\tilde{\boldsymbol{M}}$ grammians are diagonal matrices and equal to the matrix $\boldsymbol{\Sigma}$:

$$\tilde{\boldsymbol{P}} = \tilde{\boldsymbol{M}} = \boldsymbol{\Sigma} \tag{3.23}$$

where $\boldsymbol{\Sigma}$ is defined as:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & & & 0 \\ & \sigma_2 & & \\ & & \ddots & \\ 0 & & & \sigma_n \end{bmatrix} \tag{3.24}$$

According to (3.14) and (3.15), the diagonal elements of matrix $\boldsymbol{\Sigma}$ represent the Hankel singular values σ_i, ordered as $\sigma_i \geq \sigma_{i+1} > 0$ $(i = 1, \cdots, n)$.

The definition (3.23) implies the fact that the balanced realization is characterized by equal weights for controllability and observability properties.

The physical interpretation of the grammians presented on page 61, is quite evident in the case of balanced realization. For showing that, without loss of generality :

$$\underline{x}_i = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}$$

can be considered as an initial state. As the grammians, in the balance realization, are equal and diagonal then, according to (3.9) and (3.10), the input energy and the output energy from the state $\underline{x}_i$ are respectively:

$$\int_{-\infty}^{0} \underline{u}^T(t)\underline{u}(t)dt = \frac{1}{\sigma_i}, \qquad \int_{0}^{\infty} \underline{y}^T(t)\underline{y}(t)dt = \sigma_i \tag{3.25}$$

Thus, if σ_i is small, then a relatively large amount of input energy is necessary to drive the system from the state $\underline{x}_i$. Further a little amount of energy is transmitted to the output. Therefore, it can be said that, if σ_i is small then the states in the $\underline{x}_i$ direction are very weakly coupled to the input (hard to control) and to the output (hard to observe). This means that, in the balanced realization, the states which require a high input energy to control are also the states which are difficult to observe and vice-versa.

Further, as long as $\boldsymbol{\Sigma}$ has distinct eigenvalues σ_i, the balance realization is essentially unique *[Moore, 1981]*. This affirmation is based on the following fact: assume that the states $\underline{x}_b$ correspond to a balance representation, characterized by the grammians equal with $\boldsymbol{\Sigma}$. If it is trying to find another nonsingular transformation $\boldsymbol{T}$, thus $\underline{z} = \boldsymbol{T}\underline{x}_b$ is still a balance realization with the same $\boldsymbol{\Sigma}$ as grammians, then the following relation exists:

$$\boldsymbol{T}\boldsymbol{\Sigma}\boldsymbol{T}^T = \boldsymbol{T}^{-T}\boldsymbol{\Sigma}\boldsymbol{-1} = \boldsymbol{\Sigma}$$

which further means that:

$$\boldsymbol{T}\boldsymbol{\Sigma}^2\boldsymbol{T}^{-1} = \boldsymbol{\Sigma}^2$$

If σ_i are distinct, then $\boldsymbol{T}$ must be a diagonal matrix with ± 1 on the diagonal, fact which implies that $\underline{z} = \pm \underline{x}_b$. Therefore it is said that the balanced realization is unique within a change of sign. The sign of the balanced states does not influence the singular values of the system.

Balanced realization algorithm

There are several algorithms for computing the balanced realization. The objective of any balancing algorithm is to find a minimal realization $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \boldsymbol{D})_n$ of the minimal realization $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$ such that the controllability and observability grammians for $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \tilde{\boldsymbol{D}})_n$ are equal and diagonal. In the following, the algorithm[1] suggested by *[Glover, 1984]* is considered:

1. Determine the controllability $\boldsymbol{P}$ and observability $\boldsymbol{M}$ grammians, by solving the Lyapunov equations (3.8).

2. Perform a Cholesky factorization for the controllability grammian, namely write $\boldsymbol{P}$ as follows:

$$\boldsymbol{P} = \boldsymbol{R}^T \boldsymbol{R}$$

3. Form the positive definite matrix: $\boldsymbol{S} \triangleq \boldsymbol{R}\, \boldsymbol{M}\, \boldsymbol{R}^T$.

4. Perform the singular value decomposition: $\boldsymbol{S} = \boldsymbol{U}\, \boldsymbol{\Sigma}^2 \boldsymbol{U}^T$.

5. Calculate the transformation matrix:

$$\boldsymbol{T} = \boldsymbol{\Sigma}^{-1/2} \boldsymbol{U}^T \boldsymbol{R}$$

[1] A dual procedure based on the observability grammian can be done.

6. The balanced representation $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \boldsymbol{D})_n$ is then obtained by using expressions (3.11).

The algorithm is based on the crucial assumption that the system must be asymptotically stable, otherwise the Cholesky factorization is not solvable and the Lyapunov equation does not have a unique solution. A slight problem with the above algorithm is that it can be numerical ill-conditioned if there exist small singular values (for more details see *[Wahlberg, 1987]*).

It must be stressed that the algorithm starts from a minimal realization $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})_n$. This start condition can be neglected in the case when the algorithm is slightly modified to check the sign of σ_i and to remove the possible negative values of σ_i during the procedure *[Enns, 1984a]*.

3.3.2 Balanced Truncation Approximation (BTA)

Since its introduction by *[Moore, 1981]*, BTA has become a popular and successful method for system approximation.

The main idea of BTA is that the reduced order model is obtained from the balanced realization by neglecting the weakly controllable/observable states of the balanced realization. The singular values of the balanced grammian, ordered as:

$$\sigma_1 \geq \ldots \geq \sigma_k \gg \sigma_{k+1} \geq \ldots \sigma_n > 0$$

provide a measure for determining how controllable/observable a given state direction is. Thus, each σ_i is associated with a particular state x_i of the balanced state space system, and the relative size of σ_i indicates to what extent x_i contributes to the input/output behavior of the system.

For model reduction to be meaningful it is assumed that a k exists such that $\sigma_k \gg \sigma_{k+1}$. This means that the state x_{k+1} will affect the input/output behavior much less than the state x_k. The choice of order k strongly depends on the application and is based on a set of tests, which will be described later. As a rule of thumb k can be selected as that index, where the singular value σ_{k+1} has a significant drop compared with σ_k.

Taking $\sigma_k \gg \sigma_{k+1}$ into account, matrix $\mathbf{\Sigma}$, defined in (3.24), can be divided in two diagonal blocks $\mathbf{\Sigma_1}$ and $\mathbf{\Sigma_2}$:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma_1} & \vdots & 0 \\ \cdots & \vdots & \cdots \\ 0 & \vdots & \mathbf{\Sigma_2} \end{bmatrix} \tag{3.26}$$

where $\mathbf{\Sigma_1}$ contains the largest k Hankel singular values and therefore the most significant modes of the system, while $\mathbf{\Sigma_2}$ contains the most insignificant Hankel singular values. The states corresponding to $\mathbf{\Sigma_1}$ can be interpreted as the very controllable and observable states. The states corresponding to $\mathbf{\Sigma_2}$ are nearly uncontrollable and nearly unobservable, and thus insignificant.

Based on (3.26), the states in the balanced representation $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \boldsymbol{D})_n$ can also be disconnected as follows:

$$\begin{aligned} \begin{bmatrix} \dot{\underline{x}}_1 \\ \cdots \\ \dot{\underline{x}}_2 \end{bmatrix} &= \begin{bmatrix} \boldsymbol{A}_r^b & \vdots & \boldsymbol{A}_{12} \\ \cdots & \cdots & \cdots \\ \boldsymbol{A}_{21} & \vdots & \boldsymbol{A}_{22} \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_r^b \\ \cdots \\ \boldsymbol{B}_2 \end{bmatrix} \underline{u} \\ \underline{y}(t) &= \begin{bmatrix} \boldsymbol{C}_r^b & \vdots & \boldsymbol{C}_2 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \cdots \\ \underline{x}_2 \end{bmatrix} + \boldsymbol{D}\,\underline{u} \end{aligned} \tag{3.27}$$

Thus the balanced state space system (3.27) can be regarded as two interconnected subsystems $(\boldsymbol{A}_r^b, \boldsymbol{B}_r^b, \boldsymbol{C}_r^b, \boldsymbol{D})_k$ and $(\boldsymbol{A}_{22}, \boldsymbol{B}_2, \boldsymbol{C}_2)_{n-k}$. The subsystem $(\boldsymbol{A}_r^b, \boldsymbol{B}_r^b, \boldsymbol{C}_r^b, \boldsymbol{D})_k$ is considered to be an approximation of $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}, \boldsymbol{D})_n$:

$$G(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D} \approx \boldsymbol{C}_r^b(s\boldsymbol{I} - \boldsymbol{A}_r^b)^{-1}\boldsymbol{B}_r^b + \boldsymbol{D} \tag{3.28}$$

The reduced order model is thus obtained by keeping only the states corresponding to $\mathbf{\Sigma}_1$.

Following the work of *[Pernebo and Silvermann, 1979]*, if the matrix $\mathbf{\Sigma}$ in (3.26) has distinct diagonal elements ($\sigma_i \neq \sigma_{i+1}$) then every subsystem of the balanced representation is asymptotically stable. Further, as the reduced order model retains the completely controllable and observable states, then it means that $(\boldsymbol{A}_r^b, \boldsymbol{B}_r^b, \boldsymbol{C}_r^b, \boldsymbol{D})_k$ is minimal and asymptotically stable.

Other interesting practical remarks can be outlined:

- In general, the poles of the reduced model $(\boldsymbol{A}_r^b, \boldsymbol{B}_r^b, \boldsymbol{C}_r^b, \boldsymbol{D})_k$ are not a subset of the poles of the full system.

- In general, the weak balanced subsystem $(\boldsymbol{A}_{22}, \boldsymbol{B}_2, \boldsymbol{C}_2)_{n-k}$ does not correspond to the error system $G(s) - \tilde{G}_k(s)$.

- It is important to note that the BTA technique requires the high order model to be asymptotically stable. However, in many practical cases, the high order model is unstable. Some work has been done concerning BTA for unstable models *[Glover, 1984]* and *[Enns, 1984b]*. It was outlined that a requirement for the model fidelity is that the reduced order model must have the same number of unstable poles as the high order model. Based on this affirmation, typically, the initial unstable system is decomposed in a stable and an unstable part. The reduced model can be obtained by reducing the order of the stable part only, leaving the unstable part as is. Some open questions concerning the error of the model still exist. However, for the case when the unstable poles are on the imaginary axis, *[Enns, 1984b]* the model error seems to be bounded. Nevertheless the analyse concerning the problem of reduction of an unstable system is beyond the scope of the thesis.

3.3.3 Hankel Norm Approximation (HNA)

The theory of Hankel operators and Hankel norm is based on rather deep functional analysis. The Hankel matrix and the Hankel norm were already presented in Sections 3.2.3 and 3.2.4, respectively. In this section the technical details and the algorithm are avoided (see details in *[Glover, 1984]*). Only the main ideas on the HNA are pointed out.

HNA of $G(s)$, denoted by:

$$\tilde{G}_k(s) = \boldsymbol{C}_r^h(s\boldsymbol{I} - \boldsymbol{A}_r^h)^{-1}\boldsymbol{B}_r^h + \boldsymbol{D}_r^h$$

is chosen such that the Hankel norm of the error system $G(s) - \tilde{G}_k(s)$:

$$||G(s) - \tilde{G}_k(s)||_H$$

is minimized. The choice of the finite $\boldsymbol{D}_r^h$ (see *[Glover, 1984]*) is arbitrary since the Hankel norm does not depend on $\boldsymbol{D}_r^h$.

According to *[Glover, 1984]*, the Hankel norm of the error between a stable rational transfer function $G(s)$ and its reduced order transfer functions $\tilde{G}_k(s)$ satisfies the following inequality:

$$||G(s) - \tilde{G}_k(s)||_H \geq \sigma_{k+1} \tag{3.29}$$

It is said that a k order approximation $\tilde{G}_k(s)$ is optimal if the Hankel norm error is equal to σ_{k+1}.

The first step in finding the HNA via the Glover algorithm, see *[Glover, 1984]* for a comprehensive treatment, is also the balanced realization. However, the computations of this algorithm are more complicated than those from BTA algorithm.

3.4 On the optimality aspect

The problem of approximating a high order linear dynamic system with a relative more simple system is like dealing with the compromise between the minimization of an ideal error norm and a computational burden. It is important to note that both BTA and HNA techniques require $G(s)$ to be asymptotically stable, which is often a restrictive condition.

The theoretical optimum can normally not be determined. Consequently, computable lower bounds on the theoretical optimal error are used in order to check if a method is satisfactory or not.

About the optimality of BTA

This approach is completely independent of optimality considerations and therefore there is no expectation that the reduced order models, obtained via BTA, are in any sense optimal. However, from the computational experience reported by different researches *[Moore, 1981]*, *[Glover, 1984]*, it is concluded that BTA can be near optimal.

BTA has the error bound:

$$\|G(s) - \tilde{G}_k(s)\|_H \leq \|G(s) - \tilde{G}_k(s)\|_\infty \leq 2(\sigma_{k+1} + \cdots \sigma_n) \qquad (3.30)$$

namely an upper bound on the H_∞ norm of the error system, which is twice the sum of the discarded Hankel singular values. This error bound gives strong theoretical support to the observation that the BTA gives good

results in practice. The possible multiple Hankel singular values are considered only once in the inequality (3.30).

An interesting work was done by *[Mustafa and Davidson, 1993]*, where it was concluded that for a certain class of systems, namely the stable systems where $\boldsymbol{A}$ is symmetric and $\boldsymbol{B}\,\boldsymbol{B}^T = \boldsymbol{C}^T\boldsymbol{C} = \boldsymbol{I}_n$, BTA is Hankel norm optimal.

BTA is computationally less expensive than HNA, fact which makes BTA to be very attractive in practice.

About the optimality of HNA

HNA approach is by definition an optimization problem. Therefore it is expected that the error resulting from HNA technique would be less than or equal to that resulting from BTA technique.

HNA is characterized by the lower error bound:

$$\sigma_{k+1} \leq \|G(s) - \tilde{G}_k(s)\|_H \leq \|G(s) - \tilde{G}_k(s)\|_\infty \qquad (3.31)$$

[Glover, 1984] proves that there exists a finite direct term $\boldsymbol{D}_r^h$, such that:

$$\|G(s) - \tilde{G}_k(s)\|_\infty \leq (\sigma_{k+1} + \cdots + \sigma_n) \qquad (3.32)$$

Even though the $\boldsymbol{D}_r^h$ term is independent of Hankel norm minimization, it can be used as a freedom to ensure that the HNA technique yields a solution which has an error bound half that of the BTA technique. However, the computations for the HNA technique are computationally expensive and according to *[Enns, 1984b]*, in the case when the direct term in the reduced model is set to zero, the following bound holds:

$$||G(s) - \tilde{G}_k(s)||_\infty \leq 2(\sigma_{k+1} + \cdots \sigma_n)$$

This aspect means that, BTA can be regarded as an attractive technique for performing model reduction, due to the fact that it is computationally less expensive and satisfies the same error bound as the HNA, for the same restrictions on the reduced order model.

A final note to make here is that a frequency weighting can be incorporated into such model reduction schemes to emphasize frequency ranges over which it is desired to preserve the nominal input/output properties as much as possible during approximation. This aspect was proposed and investigated by *[Enns, 1984b]*.

3.5 Testing the reduced order approximation

In this section, some practical tools for testing the reduced order approximation are suggested.

As it was already pointed out, the choice of the order k of the reduced model is crucial in the model reduction strategy and it is highly dependent upon the practical application. However, the potential reduced model $(\boldsymbol{A}_r, \boldsymbol{B}_r, \boldsymbol{C}_r)_k$ can be judged on the basis of the following reliable open loop guidelines:

- The error impulse response function, defined as:

$$h_e(t) \triangleq \boldsymbol{C}e^{\boldsymbol{A}t}\boldsymbol{B} - \boldsymbol{C}_r e^{\boldsymbol{A}_r t}\boldsymbol{B}_r \tag{3.33}$$

  should be as small as possible. A small model reduction error has a predictably small effect on the closed loop objectives, and hence the reduction procedure and control design are in some sense compatible.

- The relative error in the impulse response:

$$\frac{(\int_0^\infty h_e(t)^2 dt)^{1/2}}{(\int_0^\infty h(t)^2 dt)^{1/2}} \tag{3.34}$$

  should be small.

- The reduced model should follow the high order model without large discrepancies. This can be checked by the inspection of Bode and impulse response diagrams.

- The existence of an approximate subsystem of order k is conditioned by the inequality:

$$\left(\sum_{i=1}^{k} \sigma_i^4\right)^{1/2} \gg \left(\sum_{i=k+1}^{n} \sigma_i^4\right)^{1/2} \tag{3.35}$$

- The DC gain of the reduced order model should be the same with the DC gain of the initial model.

In the case when for a certain order k, these tests give good results, then it means that k is an appropriate order for the reduced model.

3.6 Model reduction and control of a ship's model

In this section an exemplification of the model reduction algorithms on the Mariner Class Vessel model is considered as a study case. The purpose is to verify some statements made in the previous sections and to illustrate graphically the effect of the model reduction. Finally, a simulation of LQG control, designed on the reduced order model, is presented.

3.6.1 High order ship model

For exemplification, a stochastic high order model of the Mariner Class Vessel is examined. The external disturbances which are taken into account in the model are the wind and the waves. The direction of wind and of waves is considered to be 45 deg, while the mean wind is $\overline{V} = 15$ m/s.

The high order model is thus constituted based on the expressions (2.11), (2.25), (2.39) outlined in Chapter 2. The data of the ship are summarized in Appendix A. The nonlinear model is linearized in a certain stationary point $\underline{x}_0$, e.g. $\underline{x}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & U & 0 & 0 & 0 \end{bmatrix}^T$, where U is the nominal speed $U = 7.71$ m/s (15 knots). Then the linear stochastic high order model, of order 8, can be written as:

$$\begin{pmatrix} \dot{x}_1^{wind} \\ \dot{x}_2^{wind} \\ \dot{x}_1^{wave} \\ \dot{x}_2^{wave} \\ \dot{u} \\ \dot{v} \\ \dot{r} \\ \dot{\psi} \end{pmatrix} = \boldsymbol{A}_h \begin{pmatrix} x_1^{wind} \\ x_2^{wind} \\ x_1^{wave} \\ x_2^{wave} \\ u \\ v \\ r \\ \psi \end{pmatrix} + \boldsymbol{B}_h \, \delta + \boldsymbol{W}_h \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} \tag{3.36}$$

$$\psi = \boldsymbol{C}_h \begin{bmatrix} x_1^{wind} & x_2^{wind} & x_1^{wave} & x_2^{wave} & u & v & r & \psi \end{bmatrix}^T$$

where the states are:

x_1^{wind}, x_2^{wind}	states in the wind model (2.25).
x_1^{wave}, x_2^{wave}	states in the wave model (2.39).
u	the surge speed [m/s].
v	the sway velocity [m/s].
r	the yaw rate [deg/s].
ψ	the heading angle [deg].

Inputs to the system are:

δ	the rudder angle [deg].
e_1, e_2	two white noise sources with zero mean and variance σ_e^2.

The matrices $\boldsymbol{A}_h \in R^{8\times8}$, $\boldsymbol{B}_h \in R^{8\times1}$, $\boldsymbol{C}_h \in R^{1\times8}$, $\boldsymbol{W}_h \in R^{8\times2}$ depend on the stationary point $\underline{x}_0$. They are shown in Appendix C.

3.6.2 Model reduction

The model reduction algorithms BTA and HNA will be exemplified on the high order transfer function $G_{\psi\delta}$ from the rudder angle δ to heading angle ψ[1], due to its importance in the design of a course controller. The reduction of the system will be performed in continuous time.

Due to the big conditional number of the grammians of the high order model it is concluded that, the system contains uncontrollable or unobservable states which make the ship's model very difficult to control. Therefore, the solution is to reduce the model as much as possible and to perform the control design on the most reduced model, which is still able to describe the fundamental modes of the high order model.

As it was pointed out, in order to apply BTA and HNA approaches, the system should be minimal, namely completely controllable and observable. Therefore, the first step is to find a minimal representation of the high order model $G_{\psi\delta}$. Furthermore, it is important that the system is asymptotically stable. Therefore the possible unstable states must be temporarily removed and included afterwards. This is the case of the heading angle state ψ, because it introduces an integrator in the system[1].

Thus as the heading angle ψ is temporarily removed, the transfer function $G_{r\delta}$ from rudder δ to yaw rate r can be considered instead of $G_{\psi\delta}$.

[1]The analysis of reductions of the transfer function from e_1 and e_2, respectively, to heading angle is similar and therefore is not exemplified.

[1]The mode which corresponds to the heading angle state is sensitive to the direction of the wind. It can even become unstable *[Hansen et al., 1995]*.

Finally, the high order model, which must be reduced, has order 3 (see Appendix C). Afterwards the heading angle is included in the reduced form, based on the expression $\dot{\psi} = r$.

It was shown that the controllability and observability grammians can be used to define measures of controllability and observability in certain direction of the state space. Thus, as the conditional number of the grammians of the 3 order model is still big (see Appendix C), it means that some "weak" controllable and observable states exist. In order to detect the degree of simultaneous controllability and observability, the balance realization is computed (see Appendix C). The Hankel singular values and the reduced order models (of order $k = 1; 2$), corresponding to BTA and HNA approaches, respectively, are summarized in Table 3.1. The fact that the Hankel singular values are distinct, implies that the balanced realization is unique. As it was expected (from page 67), the sum of the Hankel singular values $\sum_{i=1}^{n} \sigma_i = 0.09060$ is approximatively half size of the absolute value of the DC gain -0.18321 of the high order model. Moreover, it can be observed that, the DC gain of the reduced models is almost the same with the DC gain of the initial model. Based on the Hankel singular values, presented in Table 3.1, it can be noticed that the first Hankel singular value is more than ten times bigger than the second one. This fact suggests that a reduced first order model, obtained e.g. BTA or HNA, retains the most important input-output characteristics of the original system $G_{r\delta}$. The time constants for the first order reduced models obtained using BTA and HNA, are slightly different, while their gains are very similar.

<table>
<tr><td colspan="3">Minimal high order transfer function
$G_{r\delta}(s) = \frac{-0.00406s^2 - 0.00027s - 0.0000024}{s^3 + 0.15209s^2 + 0.00275s + 0.0000131}$
DC gain: -0.18321</td></tr>
<tr><td colspan="3">Hankel singular values of the balanced realization:
$(\sigma_1 > \sigma_2 > \sigma_3)$
$\sigma_1 = 0.08358\,;\ \sigma_2 = 0.00689\,;\ \sigma_3 = 0.00013$
$\sum_{i=1}^{i=n} \sigma_i = 0.09060$</td></tr>
<tr><td>order</td><td>BTA</td><td>HNA</td></tr>
<tr><td>$k = 1$</td><td>$\tilde{G}_{r\delta}(s) = \frac{-0.16716}{72.4993s+1}$
DC gain: -0.16716</td><td>$\tilde{G}_{r\delta}(s) = \frac{-0.16782}{80.1352s+1}$
DC gain: -0.16782</td></tr>
<tr><td>$k = 2$</td><td>$\tilde{G}_{r\delta}(s) = \frac{-0.00406s-0.00023}{s^2+0.14084s+0.00126}$
DC gain: -0.1825</td><td>$\tilde{G}_{r\delta}(s) = \frac{-0.00601s-0.0008}{s^2+0.42358s+0.00454}$
DC gain: -0.1762</td></tr>
</table>

Table 3.1. *Minimal high order transfer function and the reduced transfer functions.*

It is well known that the first order Nomoto approximation (2.18) is commonly used for analyzing steering and autopilots. Therefore it is interesting to compare the two first order reduced models, obtained via BTA and HNA, with Nomoto approximation which, for the considered ship, is expressed by:

$$\tilde{G}^N_{r\delta}(s) = \frac{-0.1830}{105.6s + 1} \tag{3.37}$$

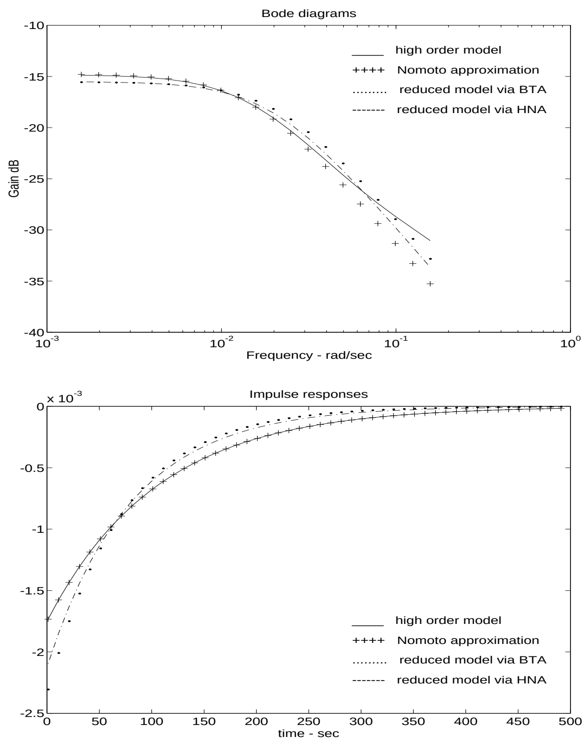

Figure 3.2. *Visualized analysis of the first order reduced models.*

It is observed that the DC gain of the first order reduced models (see Table 3.1) is closed to the DC gain of Nomoto approximation. However, the time constants differ. Figure 3.2 illustrates the Bode and impulse response diagrams for the minimal high order transfer function, for Nomoto approximation and for the two first order models obtained via BTA and HNA. As it was expected, Nomoto approximation fits very well the high order model at low frequencies. However, it is observed that the first order models via BTA and HNA techniques also consist good approximations of the high order transfer function at low frequencies ($\omega \leq 0.01$ rad/s). For low frequencies, the difference between BTA and HNA reduced models is insignificant.

Test	k	**BTA**	**HNA**
Square of impulse response error	1	$1.3286 \cdot 10^{-5}$	$1.4878 \cdot 10^{-5}$
	2	$2.9649 \cdot 10^{-10}$	$6.9894 \cdot 10^{-6}$
Relative impulse response error	1	$6.231 \cdot 10^{-2}$	$6.9770 \cdot 10^{-2}$
	2	$1.3905 \cdot 10^{-6}$	$3.2780 \cdot 10^{-2}$
Hankel norm of the error $E(s)$	1	$1.3650 \cdot 10^{-2}$	$9.5400 \cdot 10^{-3}$
	2	$2.2961 \cdot 10^{-4}$	$4.8030 \cdot 10^{-3}$
H_∞ norm of the error $E(s)$	1	$1.4030 \cdot 10^{-2}$	$1.3370 \cdot 10^{-2}$
	2	$2.5067 \cdot 10^{-4}$	$4.8230 \cdot 10^{-3}$
$\dfrac{\sqrt{\sum_{i=k+1}^{n} \sigma_i^4}}{\sqrt{\sum_{i=1}^{k} \sigma_i^4}}$	1	$6.7900 \cdot 10^{-3}$	
	2	$2.2488 \cdot 10^{-6}$	

Table 3.2. *Tests on the reduced models.*

Both first order reduced models (BTA and HNA) seem to be good approximations of the high order model for the control purpose. Therefore, further, the design of the controller will be done on a first order reduced model.

Some tests can be done further in order to compare the results from BTA and HNA techniques. Thus the Hankel norm and H_∞ norm of the error system $E(s) = G_{r\delta}(s) - \tilde{G}_{r\delta}(s)$ can be computed. The tests described in Section 3.5 are performed. The higher order $k = 2$ reduced models are also taken into discussion. The results are summarized in Table 3.2. Looking at the impulse response error and relative impulse response error, a slightly improvement in the performance it is observed when BTA technique is used. It can also easily be noted that, the inequality:

$$\|E(s)\|_H \leq \|E(s)\|_\infty$$

is satisfied for all of the reduced order models and both for BTA and HNA techniques. As it was expected, the second order approximations provides better test results then the first order approximations. However, it can be observed that the accuracy of the first order models is good enough for control purpose.

For the first order model, a simple check on the Hankel and H_∞ norms of the system error can be done, as following:

- In the case when BTA is used, according to:

  $$\|E(s)\|_H \leq \|E(s)\|_\infty \leq 2(\sigma_{k+1} + \cdots \sigma_n)$$

  the BTA guarantees a maximum H_∞ error of $2 \cdot \sum \sigma_i$, where the sum is taken over the removed states. It can be noticed that the above inequalities are satisfied:

  $$1.3650 \cdot 10^{-2} \leq 1.402 \cdot 10^{-2} \leq 2 \cdot 7.02 \cdot 10^{-3} = 1.404 \cdot 10^{-2}$$

- In the case when HNA is used, according to:

$$\sigma_{k+1} \leq \|E(s)\|_H \leq \|E(s)\|_\infty$$

the inequalities:

$$6.89 \cdot 10^{-3} \leq 9.5400 \cdot 10^{-3} \leq 1.3370 \cdot 10^{-2}$$

are satisfied.

It should be stressed that, in this exemplification, the direct term was set to zero. Therefore, conform to *[Enns, 1984b]*, the HNA guarantees a maximum H_∞ error of $2 \cdot \sum \sigma_i$ (not $\sum \sigma_i$), where the sum is taken over the removed states:

$$\|E(s)\|_\infty = 1.337 \cdot 10^{-2} \leq 2(\sigma_{k+1} + \cdots \sigma_n) = 0.01404$$

It is observed that the first order reduced model, obtained via BTA, satisfies the same error bound as the the first order model, obtained via HNA. Therefore, due to the fact that BTA is much less computationally expensive than HNA, BTA seems to be an attractive technique for performing the model reduction in the present case. Thus, the first order model $\tilde{G}_{r\delta}$, via BTA, will be considered as an approximation for the high order model.

Once the reduced model $\tilde{G}_{r\delta}$ is get, the heading angle state should be included in this reduced form, based on the expression:

$$\dot{\psi} = r \tag{3.38}$$

3.6.3 Control based on reduced design model

In this section, it is illustrated how the classical servo-problem applied on a stochastic high order ship model can be handled by a controller designed on a low order model. The theory on the design and the implementation of the controller is beyond the scope of this section, but however, for exemplification, the design of LQG controller it is briefly reviewed in Appendix D. The requirement of the controller is to ensure that, despite of random disturbances, the system output closely tracks the reference signal $\overline{w}$.

The classical autopilot on board of sea-going vessels is a course controller. Therefore, in the design of the controller, the transfer function $G_{\psi\delta}$ from rudder to heading angle must be available.

According to (3.38), the reduced approximation transfer function from rudder angle to heading angle is obtained as:

$$\tilde{G}_{\psi\delta}(s) = \frac{1}{s}\,\tilde{G}_{r\delta}(s)$$

where $\tilde{G}_{r\delta}$ represents the first order reduced model, obtained via BTA (see Table 3.1). The reduction of the system was performed in continuous time, and therefore in order to design a discrete controller , the reduced model should be discretized. According to Shannon's theorem *[Poulsen, 1995]*, a sampling period $T_s = 12$ seconds can be used.

Thus, as a result of the model reduction and of using the representational theorem *[Poulsen, 1995]*, the proper form for the design of the controller can finally be expressed as:

$$A(q^{-1})\psi(t) = q^{-d}\, B(q^{-1})\delta(t) + C(q^{-1})e(t) + l \tag{3.39}$$

where $\psi(t)$ is the heading angle, $\delta(t)$ is the rudder angle, $e(t)$ is a zero mean white Gaussian noise sequence with constant variance, while l can be regarded as mean of the colored disturbances or as level due to the operating point. $A(q^{-1})$, $B(q^{-1})$, $C(q^{-1})$ are polynomials in backward shift operator.

The possible steady state error, due to the presence of l, can be eliminated by considering a LQG controller with incremental action (see Appendix D):

$$R(q^{-1})\,\Delta\delta(t) = V(q^{-1})\,w(t) - S(q^{-1})\,\psi(t) \tag{3.40}$$

A continuous simulation of the stochastic high order system, controlled by the reduced order LQG controller, is illustrated in Figure 3.3. The graphics of the heading angle ψ, of the yaw rate r, of the rudder angle δ and of the rudder rate are presented. In the simulation, two strategies are considered for exemplification: 1) case when the wind and the waves have the same direction (45 deg) and 2) case when the wind and the waves have different directions (wave angle 45 deg and wind angle 135 deg). For each strategy, a new reduced design model is computed.

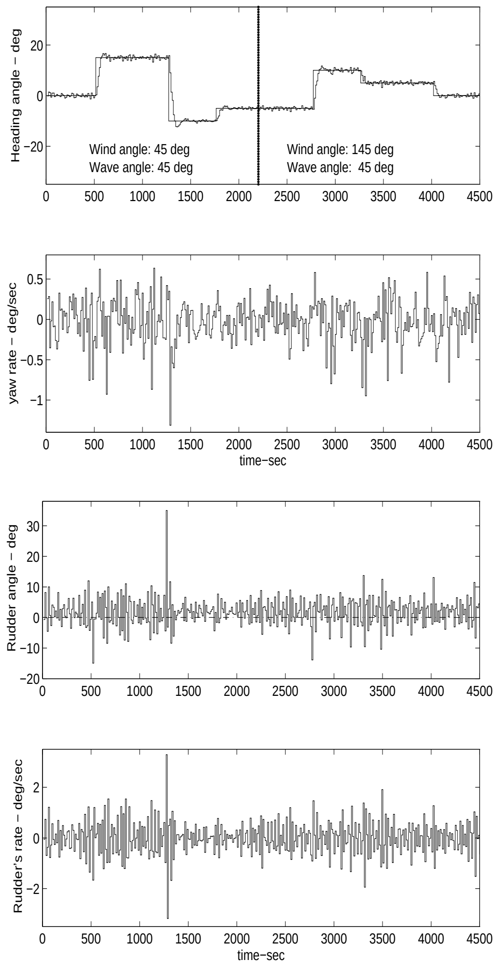

Figure 3.3. *LQG control ($\rho = 0.01$) on the Mariner Class Vessel, for different directions of wind and wave. Mean of the wind is 15 m/s.*

However, the influence of the disturbances on the reduced model seems to be very small (see Appendix D).Different references for the heading angle are imposed in order to check the tracking ability of the reduced controller. It is known that the LQG design technique requires the manipulation of matrices with dimensions greater than or equal to that of the plant model. Nevertheless, the reduced order model used in the design of the controller implies a more simple control law (see Appendix C) and a decreased computational effort. Moreover, it is observed that the reduced controller is numerically more efficient. The simulation reflects that the controller designed on the reduced order model ensures that the system output closely tracks the reference signal.

3.7 Summary

In the present chapter, some aspects related with the model reduction problem were presented. The problem of approximating a high order linear dynamic system with a relatively more simple system is dealing with the compromise between the minimization of an ideal error norm and a computational burden. There does not exist an universal model reduction algorithm, as the relative importance of various plant characteristics is highly dependent upon the application.

Two model reduction methods, Balanced Truncation Approximation (BTA) and Hankel Norm Approximation (HNA), were investigated and found to have desirable properties as model reduction tools. Both techniques use the singular values of the system to find a reduced order model. The size of

the singular values determine how well the lower order model approximates the system. The computations for these methods have as starting point the balanced realization, characterized by states which are as controllable as observable. BTA technique does not attempt to minimize any error criterion. The reduced order model is simply obtained by neglecting the states which are weakly controllable/observable in the balanced realization. HNA technique is based on the minimization of the Hankel norm of the system error, and thus it is considered to provide the optimal reduced order model. However, it is computationally expensive related to BTA and for the same assumptions on the reduced order model (e.g. $\boldsymbol{D}_r = 0$), it satisfies the same error bound as BTA technique.

An exemplification of these two reduction methods was done for a model of the Mariner Class Vessel. The choice of a proper reduced model was illustrated by comparison and verification of a set of tests. Due to its good performance at low frequencies, the Nomoto approximation was also used to judge the first order reduced models. It was pleasant to see that the first order reduced models, obtained via BTA and HNA, were very similar to the empirical approximation given by *[Nomoto, 1957]*. The H_∞ norm and the Hankel norm of the error system were also computed. It was observed that the H_∞ norm satisfied the same error bounds for both reduction methods, BTA and HNA. Finally, due to the fact that BTA involves more simple calculations than HNA, the reduced model obtained by BTA was chosen. Further this reduced model was used further for control purpose. It was shown how the classical servo problem for ships can be handled by a controller designed on a low order model. The example demonstrated the effectiveness of the model reduction technique for obtaining successful control design.

Chapter 4

Unconstrained predictive control

Due to the ability to handle a wide variety of processes, long range predictive controllers have received a lot of attention in recent years. Early algorithms such as MAC (Model Algorithmic Control), DMC (DynamicMatrix Control), presented in *[Richalet et al., 1978]*, and *[Cutler and Ramaker, 1980]*, respectively, were based on deterministic impulse or step response models. Later, controllers based on fewer parameters in the model representation, such as EPSAC (Extended Prediction Self Adaptive Control) *[Keyser and Cauwenberghe, 1985]* and EHAC (Extended Horizon Adaptive

Control) *[Ydstie, 1984]* were developed. A comparative study of these different predictive algorithms is presented in *[Keyser et al., 1988]*. Generalized Predictive Control, introduced by *[Clarke et al., 1987]*, is a recent generalization of the predictive algorithms, as well as being a natural, long-range extension of the well-known Generalized Minimum Variance Controller, described in *[Clarke and Gawthrop, 1979]*. As a unification of these previous multi-step predictive algorithms, a Unified Predictive Controller (UPC) is described in *[Soeterboek, 1991]*. UPC controller is very close to GPC, the single difference being that UPC is based on a more general linear model structure and criterion function. Thus UPC controller can provide extra flexibility compared to the GPC controller.

All these predictive algorithms are similar in that they are based on the receding horizon control approach, which consists of the following steps:

- A prediction of the process output, over a horizon of sampling periods, is made at each present moment. This forecast is made by means of a real time identified model of the process dynamics, being a function of the future control action.

- The optimization of the control cost function, that measures how the prediction response fits the desired characteristics, is done.

- The resulting best candidate is then applied as a control input to the real process, but only at the present moment. At the next sampling instant the whole procedure is repeated, leading to update the control action with corrections based on the latest measurements.

The reason for using the receding horizon control approach is that this allows to compensate for future disturbances or modelling errors.

Some of the most attractive features of GPC and UPC controller are:

- Robustness against incorrect prior choice of model order and of time delay. It is well-known that MV and GMV controllers have bad performances in the case when the time delay of the model does not coincide with the time delay of the system *[Åström and Wittenmark, 1973]*.
- The concept of predictive control is not restricted to SISO processes *[Kinnaert, 1989]*.
- Ability to handle non-minimum phase and open-loop unstable systems *[Clarke et al., 1987]*.
- Skill to handle process constraints in a systematic way during the design of the controller *[Soeterboek, 1991]*.
- Ability to deal with pre-scheduled reference trajectories.

This chapter is focused on the design algorithm of the predictive controller in the case of a slightly extended model structure and control criterion, than that used in the UPC algorithm *[Soeterboek, 1991]*. The method belongs to the long range predictive control schemes. In order to illustrate the ability of the predictive control to deal with pre-scheduled reference trajectories, a predictive track keeping system for the Mariner Class Vessel is presented in Section 4.4.

Throughout this chapter it is assumed that the system description is known and that no hard constraints are imposed. The presented aspects are known

in the technical literature, and they are included here because they form the basic structure of the optimal constrained predictive control, which will be discussed and illustrated in the next chapter.

4.1 The model and the cost function

Predictive control belongs to the class of model-based controller design concepts, i.e. a model of the process must be available to design the controller. Consider the general linear model structure:

$$A(q^{-1})y(t) = q^{-d}\frac{B(q^{-1})}{A_1(q^{-1})}u(t) + \frac{C(q^{-1})}{A_2(q^{-1})}e(t) + l \tag{4.1}$$

where q^{-1} is the backward shift operator, $y(t)$ denotes the system output, $u(t)$ is the control signal, $e(t)$ is a sequence of zero mean white noise with variance σ_e^2, while l is related to the mean of the disturbances or as level due to the operating point. d denotes the time delay of the process. There is no direct link from $u(t)$ to $y(t)$, i.e. $d \geq 1$. The common poles in the transfer function from the control signal and noise to the system output are gathered in the polynomial $A(q^{-1})$, while the polynomials $A_1(q^{-1})$ and $A_2(q^{-1})$ are coprime and contain only those poles which are specific for the two transfer functions. Without loss of generality, it can be considered that the polynomials $A(q^{-1})$, $A_1(q^{-1})$, $A_2(q^{-1})$ and $C(q^{-1})$ are monic. It is also assumed that $C(q^{-1})$ has all its roots inside the unit circle or on the unit circle. This is justified by the spectral factorization theorem (see e.g. *[Poulsen, 1995]*).

Note that model (4.1) is very general. Different well-known model structures can be obtained by choosing specific forms of the polynomials: e.g. $C(q^{-1}) = 1$, $A_1(q^{-1}) = 1$, $A_2(q^{-1}) = 1 - q^{-1}$ correspond to the ARIMAX model structure used in the GPC controller, while $A(q^{-1}) = 1$ corresponds to the Box-Jenkins model structure, used in UPC controller.

In order to be able to handle different control situations, the following general cost function is taken into account[1]:

$$J = E\left\{ \sum_{i=H_m}^{H_p} [\frac{P_n}{P_d}\hat{y}(t+i) - w(t+i)]^2 + \rho \sum_{i=0}^{H_p-d} [\frac{Q_n}{Q_d}u(t+i)]^2 \right\} \quad (4.2)$$

The criterion is minimized under the following equality constraint:

$$\phi \, \frac{P_n}{P_d} \, u(t+i) = 0, \quad H_c \leq i \leq H_p - d \quad (4.3)$$

The parameters used in (4.2) and (4.3) are described in Table 4.1.

The interpretation of constraint (4.3) is related to the stabilization of non-minimum phase systems. It implies that the choice $\rho = 0$ works even if the system is non-minimum phase. This is unlike GMV controller which requires a positive control weight in connection with non-minimum phase systems. Moreover, the constraint reduces the dimension of the optimization problem. As it will be shown, this constraint is incorporated in the optimization problem, and therefore it is also known as a soft constraint. In the next

[1]For simplify the notation, the dependence of the polynomials A, A_1, A_2, B, C, P_n, P_d, Q_n, Q_d, ϕ on q^{-1} will not be explicitly mentioned. E denotes expectation operator.

	Description	**Type**
H_p	prediction horizon	integer
H_m	minimum cost horizon	integer
H_c	control horizon	integer
$\frac{P_n}{P_d}$	filter on tracking	polynomial fraction
Φ	weighting polynomial	polynomial
ρ	positive weighting factor	real
$\frac{Q_n}{Q_d}$	filter on the controller output	polynomial fraction

Table 4.1. *Parameters used in the predictive control.*

chapter, besides soft constraints, hard constraints will be explained and taken into account.

In the cost function (4.2), the output $y(t)$ and the input $u(t)$ signals are filtered. These filtered signals will be denoted by: $\hat{y}_f(t) = \frac{P_n}{P_d}\hat{y}(t)$ and $u_f(t) = \frac{Q_n}{Q_d}u(t)$. They can be used to minimize J in certain areas of the signals. P_n, P_d and Q_n, Q_d are coprime, respectively. The DC - gain of P_n/P_d should be 0 dB in order to avoid offsets from the reference.

4.2 Derivation of the control law

Before the derivation of the control law, a set of signal vectors and their dimensions, used in the minimization of the cost function (4.2), are summarized, as follows:

$$
\begin{array}{lcll}
\underline{u} & = & \left[u(t) \;\; \ldots \;\; u(t+H_p-d) \right]^T, & [\underline{u}] = (H_p-d+1)\times 1 \\
\underline{u}_f & = & \left[u_f(t) \;\; \ldots \;\; u_f(t+H_p-d) \right]^T, & [\underline{u}_f] = (H_p-d+1)\times 1 \\
\underline{u}_1 & = & \left[u(t) \;\; \ldots \;\; u(t+H_c-1) \right]^T, & [\underline{u}_1] = H_c\times 1 \\
\underline{u}_{1f} & = & \left[u_f(t) \;\; \ldots \;\; u_f(t+H_c-1) \right]^T, & [\underline{u}_{1f}] = H_c\times 1 \\
\underline{u}_2 & = & \left[u(t+H_c) \;\; \ldots \;\; u(t+H_p-d) \right]^T, & [\underline{u}_2] = (H_p-H_c-d+1)\times 1 \\
\underline{u}_{2f} & = & \left[u_f(t+H_c) \;\; \ldots \;\; u_f(t+H_p-d) \right]^T, & [\underline{u}_{2f}] = (H_p-H_c-d+1)\times 1 \\
\underline{w} & = & \left[w(t+H_m) \;\; \ldots \;\; w(t+H_p) \right]^T, & [\underline{w}] = (H_p-H_m+1)\times 1 \\
\underline{\hat{y}}_f & = & \left[\hat{y}_f(t+H_m) \;\; \ldots \;\; \hat{y}_f(t+H_p) \right]^T, & [\underline{\hat{y}}_f] = (H_p-H_m+1)\times 1
\end{array}
$$

Note that $\underline{u}_1$ contains those elements of the controller output sequence that must be calculated, while the elements in $\underline{u}_2$ must satisfy soft constraint in (4.3).

As the considered model (4.1) is linear, the cost function (4.2) is quadratic and no hard constraints are imposed, the cost function (4.2) can be minimized analytically. The optimal control law is derived by minimization of criterion function (4.2) with respect to $\underline{u}_1$:

$$\underline{g} = \frac{\partial J}{\partial \underline{u}_1} = \underline{0} \tag{4.4}$$

where $\underline{g}$ denotes the gradient and the symbol ∂ denotes the partial

derivative. It is well-known that if the Hessian:

$$\boldsymbol{H} = \frac{\partial^2 J}{\partial \underline{u}_1^2}$$

is positive definite and the cost function J is a quadratic function then, a global minimum exist *[Luenberger, 1984]*.

In order to calculate the gradient with respect to $\underline{u}_1$, it is useful to write the criterion function (4.2) in the following matrix compact form:

$$J = (\underline{\hat{y}}_f - \underline{w})^T (\underline{\hat{y}}_f - \underline{w}) + \rho\, \underline{u}_f^T\, \underline{u}_f \tag{4.5}$$

Thus, the gradient of (4.5) with respect to $\underline{u}_1$ becomes:

$$\frac{\partial J}{\partial \underline{u}_1} = 2\, \frac{\partial \underline{\hat{y}}_f}{\partial \underline{u}_1} (\underline{\hat{y}}_f - \underline{w}) + 2\, \rho\, \frac{\partial \underline{u}_f}{\partial \underline{u}_1}\, \underline{u}_f \tag{4.6}$$

Expression (4.6) shows that, in order to compute the gradient, the prediction of the filtered output $\underline{\hat{y}}_f$ and the partial derivatives $\frac{\partial \underline{\hat{y}}_f}{\partial \underline{u}_1}$, $\frac{\partial \underline{u}_f}{\partial \underline{u}_1}$ are required. They are sketched in the following.

Prediction

Applying the filter P_n/P_d to the output (4.1) gives:

$$y_f(t+i) = q^{-d} \frac{P_n B}{P_d A A_1} u(t+i) + \frac{P_n C}{P_d A A_2} e(t+i) \tag{4.7}$$

The noise can be split into future and past noise by using the Diophantine equation:

$$\frac{P_n C}{P_d A A_2} = \mathcal{E}_i + q^{-i} \frac{\mathcal{F}_i}{P_d A A_2} \tag{4.8}$$

which provides the polynomials $\mathcal{E}_i$ and $\mathcal{F}_i$. Their orders are $n_{\mathcal{E}_i} = i-1$ and $n_{\mathcal{F}_i} = \max(n_{P_n} + n_C - i, n_{P_d} + n_A + n_{A_2} - 1)$, respectively . It should be noticed that these polynomials are strongly dependent on the polynomials C and A_2, i.e. on the disturbance model.

From (4.7) and (4.8) it is found that:

$$y_f(t+i) = q^{-d} \frac{P_n B}{P_d A A_1} u(t+i) + \frac{\mathcal{F}_i}{P_d A A_2} e(t) + \mathcal{E}_i e(t+i) \tag{4.9}$$

Using the expression of $e(t)$ from (4.1) in (4.9), and taking the Diophantine equation (4.8) into account, straight calculation leads to:

$$y_f(t+i) = \frac{A_2 B}{C A_1} \mathcal{E}_i u(t+i-d) + \frac{\mathcal{F}_i}{C P_n} y_f(t) + \mathcal{E}_i e(t+i) \tag{4.10}$$

where only the last term is independent of signals measurable at time t. The minimal variance i-step ahead predictor, can be obtained by the conditional expectation of $y_f(t+i)$ [1]:

$$\hat{y}_f(t+i) = \frac{A_2 B}{C A_1} \mathcal{E}_i u(t+i-d) + \frac{\mathcal{F}_i}{C P_n} y_f(t) \tag{4.11}$$

[1]The conditional expectation $\hat{y}(t+i|t)$ is denoted simply by $\hat{y}(t+i)$.

where the prediction error for the i-th prediction is given by:

$$\epsilon(t+i) = y_f(t+i) - \hat{y}_f(t+i) = \mathcal{E}_i e(t+i)$$

The next step is to split the control signal into past and future signals, using another Diophantine equation:

$$\frac{B A_2}{A_1 C} \mathcal{E}_i = \mathcal{G}_i + q^{-i+d-1} \frac{\mathcal{L}_i}{A_1 C}, \quad i \geq d \tag{4.12}$$

The solutions of the Diophantine equation are the polynomials $\mathcal{L}_i$ and $\mathcal{G}_i$. Their order is:$n_{\mathcal{L}_i} = \max(n_B + n_{A_2} + n_{\mathcal{E}_i} - i + d - 1, n_{A_1} + n_C - 1)$ and $n_{\mathcal{G}_i} = i - d$. It should be noted that $\mathcal{G}_i$ contains the first i values of the impulse response of $\frac{BA_2}{A_1 C} \mathcal{E}_i$. Thus the prediction (4.11) becomes:

$$\hat{y}_f(t+i) = \underbrace{\mathcal{G}_i u(t+i-d)}_{future} + \underbrace{\frac{\mathcal{L}_i}{A_1 C} u(t-1) + \frac{\mathcal{F}_i}{P_d C} y(t)}_{past} \tag{4.13}$$

It is observed from (4.13) that, the prediction involves two sets of terms: the first which only consists in future control signals (from time t onward), and the second which does not depend on the future controller outputs, being fully determined at the moment t. The last one is the so called free response predictions.

Collecting the i-step ahead predictors for $i = H_m, \ldots, H_p$ and using the expression (4.13), it is get that:

$$\begin{bmatrix} \hat{y}_f(t+H_m) \\ \vdots \\ \hat{y}_f(t+H_p) \end{bmatrix} = \boldsymbol{G} \begin{bmatrix} u(t) \\ \vdots \\ u(t+H_p-d) \end{bmatrix} + \boldsymbol{L} \begin{bmatrix} \frac{u(t-1)}{CA_1} \\ \frac{u(t-2)}{CA_1} \\ \vdots \end{bmatrix} + \boldsymbol{F} \begin{bmatrix} \frac{y(t)}{CP_d} \\ \frac{y(t-1)}{CP_d} \\ \vdots \end{bmatrix} \tag{4.14}$$

The matrices $\boldsymbol{G}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are built up with the elements of the polynomials $\mathcal{G}_i$, $\mathcal{L}_i$ and $\mathcal{F}_i$, respectively. Assuming that $H_m = d$, the matrix $\boldsymbol{G}$ and its dimension are:

$$\boldsymbol{G} = \begin{bmatrix} \mathcal{G}_{H_m} \\ \mathcal{G}_{H_m+1} \\ \vdots \\ \mathcal{G}_{H_p} \end{bmatrix} = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{(H_p-H_m)} & g_{(H_p-H_m-1)} & \cdots & g_0 \end{bmatrix} \tag{4.15}$$

$$[\boldsymbol{G}] = (H_p - H_m + 1) \times (H_p - H_m + 1)$$

If $H_m > d$ then the first $H_m - d$ rows of $\boldsymbol{G}$ will be null and they should be deleted. The other two matrices in (4.14) are:

$$\begin{aligned} \boldsymbol{L} &= \left[\mathcal{L}_{H_m} \cdots \mathcal{L}_{H_p} \right]^T, \quad [\boldsymbol{L}] = (H_p - H_m + 1) \times (\max_i \, (n_{\mathcal{L}_i}) + 1) \\ \boldsymbol{F} &= \left[\mathcal{F}_{H_m} \cdots \mathcal{F}_{H_p} \right]^T, \quad [\boldsymbol{F}] = (H_p - H_m + 1) \times (\max_i \, (n_{\mathcal{F}_i}) + 1) \end{aligned} \tag{4.16}$$

Using the notations:

$$\begin{aligned} \underline{\tilde{u}}_1 &= \left[\tfrac{u_{t-1}}{CA_1}, \; \tfrac{u_{t-2}}{CA_1}, \; \ldots \right]^T, \quad [\underline{\tilde{u}}_1] = (\max_i \, (n_{\mathcal{L}_i}) + 1) \times 1 \\ \underline{\tilde{y}} &= \left[\tfrac{y(t)}{CP_d}, \; \tfrac{y(t-1)}{CP_d}, \; \ldots \right]^T, \quad [\underline{\tilde{y}}] = (\max_i \, (n_{\mathcal{F}_i}) + 1) \times 1 \end{aligned} \tag{4.17}$$

the following matrix expression for the prediction is found:

$$\boxed{\underline{\hat{y}}_f = \boldsymbol{G}\underline{u} + \boldsymbol{L}\underline{\tilde{u}}_1 + \boldsymbol{F}\underline{\tilde{y}}} \tag{4.18}$$

Once the prediction (4.18) is obtained, the next step is to determine the partial derivatives $\frac{\partial \underline{\hat{y}}_f}{\partial \underline{u}_1}$, and $\frac{\partial \underline{u}_f}{\partial \underline{u}_1}$.

Relation between $\underline{u}$ and $\underline{u}_1$

The relationship between $\underline{u}$ and $\underline{u}_1$ can be derived as the solution to the constraint:

$$\phi \, \frac{P_n}{P_d} u(t+i) = 0 \tag{4.19}$$

where $H_c \leq i \leq H_p - d$. This is known as a soft constraint due to the fact that it can be directly incorporated in the controller. The idea that, the future control signals should satisfy a linear constraint, originates from *[Cutler and Ramaker, 1980]*.

The required relationship between $\underline{u}$ and $\underline{u}_1$ can be acquired from the following Diophantine equation:

$$\frac{P_d}{\phi P_n} = \mathcal{V}_i + q^{-i+H_c-1}\frac{\mathcal{Z}_i}{\phi P_n} \tag{4.20}$$

The solutions of the Diophantine equation are the polynomials $\mathcal{V}_i$ and $\mathcal{Z}_i$, of order $n_{\mathcal{V}_i} = i - H_c$ and $n_{\mathcal{Z}_i} = \max\left(n_{P_d} - i + H_c - 1, n_\phi + n_{P_n} - 1\right)$, respectively. Multiplication by $\phi P_n / P_d$ yields:

$$\frac{\phi P_n}{P_d}\mathcal{V}_i = 1 - q^{-i+H_c-1}\frac{\mathcal{Z}_i}{P_d} \tag{4.21}$$

Then considering (4.19) it is found that:

$$u(t+i) = \frac{\mathcal{Z}_i}{P_d}u(t+H_c-1)$$

with $i = H_c, \ldots, H_p - d$.

Separation of the signals into future and past terms can be then obtained from a new Diophantine equation:

$$\frac{\mathcal{Z}_i}{P_d} = \mathcal{M}_i + q^{-H_c}\frac{\mathcal{N}_i}{P_d}$$

where $n_{\mathcal{M}_i} = \min\left(H_c, n_\phi + n_{P_n}\right) - 1$ and $n_{\mathcal{N}_i} = \max\left(n_{\mathcal{Z}_i} - H_c, n_{P_d} - 1\right)$. It can be noted that the orders of $\mathcal{M}_i$ and $\mathcal{N}_i$ are independent of i. Finally:

$$u(t+i) = \mathcal{M}_i u(t+H_c-1) + \frac{\mathcal{N}_i}{P_d}u(t-1)$$

where $H_c \leq i \leq H_p - d$. Collecting the control signals $u(t), \cdots, u(t+H_p-d)$ leads to:

$$\begin{bmatrix} u(t) \\ \vdots \\ u(t+H_c) \\ \vdots \\ u(t+H_p-d) \end{bmatrix} = \boldsymbol{M}\begin{bmatrix} u(t) \\ \vdots \\ u(t+H_c-1) \end{bmatrix} + \boldsymbol{N}\begin{bmatrix} \frac{u(t-1)}{P_d} \\ \frac{u(t-2)}{P_d} \\ \vdots \end{bmatrix} \tag{4.22}$$

where the matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ are:

$$\boldsymbol{M} = \left[\begin{array}{cccccc} 1 & 0 & \ldots & \ldots & 0 & 0 \\ 0 & 1 & & & & \\ \vdots & & \ddots & & & \vdots \\ \vdots & & & \ddots & & \vdots \\ & & & & \ddots & 0 \\ 0 & & \ldots & \ldots & 0 & 1 \\ \hline 0 & \ldots & 0 & m^{H_c}_{n_{\mathcal{M}_i}} & \ldots & m^{H_c}_0 \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \ldots & 0 & m^{H_p-d}_{n_{\mathcal{M}_i}} & \ldots & m^{H_p-d}_0 \end{array}\right] \begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right\} H_c \\ \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} H_p - H_c - d + 1 \end{array} \tag{4.23}$$

$$\boldsymbol{N} = \left[\begin{array}{ccc} 0 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & 0 \\ \hline n_0^{H_c} & \ldots & n_{n_{\mathcal{N}_i}}^{H_c} \\ \vdots & & \vdots \\ n_0^{H_p-d} & \ldots & n_{n_{\mathcal{N}_i}}^{H_p-d} \end{array}\right] \begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} H_c \\ \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} H_p - H_c - d + 1 \end{array} \tag{4.24}$$

The matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ contain the elements of $\mathcal{M}_i$ and $\mathcal{N}_i$ polynomial, respectivelly, for different i ($H_c \leq i \leq H_p - d$):

$$\begin{aligned} \mathcal{M}_i &= m_0^i + m_1^i q^{-1} + \ldots + m_{n_{\mathcal{M}_i}}^i q^{-n_{\mathcal{M}_i}} \\ \mathcal{N}_i &= n_0^i + n_1^i q^{-1} + \ldots + n_{n_{\mathcal{N}_i}}^i q^{-n_{n_{\mathcal{N}_i}}} \end{aligned} \tag{4.25}$$

and their dimensions are $[\boldsymbol{M}] = (H_p - d + 1) \times H_c$ and $[\boldsymbol{N}] = (H_p - d + 1) \times n_{\mathcal{N}_i}$. Note that, if $H_c = H_p - d + 1$, then $\boldsymbol{M} = \boldsymbol{I}$ and $\boldsymbol{N} = \boldsymbol{0}$. Finally, using the notation:

$$\underline{\tilde{u}}_2 = \left[\begin{array}{ccc} \frac{u(t-1)}{P_d}, & \frac{u(t-2)}{P_d}, & \ldots \end{array}\right]^T$$

the relationship between $\underline{u}$ and $\underline{u}_1$ can be written in the compact form:

$$\boxed{\underline{u} = \boldsymbol{M}\underline{u}_1 + \boldsymbol{N}\underline{\tilde{u}}_2} \tag{4.26}$$

Relation between $\underline{u}$ and $\underline{u}_f$

In the derivation of (4.6), the relationship between $\underline{u}$ and $\underline{u}_f$ is also needed. Therefore, the expression:

$$u_f(t+i) \;=\; \frac{Q_n}{Q_d}\, u(t+i), \quad 0 \leq i \leq H_p - d \tag{4.27}$$

will be used. The right-hand side term can be split into future and past signals, by solving the Diophantine equation:

$$\frac{Q_n}{Q_d} = \Phi_i + q^{-i-1}\frac{\Omega_i}{Q_d}$$

where $i = 0, \ldots, H_p - d$, $n_{\Phi_i} = i$, $n_{\Omega_i} = \max(n_{Q_n} - i - 1, n_{Q_d} - 1)$. Using this Diophantine equation, (4.27) becomes:

$$u_f(t+i) = \Phi_i u(t+i) + \frac{\Omega_i}{Q_d} u(t-1) \tag{4.28}$$

Collecting the signals for $0 \leq i \leq H_p - d$ and using the notation:

$$\underline{\tilde{u}}_3 = \left[\begin{array}{ccc} \frac{u(t-1)}{Q_d}, & \frac{u(t-2)}{Q_d}, & \ldots \end{array}\right]^T$$

the relation between $\underline{u}$ and $\underline{u}_f$ is found as:

$$\boxed{\underline{u}_f = \boldsymbol{\Phi}\underline{u} + \boldsymbol{\Omega}\underline{\tilde{u}}_3} \tag{4.29}$$

where $\boldsymbol{\Phi}$ ($[\boldsymbol{\Phi}] = (H_p - d + 1) \times (H_p - d + 1)$) is a lower triangular matrix of dimension. It consists of the elements of the polynomial Φ_i, in a similar way to the construction of $\boldsymbol{G}$ shown in (4.15). The dimension of matrix $\boldsymbol{\Omega}$ is $(H_p - d + 1) \times \max(n_{\Omega_i})$.

Collecting the results (4.18), (4.26), (4.29):

$$\begin{aligned} \underline{\hat{y}}_f &= \boldsymbol{G}\underline{u} + \boldsymbol{L}\underline{\tilde{u}}_1 + \boldsymbol{F}\underline{\tilde{y}} \\ \underline{u} &= \boldsymbol{M}\underline{u}_1 + \boldsymbol{N}\underline{\tilde{u}}_2 \\ \underline{u}_f &= \boldsymbol{\Phi}\underline{u} + \boldsymbol{\Omega}\underline{\tilde{u}}_3 \end{aligned} \tag{4.30}$$

it is obtained that:

$$\begin{aligned} \underline{\hat{y}}_f &= \boldsymbol{G}\boldsymbol{M}\underline{u}_1 + \boldsymbol{G}\boldsymbol{N}\underline{\tilde{u}}_2 + \boldsymbol{L}\underline{\tilde{u}}_1 + \boldsymbol{F}\underline{\tilde{y}} \\ \underline{u}_f &= \boldsymbol{\Phi}\boldsymbol{M}\underline{u}_1 + \boldsymbol{\Phi}\boldsymbol{N}\underline{\tilde{u}}_2 + \boldsymbol{\Omega}\underline{\tilde{u}}_3 \end{aligned} \tag{4.31}$$

Thus, the partial derivatives in (4.6) are obtained as:

$$\frac{\partial \underline{\hat{y}}_f}{\partial \underline{u}_1} = \boldsymbol{M}^T\boldsymbol{G}^T, \qquad \frac{\partial \underline{u}_f}{\partial \underline{u}_1} = \boldsymbol{M}^T\boldsymbol{\Phi}^T \tag{4.32}$$

The gradient (4.6) becomes:

$$\begin{aligned} \frac{\partial J}{\partial \underline{u}_1} &= 2\boldsymbol{M}^T[\boldsymbol{G}^T\boldsymbol{G} + \rho\boldsymbol{\Phi}^T\boldsymbol{\Phi}]\boldsymbol{M}\underline{u}_1 + 2\boldsymbol{M}^T[\boldsymbol{G}^T(\boldsymbol{L}\underline{\tilde{u}}_1 \\ &+ \boldsymbol{F}\underline{\tilde{y}} - \underline{w} + \boldsymbol{G}\boldsymbol{N}\underline{\tilde{u}}_2) + \rho\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{N}\underline{\tilde{u}}_2 + \boldsymbol{\Omega}\underline{\tilde{u}}_3)] \end{aligned} \tag{4.33}$$

while the Hessian matrix is found as:

$$\boldsymbol{H} = 2\boldsymbol{M}^T(\boldsymbol{G}^T\boldsymbol{G} + \rho\boldsymbol{\Phi}^T\boldsymbol{\Phi})\boldsymbol{M} \tag{4.34}$$

Once the gradient (4.33) is obtained, the optimal solution is given by:

$$\begin{aligned} \underline{u}_1 &= \tfrac{1}{2}\boldsymbol{H}^{-1}\boldsymbol{M}^T[\boldsymbol{G}^T(\underline{w} - \boldsymbol{L}\underline{\tilde{u}}_1 - \boldsymbol{F}\underline{\tilde{y}} - \boldsymbol{G}\boldsymbol{N}\underline{\tilde{u}}_2) \\ &- \rho\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{N}\underline{\tilde{u}}_2 + \boldsymbol{\Omega}\underline{\tilde{u}}_3] \end{aligned} \tag{4.35}$$

The inversion of the Hessian $\boldsymbol{H}$ is of crucial importance for the control. When choosing the control weighting factor $\rho > 0$, the matrix $\boldsymbol{H}$ is always invertible, but ρ is often hard to determine a priori, so therefore usually ρ=0. In this case, H_m should be chosen so that $H_m \geq d$ and $H_p - H_m + 1 \geq H_c$ in order to keep $\boldsymbol{H}$ positive definite (see e.g. *[Lauritsen, 1994]*).

Although the signals $u(t), \cdots, u(t + H_c - 1)$ are calculated at each sample instant, only the first of these is applied to the system (receding horizon effect). The reason for using the receding horizon approach is that it allows to compensate for future disturbances or modelling errors. Thus future disturbances and modelling errors are taken into account in the design of the controller.

Straight calculations, see e.g. *[Soeterboek, 1991]*, yield the well-known polynomial form of the control law:

$$R(q^{-1})u(t) = Q(q^{-1})w(t) - S(q^{-1})y(t) \tag{4.36}$$

which is a convenient representation with respect to the implementation and the analysis of the controller.

In the predictive controller, future values of the reference signal appear. In general, these values are not known, but if they are known then, this prior knowledge should be exploited, because the long range predictive controllers have the advantage to use this extra information. The trajectory may be an arbitrary sequence of points describing a path that e.g. a ship must track (see e.g. Appendix E). However, in the cases, when the future reference trajectory is not known a priori, it may be predicted or some assumptions about the future reference can be done. For example, a simple first order trajectory can be defined as:

$$w(t+i) = S_p + i\,\beta, \quad i = H_m, \ldots, H_p \tag{4.37}$$

where β indicates the increment at each sampling time and S_p is the set-point at instant t, which is going to maintain a fixed value. Another simple solution is to assume that the future reference signal is constant and equal to the current set-point S_p, i.e.:

$$w(t+i) = w(t), \quad i = H_m, \ldots, H_p \tag{4.38}$$

The way the trajectory is generated plays an important role in the implementation of the predictive controller.

4.3 Choice of design parameters

The behavior of a predictive controller is strongly dependent on the values of the set of design parameters. The choice of these parameters (presented in Table 4.1) may be rather complicated, as there is no clear theory which predicts the closed loop behavior for arbitrary horizons and weighting factors. However, in the technical literature (e.g. *[Clarke et al., 1987]*, *[Soeterboek, 1991]*), some guidelines for the choice of these parameters can be found:

- H_m: should be chosen equal to the system time delay d. If the time delay is unknown then, H_m can be set on 1 *[Clarke et al., 1987]*. H_m should not be used as a tuning parameter because, e.g. in the case of non-minimum phase process an unpredictable behavior of the closed loop system can result for other values of H_m*[Soeterboek, 1991]*. The reason is that by increasing H_m, the tracking error in the near future is not taken into account in the criterion function.

- $H_c \leq n$: A proper value of this design parameter depends on the complexity of the system[1]. The choice of H_c can be seen as a compromise between the performance and robustness. For instance, increasing H_c leads to more active control, faster closed loop, but also to a poorer stability robustness. As shown in (4.35), the solution of the optimization problem involves inversion of the Hessian matrix, $[\boldsymbol{H}] = H_c \times H_c$. Therefore it is of interest to keep H_c at an acceptable minimum, where both the computational effort and the qualities associated with the resulting controller, are considered. In general, for simple systems,

[1] n is the order of the system.

$H_c = 1$ is acceptable, but in the case of 'difficult' plants, a good control is achieved when H_c is at least equal to the number of unstable/underdamped poles *[Clarke et al., 1987]*.

- H_p large: For open loop stable systems it is recommended to choose a value of H_p close to the rise time of the system. The choice should ensure that, all output values significantly affected by the current control, are included in the cost function. When H_p is increased, the stability of the system is improved and the closed loop system becomes slower. In the case of non-minimum phase systems, it is recommended *[Soeterboek, 1991]* that, H_p is chosen, related with the bandwidth of the process:

$$H_p = int(2\,\omega_s/\omega_b)$$

where ω_s is the sampling frequency, ω_b is the bandwidth of the open loop process and *int* denotes the integer part of the fraction.

- ρ small: ρ is difficult to choose a priori. The easiest choice is $\rho = 0$, even though sometimes for numerical reasons it may be convenient to choose ρ as a small value. In many cases increasing the weighting factor ρ makes the closed loop system slower.

- $\frac{Q_n}{Q_d}$: is used to enhance or attenuate certain frequencies in the control signal.

- $\frac{P_n}{P_d}$: it is used to tune the servo and regulator behavior of the closed loop system. In order to avoid steady state errors, the condition

$\frac{P_n(1)}{P_d(1)} = 1$ should be satisfied. According to *[Soeterboek, 1991]*, polynomial P_n can be used to define the desired servo behaviour of the closed loop system. Thus P_n can be chosen as:

$$P_n = 1 - \alpha q^{-1} \tag{4.39}$$

where $\alpha = e^{-2.3T_s/t_r}$ and t_r is the desired rise time of the closed loop system. A second order polynomial can be also used *[Soeterboek, 1991]*.

- ϕ and A_2: are used in order to get offset free control. For this ϕ and A_2 must have as factor $1 - q^{-1}$.

- C: is usually denoted by T, in the case when C is difficult to estimate[1]. T is known as the observer polynomial and can be thought as a design parameter, used to improve both the robustness and the regulator behaviour of the closed loop system. Slow observer poles are used to improve the robustness, while fast poles are used for good disturbance rejection *[Robinson and Clarke, 1991]*. Thus a compromise between these two aspects must exist for a proper design of T. In the ideal case, when the process model is exact, T has no influence on the servo perfromance *[Soeterboek, 1991]* and therefore it can be merely used to reject the disturbances *[Robinson and Clarke, 1991]*. Details on the effect of the observer polynomial can be found in *[Soeterboek, 1991]*, *[Mohtadi, 1988b]* and *[Robinson and Clarke, 1991]*.

[1] The identification of the disturbance model is usually difficult, because the algorithm developed for this purpose (RML, RELS) are subjected to convergence problems and they are computationally expensive *[Ljung, 1987]*.

4.4 Track keeping systems

The purpose of this section is to illustrate the ability of the predictive control to deal with a pre-scheduled reference trajectory, as e.g. is the case of a predictive route tracking system for the ship[2].

The classical autopilot of sea-going vessels is a course controller. To solve a navigation problem, especially in the presence of current, from time to time, on board, some corrections of the heading angle are done, usually by hand, in order to adjust the real position to a desired position. However, in a number of practical applications (e.g. mine hunters, dredging vessels, and also boats which have to follow a water passage with a certain depth), the deviation of the ship from the desired trajectory must be taken automatically into account in the design of the controller. Therefore the position of the ship must be available.

The track controllers should get informations about the position of the ship from a navigation system. Over the years, different navigation systems such as Omega, Loran C, Decca were used. Their precision was not always satisfactory (e.g. the tracking errors for Decca systems are about 200-300 meters). Recently, the satellite navigation system (GPS - Global Positioning System, DGPS - Differential Global Positioning System) has been adopted. It is very attractive because, combined with a Kalman filter algorithm *[Fossen, 1994]*, it can provide a high precision for the position of the ship (5-10 meters).

[2] In *[Holzhüter, 1990]*, the optimal tracking problem for a ship using a LQG controller and filtering theory is discussed.

Hence, due to the recent sophisticated measuring equipment, typically available today on board ship, the present study case, considers the design of a tracking controller based on the assumption that, the states are measurable.

4.4.1 Statement of the tracking problem

A route tracking system is usually designed such that the ship can move forward with constant speed U at the same time as, the sway position (deviation) is minimized. A navigation example, when a ship should follow a desired predefined trajectory, is sketched in Figure 4.1.

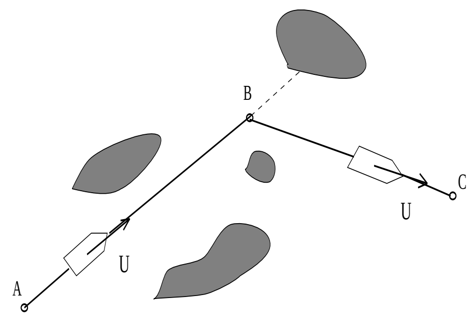

Figure 4.1. *Narrow passage.*

The route tracking system consists of:

- The route generator – it provides a list of way-points, which constitutes a desired route.
- The tracking controller – it must manage track changing maneuvers.

4.4.2 Maneuvering dynamics and kinematics

In this section the maneuvering dynamics, the kinematics and the assumptions used in the design of a track keeping system are summarized.

The modelling of the forces, which act on the ship, were described in Chapter 2. As in the design of a route tracking system the surge speed u of the ship is assumed constant, the steering equations must be taken into account. According to *[Blanke, 1981]*, the steering equations (see Chapter 2) can be written as:

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \boldsymbol{A}\, u \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \boldsymbol{B}\, u^2\, \delta + \boldsymbol{C} \begin{bmatrix} v^3 \\ r v^2 \end{bmatrix} + \boldsymbol{D} \begin{bmatrix} Y_{ext} \\ N_{ext} \end{bmatrix} \qquad (4.40)$$

where:

u	the forward speed [m/s].
v	the sway velocity [m/s].
r	the yaw rate [deg/s].
ψ	the heading angle [deg].
δ	the rudder angle [deg].
Y_{ext}, N_{ext}	external disturbances force and moment which act on the ship.

and $\boldsymbol{A} \in R^{3\times3}$, $\boldsymbol{B} \in R^{3\times1}$, $\boldsymbol{C} \in R^{3\times2}$, $\boldsymbol{D} \in R^{3\times2}$.

The assumptions, which are used in the design of a track system, are as follows:

- The ship moves at a constant forward speed ($u \approx U$). For example, for the Mariner Class Vessel $U = 7.71$ [m/s] (15 knots).
- The travel time is not fixed, thus the navigation between two points is not constrained by time.
- The states of the model (4.40), the intensity and the direction of the disturbances are measurable.
- the sea is calm enough so the effect of roll, pitch and heave motions can be neglected.

The basic coordinate definitions (see Chapter 2) for the description of the ship's motion are sketched in Figure 4.2.

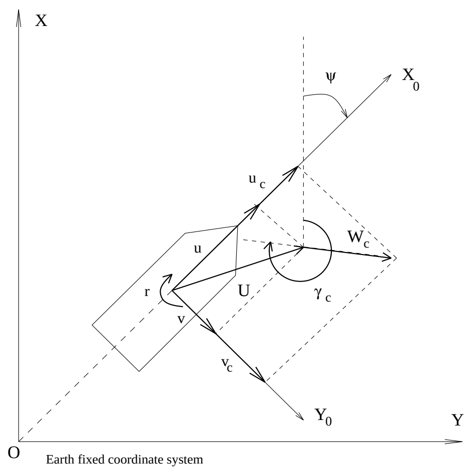

Figure 4.2. *Coordinate systems for global tracking.*

According to this, the kinematic relations can be expressed as:

$$\begin{aligned} \dot{x} &= (u + u_c)\ \cos(\psi) - (v + v_c)\ \sin(\psi) \\ \dot{y} &= (u + u_c)\ \sin(\psi) + (v + v_c)\ \cos(\psi) \end{aligned} \tag{4.41}$$

(x, y) and (u_c, v_c) are the two components of the position and of the current speed, respectively, in the local coordinate system:

$$\begin{aligned} u_c &= W_c\ \cos(\gamma_c - \psi - \pi) \\ v_c &= W_c\ \sin(\gamma_c - \psi - \pi) \end{aligned} \tag{4.42}$$

W_c and γ_c, defined in Chapter 2, represent the speed and the direction of the current, respectively. It can be noticed that the expressions (4.41) are nonlinear in the states u, v and ψ.

4.4.3 Tracking and coordinate systems

In the design of a tracking controller, the relation between the coordinate systems is important. The coordinate systems are pointed out in Figure 4.3, where an example of a desired route composed of two straight lines, is sketched. The route is defined by three turning points A, B, C.

In the present work, the tracking problem is solved by taking three coordinate systems into account :

- the global coordinate system $X_o O_o Y_o$ (Earth fixed coordinate system).

- the reference coordinate system (e.g. $X_{r1} A\, Y_{r1}$) tangent to the trajectory (e.g. linear section AB). This coordinate system is not fixed.

- the local coordinate system XOY - fixed to the body of the ship.

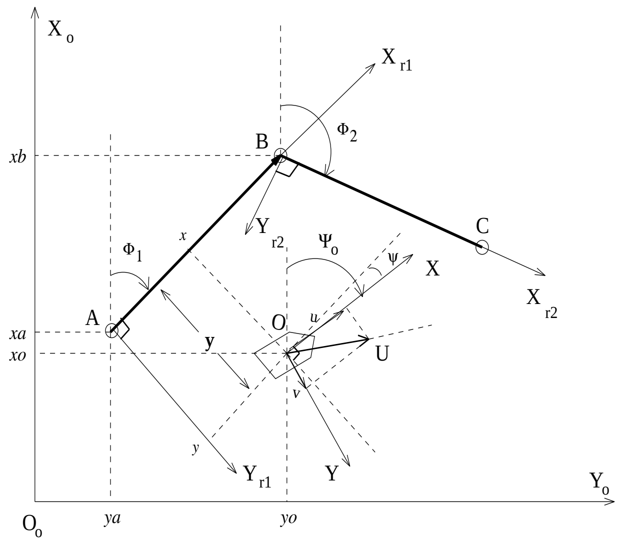

Figure 4.3. *Global, reference and local coordinate systems.*

For computational convenience, the tracking error of the ship is defined directly in the reference coordinate system, as the perpendicular distance y of the ship's center of gravity to the track reference line (see Figure 4.3). Thus, as a performance index, it is natural to adopt:

$$J_{tr} = \min_t \sum y^2(t) \tag{4.43}$$

where the tracking error y is computed using (4.41). In the reference coordinate system, the local heading angle ψ is defined as the angle between the track reference line and the ship's centreline (e.g. in Figure 4.3, $\psi = \Psi_0 - \Phi_1$).

Each time the reference coordinate system changes, the direction of the current, wind and waves related to the ship must be recomputed. However, each time, the position and the heading angle should be expressed, both in the reference coordinate system (for determining the controller output signal) and in the global coordinate system (for plotting the ship's track). In Figure 4.3, e.g. once the coordinates of the ship are computed in the reference coordinate system (i.e. $X_{r1}AY_{r1}$) using relation (4.41), the corresponding global coordinates and the heading angle are computed by:

$$\begin{aligned} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} &= \begin{bmatrix} \cos(\Phi_1) & -\sin(\Phi_1) \\ \sin(\Phi_1) & \cos(\Phi_1) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} x_a \\ y_a \end{bmatrix} \\ \Psi_o &= \Phi_1 + \psi \end{aligned} \tag{4.44}$$

where Φ_1 is the rotation of the reference coordinates system with respect to the global coordinate system:

$$\Phi_1 = arctg[\frac{yb - ya}{xb - xa}] \tag{4.45}$$

where (xa, ya), (xb, yb) are the global coordinates of the points A and B, respectively, while (x_0, y_0) and $\boldsymbol{\psi}_0$ are the position and heading angle of the ship, respectively, in the global coordinate system. When there are no disturbances or reference change, $\psi = 0$ and thus $\boldsymbol{\Psi}_0 = \Phi_1$. If there exists

any current then $\psi \neq 0$ because a correction must be added to the desired course of the ship in order to correct the course with respect to the desired track.

At each turning point, when a new reference trajectory is defined, the position of the ship should be updated corresponding to the new reference coordinate system.

4.4.4 Route generator and the approach of a new trajectory section

The route generator generates the desired route from a given set of coordinates. If weather data are available, the fuel can be saved by the generation of a trajectory such that the ship's wind and water resistance are minimized *[Fossen, 1994]*.

When the trajectory consists of different sections to be approached, the way and the moment of changing the section must be defined. There are different possibilities in going from one section to another:

- Manually - where at the end of the section the track keeping controller is switched off and the ship is manually brought into the neighborhood of the next section. It has the disadvantage of a possible large overshoot.

- By defining turning points – For example, a scheduled route can be calculated as a combination of straight lines and circular arcs, when a set of global coordinates (X_i, Y_i), $(i = 1, \cdots, n)$ for each turning point P_i and the desired turning radii R_j $(j = 1, \cdots, n-2)$ are available.

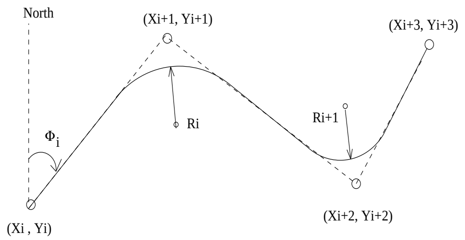

Figure 4.4. *Scheduled route generation.*

Once the straight lines are defined by connecting the successive turning points, the generation of the circular curves with radii R_j are uniquely determined (see Appendix E). Thus, depending on the turning point P_i and the radius R_j, the change moment from one section to another it can be computed in advance *[Amerongen and Land, 1980]*. Figure 4.4 shows an example of route generation by defining turning points.

The direction of the reference trajectory related to the global coordinate system can be computed by:

$$\Phi_i = arctg[\frac{Y_{i+1} - Y_i}{X_{i+1} - X_i}] \tag{4.46}$$

The size of the radius is flexible, depending on the maneuvering demand and the disturbances. For example, a big radius in open sea maneuvering can be chosen, but not in confined waters also. In the case of current's presence, a rather large turning circle has to be assumed, because the turning circle dimension depends strongly on the angle between the current direction and the ship's heading *[Amerongen and Land, 1980]*.

- Automatically – when the decision to change from one section to another is made automatically, based on the previous knowledge about the track and environmental conditions. For example, the predictive control approach is able to switch automatically, in advance, to a new section, based on the prediction horizon H_p.

In this work, the automatical method is adopted by using a predictive autopilot. The predictive controller has the ability to predict H_p steps-ahead and to detect the change of the trajectory in advance. In this way, a large overshoot can be avoided.

4.4.5 Predictive tracking controller

The tracking controller must provide both track-keeping and track-changing capabilities. Track-changing is however reduced to a track-keeping problem due to the pre-computation of the reference trajectory.

The tracking controller should minimize the mean squares of the deviation y from the desired track:

$$J_{tr} = \min_t \sum y^2(t) \tag{4.47}$$

In the design of the controller, mainly the behavior of the ship on straight tracks is important. Thus, due to the fact that generally only small rudder angles δ are involved, a simple linear model may be used in the design and the steering machine dynamics may be disregarded.

It is assumed that the ship moves with constant speed U (nominal speed) along the trajectory and the relative heading angle ψ is small (see Figure 4.3). The stochastic nonlinear system and the nonlinear kinematics relation (4.41) are linearized along each reference trajectory[1]. The linear approximation is derived under the assumption that the desired local heading angle is $\psi = 0$. Hence, during track control, the heading angle ψ will be close to zero, and the following approximations will be valid:

$$\sin(\psi) \approx \psi, \quad \cos(\psi) \approx 1 \tag{4.48}$$

Thus the kinematic equations of motion (4.41) can be reduced to the following set of linear equations (where $u \approx U$):

$$\begin{aligned} \dot{x} &= U + d_x \\ \dot{y} &= U\psi + v + d_y \end{aligned} \tag{4.49}$$

where d_x, d_y describe the errors due to the linearization and drift caused by the environmental disturbances.

In the design of the controller, the fully linearized model, is reduced to a low order model by using e.g. the Balanced Truncation Approximation method, described in Chapter 3. Thus, the reduced transfer functions from rudder δ to yaw rate r and to sway speed v, respectively, are determined:

$$r(s) = \tilde{G}_{r\delta}\,\delta(s), \quad v(s) = \tilde{G}_{v\delta}\,\delta(s) \tag{4.50}$$

[1] The equations can be normalized, in dimensionless form, using ship length L, velocity U and time t as basis for the normalization *[Holzhüter, 1990]*.

The reduced form of the transfer function which models the effect of the stochastic disturbances on the ship is also get as:

$$\eta(s) = \tilde{H}_{\eta e}\, e(s) \tag{4.51}$$

where $e(s)$ is a zero-mean Gaussian white noise process with constant variance. In practice, it is found that identifying a disturbance model $\tilde{H}_{\eta e}$ is difficult, because the algorithms developed for this purpose (RML, RELS) are subjected to convergence problems or are computationally burdensome. Therefore in the design of the controller, design polynomials usually are used instead of the polynomials of the transfer function $\tilde{H}_{\eta e}$.

Further, as it was pointed out in Chapter 2, in the absence of the roll and pitch modes ($\Phi = \theta = 0$), the heading angle is simply defined as:

$$\dot{\psi} = r \tag{4.52}$$

Taking (4.49), (4.50), (4.51), (4.52) into account, the resulting reduced model, which expresses the tracking error y:

$$y(s) = \frac{U}{s^2}\,\tilde{G}_{r\delta}\,\delta(s) + \frac{1}{s}\,\tilde{G}_{v\delta}\,\delta(s) + \frac{1}{s}\,\eta(s) \tag{4.53}$$

The discrete form of (4.53) can be then written as:

$$A(q^{-1})y(t) = q^{-d}\frac{B(q^{-1})}{A_1(q^{-1})}\delta(t) + \frac{C(q^{-1})}{A_2(q^{-1})}e(t) \tag{4.54}$$

As it was mentioned, the goal of this section is to develop the tracking controller as a predictive controller. The reasons are that it is suitable to control non-minimum phase systems, to handle pre-computed reference trajectory. Therefore, the cost function (4.2) should be minimized with respect to the controller output sequence over the control horizon, as in Section 4.2. In the design it is assumed that the states are measurable. The control law can be finally expressed by:

$$R(q^{-1})u(t) = Q(q^{-1})w(t) - S(q^{-1})\delta(t) \tag{4.55}$$

which is a convenient representation for the implementation and the analysis of the controller. According to the receding horizon control approach (summarized on page 98), at each sample, only the first element of the controller output sequence from (4.55) is used to control the system.

It should be noted that, in each new reference coordinate system the incidence of current, wind and waves must be recomputed, and thus new parameters of the controller have to be determined.

4.4.6 Computer simulation

The presented predictive tracking system for the Mariner Class vessel is now illustrated. The data of the ship are given in Appendix A. Rudder angle is limited to ± 30 deg. The circumstances during the experiment are as follows:

- Speed of the ship $U = 7.71$ [m/s] (15 knots).
- Current: $u_c = 1$ [m/s] in direction 270 [deg].
- Wind and Waves: direction 180 [deg]; Mean wind speed 15 [m/s].

The derivation of the tracking controller follows the outlines of Sections 4.2 and 4.4.5. As the bandwidth of the open loop system is $\omega_b \approx 0.01$ rad/s, a sampling period $T_s = 12$ seconds ($\omega_s = 8\,\omega_b$) can be chosen. It should be noticed that the obtained model is non-minimum phase and contains a double integrator.

The desired route is e.g. supposed to have 4 sections. It is illustrated by dotted line in Figure 4.5. The start point A(-3000,500) is deliberately chosen not on the desired route, in order to check the ability of the controller to track the reference. Thus the initial deviation of the ship is 500 meters. During the track keeping mode the reference vector $\underline{w}$ is a zero vector. The track changing mode, from one section to another, is started in the moment when the distance to the next turning point is less than $\{H_p(U + u_c)T_s\}$ meters. In the changing mode, $\underline{w}$ is gradually updated, such that element $\{H_p - m\}$ of $\underline{w}$ ($m = 0, \cdots H_p - 1$) step by step, is replaced with the desired deviation with respect to the present reference coordinate system.

Several simulations are performed in order to investigate the influence of the most important design parameters. The performance of the controller is judged by the following criteria:

- The variance of the process output σ_y^2, which is defined as:

$$\sigma_y^2 = \frac{1}{N+1}\sum_{i=0}^{N} y^2(t+i) \qquad (4.56)$$

where N is the number of samples over which the criterion is calculated.

- The variance of the controller output σ_u^2, defined as:

$$\sigma_u^2 = \frac{1}{N+1}\sum_{i=0}^{N} u^2(t+i-1) \qquad (4.57)$$

Different parameters can be tuned, but according to *[Soeterboek, 1991]*, only some crucial parameters of the algorithm are usually used to change the overall performance of the system. As tuning parameters, H_c and the polynomials T and P will be considered. The other parameters of the model are kept constant during the tuning procedure:

$$\begin{array}{ll} H_m = 1 & H_p = 16 \\ Q_n = 1 & Q_d = 1 \\ A_1 = 1 & \rho = 0.01 \\ \phi = [1-q^{-1}] & A_2 = [1-q^{-1}] \end{array} \qquad (4.58)$$

The parameters are chosen according to the guidelines presented in Section 4.3. Thus, for a non-minimum phase model, as is the case of (4.54), the choice of H_p should be done using the rule of thumb: $H_p = int(2\,\omega_s/\omega_b)$. Therefore, $H_p = 16$ is used in the simulations. The bigger prediction horizon H_p is, the better the performance is. It was observed that in the case when

H_p is smaller, the closed loop system has bad performance or even becomes unstable. Further, as the design model (4.54) has double integrator, H_c should be least equal to 2. Making H_c larger will increase the controller output variance. The polynomial A_2 and ϕ are chosen to ensure a zero steady state error[1].

H_c	Variance	$T=(1-0.85q^{-1})$	
		$P_n = 1$, $P_d = 1$	$P_n = (1-0.576q^{-1})$, $P_d = 0.424$
2	σ_u^2	$7.273 \cdot 10^4$	$4.011 \cdot 10^4$
	σ_y^2	$4.837 \cdot 10^6$	$5.918 \cdot 10^6$
3	σ_u^2	$1.703 \cdot 10^5$	$1.157 \cdot 10^5$
	σ_y^2	$6.138 \cdot 10^6$	$4.722 \cdot 10^6$

Table 4.2. *Results for* $T = (1-0.85q^{-1})$.

H_c	Variance	$T=(1-0.85q^{-1})^2$	
		$P_n = 1$, $P_d = 1$	$P_n = (1-0.576q^{-1})$, $P_d = 0.424$
2	σ_u^2	$2.691 \cdot 10^4$	$2.208 \cdot 10^4$
	σ_y^2	$4.957 \cdot 10^6$	$6.808 \cdot 10^6$
3	σ_u^2	$9.159 \cdot 10^4$	$4.593 \cdot 10^4$
	σ_y^2	$4.135 \cdot 10^6$	$3.786 \cdot 10^6$

Table 4.3. *Results for* $T = (1-0.85q^{-1})^2$.

The results are presented in Table 4.2 and Table 4.3. It is known that the observer polynomial T can be used to decrease the variance of the

[1]Low frequency load due to the current and mean speed of the wind may exist.

controller output. Thus, it is observed that if the order of the observer polynomial T is increased, the controller output variance σ_u^2 is significantly decreased, while the process output variance σ_y^2 is hardly affected. It is also clear that, increasing H_c, makes the controller more active and thus the closed loop system faster. The influence of the polynomial P_n is easy to observe in Tables 4.2 and 4.3. Two cases are considered: $P_n = 1$ and $P_n = [1 - 0.576q^{-1}]$ (which corresponds to the rise time t_r equal to 50 seconds). By simulations, it is observed that the polynomial P_n can be chosen to reduce the overshoot of the system and the size of the control signal.

The results corresponding to the case $H_c = 3$, $T = (1 - 0.85q^{-1})^2$, $P_n = (1 - 0.576q^{-1})$ and $P_d = 0.424$ are illustrated in Figure 4.5, 4.6 and 4.7. There is a clear difference between the different sections and the change from one to another, due to the influence of the current, wind and waves.

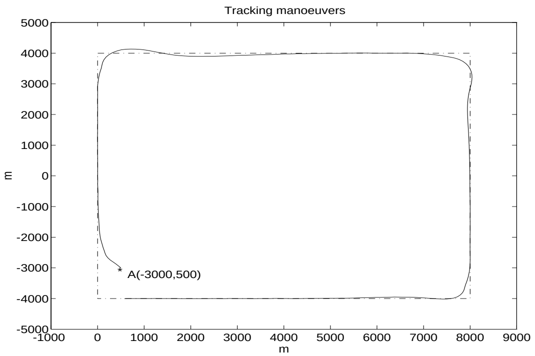

Figure 4.5. *Tracking curve.*

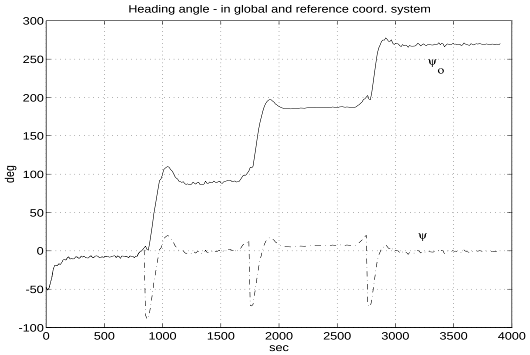

Figure 4.6. *Heading angle.*

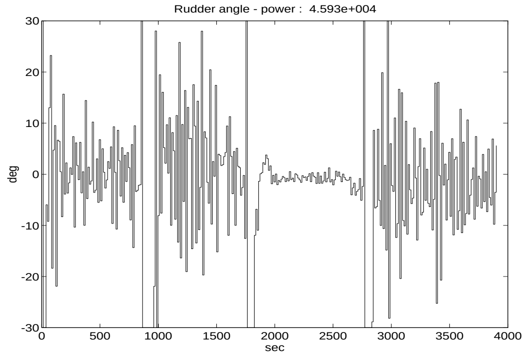

Figure 4.7. *Rudder signal.*

In Figure 4.5, the track plot is shown. The desired trajectory, in the global coordinate system, is represented by a dotted line, while the ship's track is represented by the solid line. Firstly, the ship has to follow a straight line with port beam current and wind and waves from stern. Secondly, it has to track a straight line with stern current and starboard beam wind and waves. The third trajectory has starboard beam current, and ahead wind and waves. The last section is a straight line against the current and port beam wind and waves. It is observed that the controller acts predictively in the moment when the ship is to sail from one section to another. During the straight track keeping phase the deviation of the ship from the desired track is below 4 meters. It should be noted that some simulations were done, with the speed of the current increased to 3 m/s. This corresponds to a rather stormy weather. As it was expected, the navigation was very sensitive to the generation of the scheduled route and to the ration between the speed of the ship and the current speed. It was observed that the ration between the speed of the ship and the current velocity may not be smaller than 2, fact which is also confirmed by *[Amerongen and Land, 1980]*.

In Figure 4.6, two plots are sketched: the dotted line shows the heading angle ψ in the reference coordinate system, while the solid line represents the heading angle Ψ_0 in the global coordinate system. It is observed that to compensate the sideboard current, $\psi \neq 0$ (i.e. first and third sections of the trajectory — Figure 4.5 and 4.6). When the current comes from a side, a correction should be added to the desired local heading angle of the ship (in the reference coordinate system) in order to realize the correct course with respect to the desired trajectory in the global coordinate system. The effect of the current can also be observed on the last section of the trajectory, where the navigation takes longer time compared with the second section. This is due to the fact that, on the last section, the current is against the ship, while on the second section the current is at ship's back.

Figure 4.7 shows the control signal obtained from the controller. The control input is restricted to a range between −30 deg and 30 deg. It is observed that on the third trajectory section, the energy in the control signal is small comparing with the other parts. It indicates that it is easier to control the ship, when the ship has ahead wind and waves.

4.5 Summary

The purpose of this chapter was to discuss the unconstrained predictive control approach, in order to lay the foundations of the optimal linear constrained predictive control, which will be describe in the next chapter. The design and the influence of the design parameters of the controller on the performance and robustness of the closed loop system were discussed.

In this presentation, the application of the unconstrained predictive control approach on the design and implementation of a route tracking system for the Mariner Class Vessel, was illustrated. The synthesis of the ship's control system, based upon the minimization of the mean squares prediction of the ship's deviation from the desired track, was presented. The performance of this route tracking system was tested by numerical simulations on a nonlinear stochastic ship model. The disturbances acting on the ship (current, wind and waves) were compensated quite well. Thus in the tracking keeping mode, the deviations from the ideal path were within few meters.

Chapter 5

Optimal linear constrained predictive control

It is well-known that in many practical situations, the control systems are subjected to hard constraints on system input, output and/or state signals due to the system physical and/or implementation restrictions (e.g. actuator saturation). Ignoring these hard constraints in the controller design may lead to poor performances or even instability of the control system. Several ad-hoc techniques have been proposed in the literature, but their applicability is restricted by the lack of supporting theory and by the fact that this solution is far from optimal. Therefore the devising of control strategies, which take the constraints into account in an explicit and systematic way, is of both theoretical and practical interest. Despite this importance,

little attention was paid to the literature concerning the fact that the hard restrictions should explicitly be taken into account in the controller design specifications.

As it was mentioned in Chapter 4, one of the most attractive feature of predictive control is that it has the ability to handle the process constraints in a natural and systematic way during the design of the controller. Thus, in order to calculate an optimal controller output sequence over the control horizon in the presence of hard constraints, a multi-step quadratic cost function, augmented with inequality equations, has to be minimized. This minimization can be achieved by using various optimization techniques, e.g. quadratic programming methods (QP), see e.g. *[Luenberger, 1984]*, or gradient projection methods *[Bazaraa and Shetty, 1979]*. In predictive controllers, QP techniques have been used, e.g. by *[Garcia and Morshedi, 1984]*, *[Camacho, 1993]* and *[Tsang and Clarke, 1988]*. However, QP methods are very complex and time consuming, a fact which can be very unfavorable in real time applications. Therefore, the gradient projection method *[Rosen, 1960]*, used in the minimization of the multi-step quadratic cost function, seems to be more attractive *[Soeterboek, 1991]*.

The present chapter is concerned with the gradient projection method adapted to minimize a multi-step quadratic cost function in the presence of level and rate constraints. The reasons for selecting this method are:

- It can handle both level and rate constraints (which frequently occur in practice).
- It can be used for multi-step complex cost functions.
- It is not so computationally demanding as QP methods.

To handle hard constraints, a repeated optimization is required to determine which constraints are active. The resulting predictive control is then nonlinear, even for a linear plant.

The analysis in this chapter is motivated, in particular, by the problem present in the steering machine of a ship: i.e. rudder angle is limited to e.g. plus or minus 30 deg, while the speed of the rudder is limited to e.g. plus or minus 3.5 deg/s. The performance of the closed loop system and the numerical complexity involved in solving the constrained optimization problem are illustrated by simulations on a stochastic linear model of the Mariner Class Vessel.

5.1 Statement of the control problem

It is customary to split the subject of constrained optimization into two main parts: 1) linear programming and 2) nonlinear programming. Linear programming treats the case when the constraints and the cost function are all linear in the system variables. If the variables appear in a nonlinear way in constraints or in the cost function, then the problem is considered to be a nonlinear programming problem *[Rosen, 1960]*. The present chapter focuses on linear constraints with nonlinearities appearing in the cost function only.

Constraints normally impose a pair of bounds (one lower, the other upper) on future (predicted) values of control inputs and/or outputs. Geometrically, each pair of constraints defines a region between two hyperplanes. If the intersection of such regions is nonempty, then the problem is said to be

feasible, and the aim of the predictive controller is to select the point in the feasible region, which minimizes a certain cost function.

According to *[Soeterboek, 1991]*, the constraints can be classified as:

- Soft constraints – can be taken into account by the minimization of criterion (4.2) and violations of them are temporarily allowed *[Soeterboek, 1991]*. In many cases, this situation is considered as an usual unconstrained optimization problem. The soft constraints can be incorporated into the optimization problem. An analytical solution to the optimal problem is available (as in Chapter 4).

- Hard constraints – cannot be handled by the minimization of criterion (4.2) in the way discussed in Chapter 4. No violation of these bounds are allowed at any time. When the hard constraints are present, the optimization problem is called the constrained optimization problem. An analytical solution is no longer available (as it is for the unconstrained problem) and therefore, an iterative optimization method should be used at each sample.

In practical situations, the most frequently encountered hard constraints are:

- Level constraints, where the controller output is limited between two values (the lower bound u^{min} and the upper bound u^{max}), i.e.:

$$u^{min} \leq u \leq u^{max}$$

- Rate constraints, where the change of the controller output per sample T_s is limited between two values (the lower bound Δu^{min} and the upper bound Δu^{max}), i.e.:

$$\Delta u^{min} \quad \leq \quad \frac{u(t) - u(t-1)}{T_s} \quad \leq \quad \Delta u^{max}$$

These bounds can be time–varying. Normally the limits are imposed by the actuator and therefore they can be assumed to be constant. This type of hard constraints which will be considered in this chapter.

It is known that in conventional controllers such constraints are difficult to handle in a systematic way, because of their nonlinear nature. The most commonly used way to take such constraints into account, is the ad-hoc procedure during the implementation of the controller: the calculated control signal is simply truncated to the limits. This solution may badly influence the performance of the closed loop system *[Soeterboek, 1991]*. As it will be seen in the simulations, the system can become unstable due to an ad-hoc way of handling the constraints. It is known that a possible way to avoid these problems is to operate far away from the physical limitations of the process. But this solution is usually not optimal from an economical point of view. Therefore the ability to cope with constraints may be an important robustness property of a controller. It will be shown that, the predictive controllers can manage the hard constraints in a natural and systematic way and therefore, the ad-hoc strategy can be avoided. By predicting the process output and taking the effect of the constraints on the predicted process output into account, the controller can anticipate the constraints. Thus, the predictive design concept allows us to operate safely, closer to the physical limitations of the process.

Let us assume that the plant is adequately modelled by structure (4.1), namely by:

$$A(q^{-1})y(t) = q^{-d}\frac{B(q^{-1})}{A_1(q^{-1})}u(t) + \frac{C(q^{-1})}{A_2(q^{-1})}e(t) \tag{5.1}$$

Model (5.1) could e.g. be the result of a model reduction strategy (see Chapter 3).

The optimization problem in the presence of constraints on the predictive controller can be formulated as the minimization of the extended cost function[1]:

$$J = \sum_{i=H_m}^{H_p} [\frac{P_n}{P_d}\hat{y}(t+i) - w(t+i)]^2 + \rho \sum_{i=0}^{H_p-d} [\frac{Q_n}{Q_d}u(t+i)]^2 \tag{5.2}$$

with respect to the controller output sequence over the control horizon $\Big[u(t), \quad \cdots, \quad u(t+H_c-1)\Big]$:

$$\phi \, \tfrac{P_n}{P_d} \, u(t+i) = 0, \qquad H_c \leq i \leq H_p - d \tag{5.3}$$

and subject to the set of linear constraints on the future controller outputs:

$$\begin{aligned} u^{min}(t+i) \leq u(t+i) \leq u^{max}(t+i) \qquad & i = 0, \cdots, H_c - 1 \\ \Delta u^{min}(t+i) \leq \Delta u(t+i) \leq \Delta u^{max}(t+i) \qquad & i = 0, \cdots, H_c - 1 \end{aligned} \tag{5.4}$$

[1] The parameters of the criterion and of the constraints were defined in Chapter 4.

Constraints (5.3) are soft constraints. They can easily be incorporated in the minimization of (5.2), which means that the unconstrained optimization problem can be solved analytically. Constraints (5.4) are typically hard constraints. An analytical solution to the optimization problem is no longer available and therefore an iterative optimization method must be used.

5.2 Optimal searching strategy

In this section, the general problem formulation and some aspects related to the optimization strategy are presented.

The objective of the formulated constrained control problem is to minimize a quadratic cost function in the presence of a set of linear constraints:

$$\begin{array}{ll} \text{minimize :} & J(\underline{x}) \\ \text{subject to:} & \boldsymbol{A}\,\underline{x} \;\leq\; \underline{b} \end{array} \tag{5.5}$$

where $\boldsymbol{A}$ is a matrix of dimension $[\boldsymbol{A}] = n_c \times n_x$ and $\underline{b}$ is a vector of dimension $[\underline{b}] = n_c \times 1$. They are determined by the type of control constraints. The variable n_c denotes the number of constraints, while n_x denotes the number of variables for which the criterion is optimized.

Two fundamental concepts strictly related to the constrained optimization problem will be explained in the following:

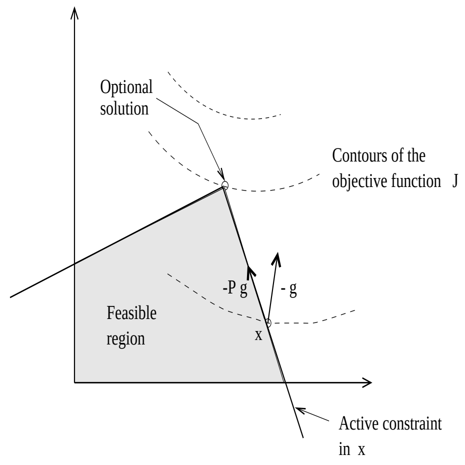

Figure 5.1. *Projection of the gradient.*

1. Feasible point.
   A point $\underline{x}$ is defined to be feasible, if $\underline{x}$ satisfies all the constraints:

   $$\boldsymbol{A}\,\underline{x} \;\leq\; \underline{b}$$

   The collection of all such points is known as the feasible region (see Figure 5.1). It is important the feasible region to be convex, i.e. any point on the line segment joining two feasible points to be also a feasible point *[Gill and Murray, 1974]*.

2. Active constraint.
   An inequality constraint $\underline{a}_i^T\,\underline{x} \leq b_i$ is defined as active or binding at a feasible point $\underline{x}_n$, if:

   $$\underline{a}_i^T\,\underline{x}_n \;=\; b_i$$

where $\underline{a}_i^T$ and b_i denote the i-th row of $\boldsymbol{A}$ and the i-th element of $\underline{b}$, respectively. The iteration number is denoted by n. The constraint is said to be inactive at $\underline{x}_n$, if $\underline{a}_i^T \underline{x}_n < b_i$, i.e. $\underline{x}_n$ is in the interior of the feasible region. Note that equality constraints are always active. If the set of active constraints at $\underline{x}_n$ is known, then the remaining inactive constraints can be (locally) ignored as they do not influence the neighborhood of $\underline{x}_n$ *[Bazaraa and Shetty, 1979]*. Therefore the problem can be treated as an equality constraint problem, i.e. the cost function is minimized with a subset of the constraints fixed as equalities. This fact forms the basis for a fundamental class of methods for all linear constrained problems known as "the active-set methods" *[Bazaraa and Shetty, 1979]*.

Most iterative nonlinear optimization strategies are based on the following algorithm:

1. At each iteration, for a given feasible point $\underline{x}_n$, a direction $\underline{d}_n$ and a step size $\lambda_{opt,n}$ are determined, such that:
   - A new feasible point $\underline{x}_{n+1}$ is obtained:

     $$\underline{x}_{n+1} = \underline{x}_n + \lambda_{opt,n}\, \underline{d}_n$$

   - $J(\underline{x}_{n+1}) \leq J(\underline{x}_n)$ is satisfied.
2. The process is repeated until the optimal solution is found.

Vector $\underline{x}_n$ has dimension $[\underline{x}_n] = n_x \times 1$ and it contains the variables for which the criterion $J(\underline{x})$ is minimized. The way $\underline{d}_n$ and $\lambda_{opt,n}$ are calculated, distinguishes the optimization algorithms.

It should be noted that the concept of a feasible region is strictly related to the constrained optimization problem. In the unconstrained problem, the steepest descent method finds the solution by moving along the negative gradient. In the presence of constraints, however, moving along the steepest descent direction may lead to infeasible points. Therefore, the constrained optimization problem should not only improve the cost function, but also maintain the feasibility of the iterative solution $\underline{x}_{n+1}$ at each step.

In the following, Rosen's gradient projection method is discussed and adapted to the constrained predictive control problem.

5.3 Rosen's gradient projection method

Rosen's gradient projection method *[Rosen, 1960]* is a general optimization method that can be used to minimize a cost function, not necessarily quadratic, in the presence of linear constraints (equalities or inequalities). It is motivated by the ordinary method of steepest descent for unconstrained problems and therefore it is also known as the gradient projection method.

The reason for selecting this method is that it turns into a simple and fast method, especially in the case of level constraints.

The search direction $\underline{d}_n$

In Rosen's gradient method, the search direction $\underline{d}_n$ is obtained by projecting the negative gradient $\underline{g}_n$ on all active constraints (see Figure 5.1).

Thus:

$$\underline{d}_n = -\boldsymbol{P}_n \ \underline{g}_n, \qquad [\underline{d}_n] = n_x \times 1 \tag{5.6}$$

where the gradient $\underline{g}_n$ of criterion J with respect to $\underline{x}_n$ is given by:

$$\underline{g}_n = \frac{\partial J}{\partial \underline{x}_n}$$

The computation of the gradient $\underline{g}_n$ is strongly dependent on the cost function J.

In *[Luenberger, 1984]*, the projection matrix $\boldsymbol{P}_n$ is defined as:

$$\boldsymbol{P}_n = \boldsymbol{I} - \boldsymbol{K}_n^T(\boldsymbol{K}_n\boldsymbol{K}_n^T)^{-1}\boldsymbol{K}_n, \qquad [\boldsymbol{P}] = n_x \times n_x \tag{5.7}$$

where $\boldsymbol{I}$ is the unity matrix, $[\boldsymbol{I}] = n_x \times n_x$ and $\boldsymbol{K}_n$ is a matrix which contains all the active constraints for the point $\underline{x}_n$:

$$\boldsymbol{K}_n\underline{x}_n = \underline{b}^* \tag{5.8}$$

$\boldsymbol{K}_n$ has dimension $[\boldsymbol{K}_n] = r \times n_x$. The rows in $\boldsymbol{K}_n$ contain the coefficients of the active constraints. It should be mentioned that the number of active constraints r changes during the optimization procedure. Vector b^* contains the values of the bounds corresponding to the active constraints ($[b^*] = r \times 1$).

If the point $\underline{x}_n$ has no active constraints (hence, $r = 0$), then the projection matrix is chosen as $\boldsymbol{P}_n \equiv \boldsymbol{I}$, which means that the algorithm turns into the method of steepest descent, i.e. $\underline{d}_n = -\underline{g}$.

The step size $\lambda_{opt,n}$

Once an improving feasible direction $\underline{d}_n$ is generated, the next step is to determine the positive step size $\lambda_{opt,n}$.

In unconstrained optimization it is known that, finding the optimal step $\lambda^*_{opt,n}$ in the search direction $\underline{d}_n$ is an one-dimensional minimization problem, and the solution (see Appendix F) is given by:

$$\lambda^*_{opt,n} = -\frac{\underline{g}_n^T \underline{d}_n}{\underline{d}_n^T \boldsymbol{H} \ \underline{d}_n} \tag{5.9}$$

where $\boldsymbol{H}$ is the Hessian matrix of the criterion function J with respect to $\underline{x}_n$. It is assumed that the Hessian matrix is positive definite[1].

In the presence of constraints, however, moving in the search direction with $\lambda^*_{opt,n}$, according to (5.9), may lead to infeasible points. Therefore, in order to find a proper optimal step size $\lambda_{opt,n}$ in the constrained optimization method, one should take into account that at each sample, the new point:

$$\underline{x}_{n+1} = \underline{x}_n + \lambda_{opt,n} \ \underline{d}_n$$

[1] The case when $\boldsymbol{H}$ is semi-positive definite is not a well posed control problem.

must be feasible, i.e.:

$$\underline{a}_j^T\,\underline{x}_n + \lambda_{opt,n}\,\underline{a}_j^T\,\underline{d}_n \leq b_j$$

This inequality implies that:

$$\lambda_{opt,n} \quad \leq \quad \lambda_{max} \tag{5.10}$$

where:

$$\lambda_{max} = \begin{cases} \min\limits_j \left(\frac{b_j - \underline{a}_j^T\,\underline{x}_n}{\underline{a}_j^T\,\underline{d}_n} \right) & \text{if } \underline{a}_j^T\,\underline{d}_n > 0 \\ \\ \infty & \text{if } \underline{a}_j^T\,\underline{d}_n \leq 0 \end{cases} \tag{5.11}$$

Thus, in the constrained optimization, the step size is limited by λ_{max}, which represents the shortest distance to the inactive constraints j, along the search direction $\underline{d}_n$.

According to (5.9) and (5.10), the optimal step for the constrained optimization problem finally can be computed as:

$$\lambda_{opt,n} = \min(\lambda_{max}, \lambda^*_{opt,n}) \tag{5.12}$$

In the case when $\lambda_{opt,n} = \lambda_{max}$, the new point turns one or more inactive constraints into active constraints. Thus, in the next step of the algorithm, the new active constraints should be taken into account. Consequently, the active constraint matrix $\boldsymbol{K}_n$ and the projection matrix $\boldsymbol{P}_n$ change.

The stop criterion

In unconstrained minimization, the necessary conditions for an optimal solution $\underline{x}_n$ illustrate the requirement for zero slope and positive curvature in any direction at $\underline{x}_n$, that is to say that, there is no descent direction at $\underline{x}_n$. In the constrained optimization, an additional complication is given by the feasible region concept. There it should be checked that the optimal point $\underline{x}_n$ is: 1) feasible and that 2) in $\underline{x}_n$ other feasible descent directions do not exist. The optimal solution is found when all the conditions are satisfied, in which case, the iterative algorithm can be stopped.

Rosen's gradient projection method is stopped when the following two conditions are satisfied:

1. Firstly, the search direction $\underline{d}_n$ should be equal to zero. In practice, the following numerical form of the condition is used:

$$\max_j \; |d_j| < \eta \qquad \text{for} \;\; j = 1, \cdots, n_x \tag{5.13}$$

   where d_j is the j-th element of $\underline{d}_n$ and η is a small positive value which expresses the accuracy of the solution.

2. The Kuhn-Tucker conditions (which correspond to the second derivative in the unconstrained optimization problem) should be satisfied, i.e. all the elements of the Kuhn-Tucker vector:

$$\underline{n}_n = -(\boldsymbol{K}_n \boldsymbol{K}_n^T)^{-1} \boldsymbol{K}_n \underline{g}_n \geq \underline{0} \tag{5.14}$$

   should be positive *[Soeterboek, 1991]*.

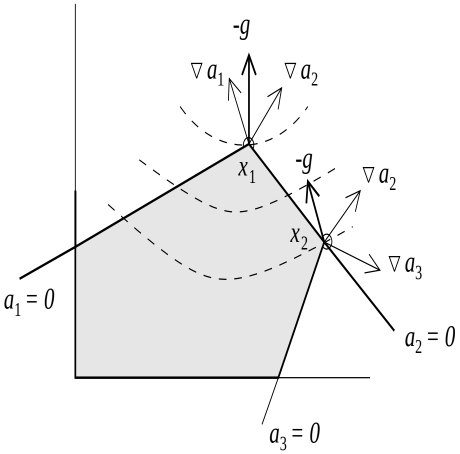

Figure 5.2. *Geometric illustration of the Kuhn-Tucker conditions.*

If there are no active constraints in $\underline{x}_n$, the Kuhn-Tucker vector is chosen as: $\underline{n}_n = \underline{0}$. If the conditions (5.14) are fulfilled, $\underline{x}_n$ is called a Kuhn-Tucker point. The Kuhn-Tucker conditions can be interpreted as such that the negative gradient should be inside the cone spanned by the gradients of the binding constraints *[Bazaraa and Shetty, 1979]*. This is illustrated in Figure 5.2 for two points: x_1 and x_2. For the sake of simplicity in the figure, the i-th equation constraint is denoted by $a_i = 0$. Three constraints delimit the feasible region. Note that for x_1, the gradient $-\underline{g}$ belongs to the cone spanned by the gradients of the binding constraints $a_1 = 0$ and $a_2 = 0$, in which case all the elements of $\underline{n}_n$ are positive and thus $\underline{x}_1$ is a Kuhn-Tucker point. On the other hand in x_2, $-\underline{g}$ lies outside the cone spanned by the gradients of the binding constraints $a_2 = 0$ and $a_3 = 0$, and therefore x_2 is not a Kuhn-Tucker point.

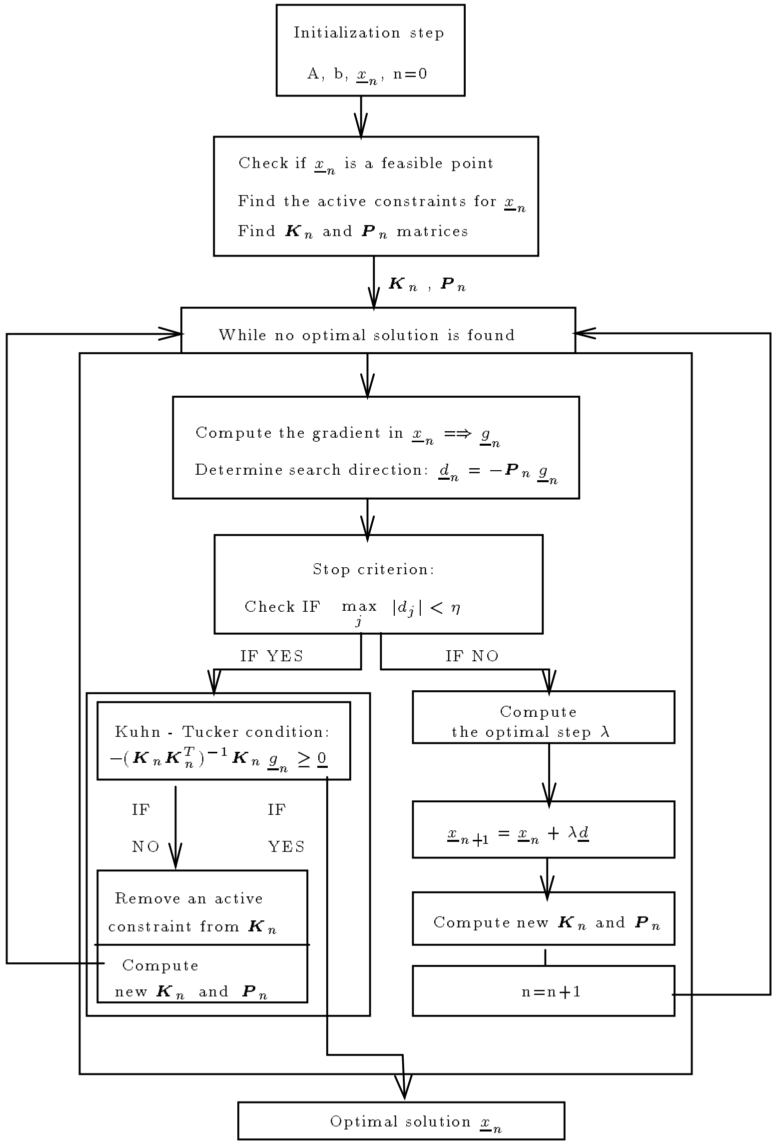

Figure 5.3. *Rosen's Gradient Projection Algorithm.*

However, when at least one of the elements of $\underline{n}_n$ in (5.14) is negative, the corresponding constraint should be relaxed, i.e. removed from the matrix of active constraints $\boldsymbol{K}_n$. Usually, the constraint corresponding to the most negative element of vector $\underline{n}_n$ is removed *[Soeterboek, 1991]*. Thus, a new direction vector $\underline{d}_n$ to an improved point, is determined by projecting the gradient onto the subspace determined by the remaining active constraints. The iterative search continues until both (5.13) and (5.14) are fulfilled, when the optimal solution is found. This optimal solution is also the global solution (thus the unique solution) if the Hessian matrix is positive definite. The positiveness of the Hessian matrix can be ensured e.g. in predictive control by a proper choice of the design parameters (see page 114). Rosen's gradient projection algorithm is sketched in Figure 5.3.

The complexity of the algorithm can be illustrated by the number of FLOPs (cumulative number of floating point operations – see Matlab) required to perform one iteration. According to *[Soeterboek, 1991]*, it depends on the number of active constraints r.

5.4 Calculation of the gradient and the implementation

The adaptation of Rosen's gradient projection method to the constrained control problem formulated in Section 5.1, is presented in this section. In the optimization algorithm, the most complicated calculation that is required

at each iteration, is the calculation of the gradient:

$$\underline{g}_n = \frac{\partial J}{\partial \underline{x}_n} \tag{5.15}$$

As the criterion function (5.2) must be minimized with respect to the controller output sequence over the control horizon H_c, the vector $\underline{x}_n$ from Rosen's method is now represented by:

$$\underline{x}_n = \underline{u}_n = \left[\ u(t), \quad \cdots \quad , u(t+H_c-1)\ \right]$$

where, throughout this chapter, $\underline{u}_n$ denotes the vector $\underline{u}_1$, defined on page 103; while n is the iteration number. The gradient (5.15) is expressed in (4.33). This expression of the gradient can be rewritten as:

$$\underline{g}_n = \boldsymbol{H} \cdot \underline{u}_n + \underline{\mathcal{K}} \tag{5.16}$$

where $\boldsymbol{H}$ is the Hessian matrix given by:

$$\boldsymbol{H} = 2\boldsymbol{M}^T(\boldsymbol{G}^T\boldsymbol{G} + \rho\boldsymbol{\Phi}^T\boldsymbol{\Phi})\boldsymbol{M} \tag{5.17}$$

and $\underline{\mathcal{K}}$ is a vector given by:

$$\begin{aligned} \underline{\mathcal{K}} \ &= \ 2\boldsymbol{M}^T[\boldsymbol{G}^T\ (\boldsymbol{L}\underline{\tilde{u}}_1 + \boldsymbol{F}\underline{\tilde{y}} - \underline{w} + \boldsymbol{G}\boldsymbol{N}\underline{\tilde{u}}_2) \\ &+ \ \rho\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{N}\underline{\tilde{u}}_2 + \boldsymbol{\Omega}\underline{\tilde{u}}_3)] \end{aligned} \tag{5.18}$$

It should be noticed that all the elements of (5.17) and (5.18) were already defined in Chapter 4. By a proper choice of design parameters in the controller (see page 114), the positive definiteness of the Hessian matrix $\boldsymbol{H}$ and thus, a unique solution of the optimization problem, is assured.

The expressions (5.16), (5.17) and (5.18) show that only the vector $\underline{u}_n$ depends on the iteration number n. The Hessian matrix $\boldsymbol{H}$ and the vector $\underline{\mathcal{K}}$ do not depend on n and hence, they do not need to be calculated at each iteration.

The algorithm for embedding Rosen's gradient projection method in a control scheme *[Soeterboek, 1991]*, i.e. to calculate the control output sequence over the control horizon at each iteration, is known as Dedicated Gradient Projection (DGP) algorithm. The DGP algorithm is sketched in Figure 5.4. In the initialization step, all the invariant matrices, vectors and variables can be calculated. In case the model is not estimated on line, the matrices used in the design of the controller ($\boldsymbol{G}$, $\boldsymbol{F}$, $\boldsymbol{L}$, $\boldsymbol{M}$, $\boldsymbol{N}$, $\boldsymbol{\Phi}$, $\boldsymbol{\Omega}$), the Hessian matrix $\boldsymbol{H}$ and its inverse do not need to be calculated at every iteration. At each sample, once the output of the plant is measured and the vector $\underline{\mathcal{K}}$ is computed, the unconstrained optimization solution is given by:

$$\underline{u}_n^{unc} = -\boldsymbol{H}^{-1}\underline{\mathcal{K}} \tag{5.19}$$

It should be remember that the Hessian matrix $\boldsymbol{H}$ has dimension $H_c \times H_c$. Hence, a small control horizon is preferable in order to save computation time and avoid possible numerical problems in the inversion of $\boldsymbol{H}$.

Each element of the sequence $\underline{u}_n^{unc}$ is checked if it satisfies the imposed hard constraints (5.4). In the affirmative case, it is enough to choose the first

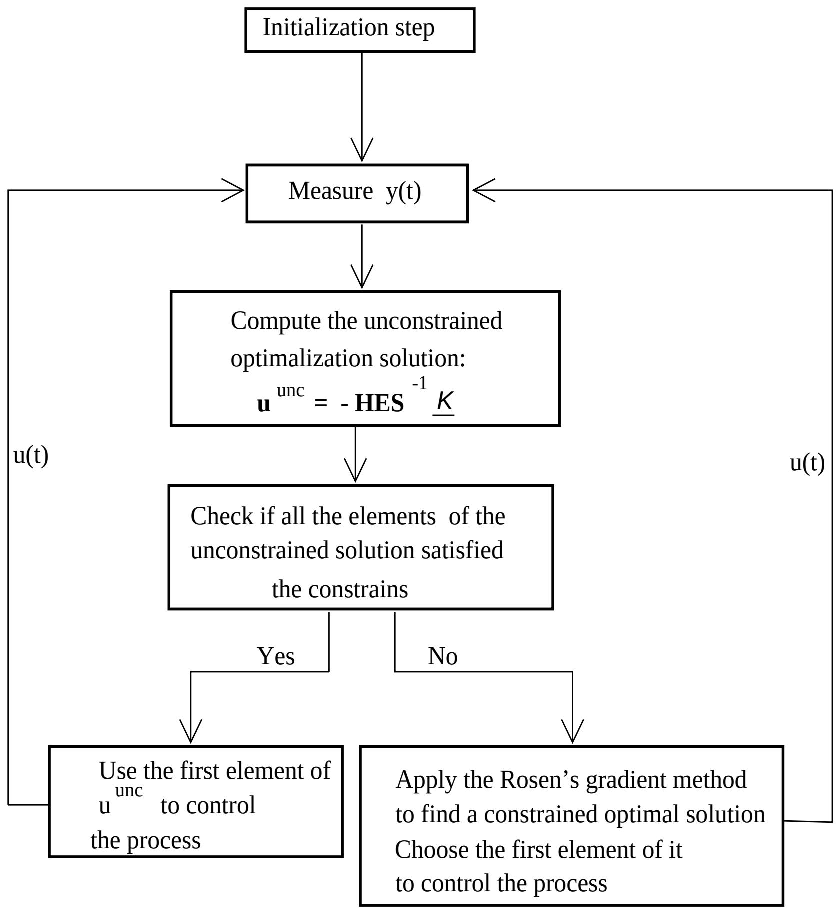

Figure 5.4. *DGP algorithm – computes the optimal constrained control signal at each sample.*

element of this sequence to control the plant. Otherwise, if there exists at least one element which violates the feasibility range, defined by the constraints, then Rosen's gradient projection algorithm is started in order to find an optimal constrained solution. Then again, the first element is picked out from the new control sequence obtained, using Rosen's gradient projection algorithm. The process is thus repeated, by applying the new control input to the plant and by measuring a new value of the output $y(t)$. It should be noted that the vector $\underline{\mathcal{K}}$ and the filtered signals $\underline{\tilde{u}}_1$, $\underline{\tilde{u}}_2$, $\underline{\tilde{u}}_3$, $\underline{y}_1$ should each sample be computed.

Initialization of the algorithm

The starting point of the optimization algorithm has a major effect on the number of iterations, required to find the optimal solution with particular accuracy η. If the desired accuracy is sufficiently small, the starting point hardly influences the solution that is found, numerically problems can appear, and the computation time increases *[Soeterboek, 1991]*.

In general, the closer the starting point is to the optimum, the faster the optimum with accuracy η is found. The only condition which is required, is that the initial point should be feasible. Three different possible ways to initialize $\underline{u}_0$ are listed below:

1. A fixed feasible point $\underline{u}_0$ can be used to define the initialization point. In linear constraint control problems, it is relatively easy to find a feasible initial point. However, in the nonlinear case, it is more difficult, but the method of finding a feasible point, known as "*phase I simplex*" (described in *[Gill and Murray, 1974]*) can be used.
2. Choose as initialization point in the on-line procedure, the optimal sequence obtained in the preceding sample, with a time shift and e.g. repeat the last element. Thus:
   - if at the moment t the optimal solution is:

     $$\underline{u} = \begin{bmatrix} u(t) & u(t+1) & \cdots & u(t+H_c-1) \end{bmatrix}^T$$

   - then at moment $t+1$ the initialization point can be chosen as:

     $$\underline{u}_0 = \begin{bmatrix} u(t+1) & \cdots & u(t+H_c-1) & u(t+H_c-1) \end{bmatrix}^T$$

3. Use an unconstrained solution and make it initial feasible point by clipping it to the limits by the constraints. For example, in the case of level constraints, one can choose the initial point u_0 to be:

   $$\underline{u}_0 = \min(\,\max(\,\underline{u},\,\underline{u}^{min}),\,\underline{u}^{max})$$

It will be shown latter that the initialization of the algorithm is strongly dependent on the types of constrained control problem.

In the following sections, Rosen's gradient projection method is adapted to some special control problems. The level and rate constraints on the predictive input control signal are discussed both separately and together.

5.5 Level constraints

In case of level constraints, the optimization problem consists in the minimization of cost function (5.2) with respect to the vector:

$$\underline{u}_n = \begin{bmatrix} u(t), & \cdots, & u(t+H_c-1) \end{bmatrix}$$

and subject to (5.3) and to the set of linear constraints:

$$\begin{aligned} u(t+j-1) &\leq u^{max}(t+j-1) \\ -u(t+j-1) &\leq -u^{min}(t+j-1) \end{aligned} \tag{5.20}$$

where $j = 1, \ldots, H_c$. Expression (5.20) typically expresses an actuator saturation. Note that in this case, the number of the constraints, which should be taken explicitly into account, is $2\,H_c$. In order to apply Rosen's gradient projection algorithm, the constraints should be written as in (5.5). Thus, according to (5.20), the structure of $\boldsymbol{A}$ and $\underline{b}$ is as follows:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & 1 \\ \hline -1 & 0 & \ldots & 0 \\ 0 & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & -1 \end{bmatrix}, \quad \underline{b} = \begin{bmatrix} u^{max} \\ \vdots \\ u^{max} \\ \hline -u^{min} \\ \vdots \\ -u^{min} \end{bmatrix} \tag{5.21}$$

For convenience, the constraints are considered time invariant.

Implementation aspects

The dimension of $\boldsymbol{A}$ is $[\boldsymbol{A}] = 2\,H_c \times H_c$, while for $\underline{b}$ it is $[\underline{b}] = 2\,H_c \times 1$. It should be observed that due to the fact that $u(t+j-1)$ is limited by either u^{min} or u^{max} and the form of the constraints is very simple, the constraints j and $j+H_c$ can not be active simultaneously. Hence, the maximum number of active constraints is equal to the control horizon H_c. This observation actually yields some simplifications in the computation of the optimal solution, because in Rosen algorithm, the projection matrix, the search direction and Kuhn–Tucker vector can easily be calculated *[Soeterboek, 1991]*. Thus, the projection matrix $\boldsymbol{P}_n$ becomes a diagonal matrix with the diagonal elements corresponding to the active constraints equal to zero, and the remaining diagonal elements equal to one. In the search direction $\underline{d}_n$, the elements corresponding to the active constraints are zero, and the elements, corresponding to the inactive constrains, are equal to the negative gradient.

According to *[Soeterboek, 1991]* the number of FLOPs required to perform one iteration of the optimization algorithm depends on the number of active constraints, and in the level constraints case, for the worst case i.e. $r = H_c$, it is $4H_c^2 + 6H_c + 1$.

Influence of the optimal constrained control

As mentioned in Section 5.1, the most commonly used way to take some constraints into account, is the ad-hoc strategy during the implementation of the controller. The control signal is simply truncated to the limits, in case it is outside the feasible region:

$$\underline{u} = \min(\,\max(\,\underline{u},\,\underline{u}^{min}),\,\underline{u}^{max}) \tag{5.22}$$

This way of controlling a process often gives unacceptable performance. In the following example, it will be shown that the closed loop performance of the system can be improved by taking the constraints explicitly into account.

Example 5.1 *A high order stochastic model of the Mariner Class Vessel is considered. The data of the ship are summarized in Appendix A. As external disturbances on the ship, wind and waves are considered. Their direction is* 45 *deg and the mean wind speed is* $\overline{V} = 15$ *m/s. The high order model, based on the expressions (2.11), (2.25), (2.39) outlined in Chapter 2, is linearized in a certain stationary point.*

A reduced order model is used in the design of the controller. This model is obtained by the BTA method, described in Chapter 3. Then the model, sampled with a sampling period of $T_s = 12$ *seconds, has the following description:*

$$A(q^{-1})\psi(t) = B(q^{-1})u(t) + C(q^{-1})e(t)$$

where:

$$\begin{aligned} A(q^{-1}) &= 1 - 1.8609\, q^{-1} + 0.8609\, q^{-2} \\ B(q^{-1}) &= -0.1425\, q^{-1} - 0.1356\, q^{-2} \\ C(q^{-1}) &= 1 - 1.2535\, q^{-1} + 0.3459\, q^{-2} \end{aligned} \tag{5.23}$$

$e(t)$ *denotes mean zero white noise with variance 1. This design model will be used extensively throughout the rest of this chapter. In order to be able*

to compare the results, the same noise sequence is used in all simulations. The design parameters of the controller are chosen as:

$$\begin{array}{ll} H_m = 1 & P_n = P_d = 1 \\ H_c = 2 & Q_n = Q_d = 1 \\ H_p = 3 & A_2 = 1 - q^{-1} \\ \rho = 0 & \phi = 1 - q^{-1} \end{array} \tag{5.24}$$

The restrictions on the input signal are:

$$u^{min} = -30 \ [deg] \quad and \quad u^{max} = 30 \ [deg] \tag{5.25}$$

A step response simulation using a constrained predictive controller is shown in Figure 5.5. The optimization algorithm is initialized at a fixed value $\underline{u}_0 = \underline{0}$. *The accuracy of finding the optimal solution is* $\eta = 10^{-6}$. *In the unconstrained case, the constraints are especially violated in the transient period, when a large input signal is required. The performance of the closed loop system is judged by the following loss functions:*

$$\begin{aligned} J_y &= (\underline{y} - \underline{w})^T(\underline{y} - \underline{w}) \\ J_u &= \underline{u}^T \underline{u} \end{aligned} \tag{5.26}$$

where $\underline{u}$, $\underline{y}$, $\underline{w}$ *are the input, output and reference signal, respectively. In case the ad-hoc limitation on the input amplitude is attempted, the output and input loss functions are* $J_y = 26.5642$ *and* $J_u = 8.8547 \cdot 10^3$, *respectively. In the optimal constrained control these loss functions are* $J_y = 26.4327$ *and* $J_u = 7.5378 \cdot 10^3$. *It is observed that in the optimal constrained control* J_u

is much smaller than that in ad-hoc control strategy, while the output loss function is almost the same. ◇

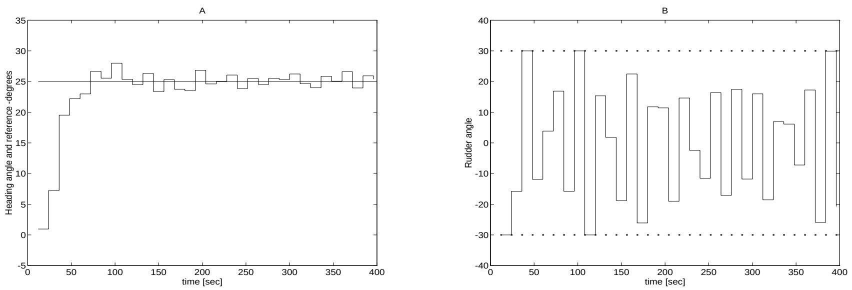

Figure 5.5. *Optimal level constrained control.*

Influence of η

The influence of the parameter η, defined in (5.13), on the performance of the closed loop system can be analyzed by performing a number of simulations using different η. It is observed that using a smaller value of the accuracy η, in finding the optimal solution, does not necessarily provide a better performance. Instead it implies an increased number of iterations for finding the optimal solution and it can be a source for numerical problems.

Example 5.2 *To illustrate the effect of η, the simulation Example 5.1 is considered. The design parameters controller are:*

$$H_m = 1, \quad H_c = 1, \quad H_p = 1, \quad \rho = 0$$

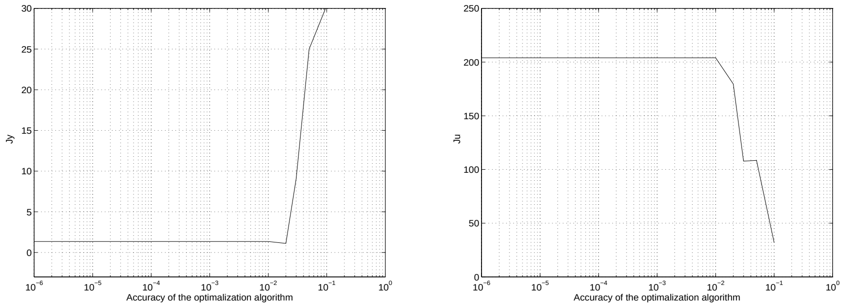

Figure 5.6. J_y and J_u *as function of* η.

and the following limits on the control signal are considered:

$$u^{min} = -20 \ [deg] \ \textit{and} \ u^{max} = 20 \ [deg]$$

The reference is set at 25 deg. The output J_y and input J_u loss functions, as function of η are shown in Figure 5.6.

Clearly, for the considered model and controller, using $\eta \leq 10^{-2}$ yields solutions close to the optimal one (obtained for $\eta \longrightarrow 0$). Hence, using $\eta = 10^{-2}$ results in a good performance with a relatively small number of iterations. Using a smaller value for η does not provide a better performance, but it can cause an increased number of iterations. Using a larger value of η results in solutions far from the optimal one, case when the output of the system is not able to follow the reference in an acceptable way. This aspect is illustrated in Figure 5.7, where two cases are considered: 1) for the maximum acceptable value of the accuracy ($\eta = 0.01$) and 2) for a value of the accuracy which is not acceptable ($\eta = 0.05$). According to the results in Figure 5.6, for the considered model and controller, the variance of the output error increases

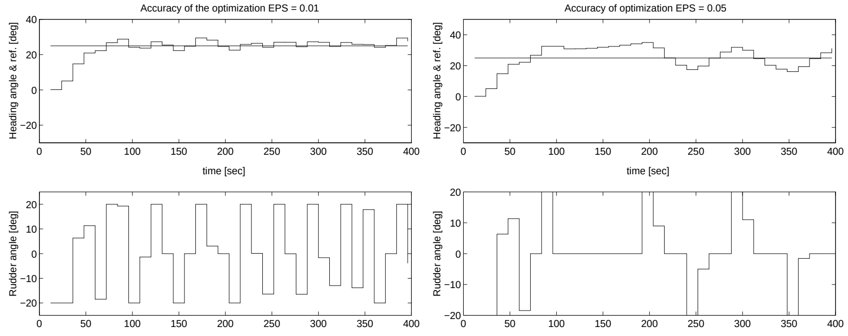

Figure 5.7. *The influence of η in the responses.*

while the variance of the control signal decreases when the accuracy η in (5.13) is bigger than 0.01. ◇

Example 5.3 *In order to get an idea on the convergence speed of the constrained optimization algorithm, the number of iterations required in every sample, as a function of accuracy η, can be examined.*

Thus Figure 5.8 shows how many iterations are required by the DGP algorithm at each sample to find the optimal solution for accuracy $\eta = 10^{-2}$ (part a) and $\eta = 10^{-3}$ (part b), respectively. A step response simulation is done for the design parameters:

$$H_m = 1, \quad H_c = 3, \quad H_p = 3, \quad \rho = 0$$

and a fix initialization of the optimization algorithm. It is observed that the number of iterations with $\eta = 10^{-2}$ is significantly smaller. Note that

in some samples, the number of iterations can be equal to zero. This is caused by the fact that the solution of the unconstrained problem does not violate any of the constraints, and therefore the unconstrained solution can be directly used to control the process. ◇

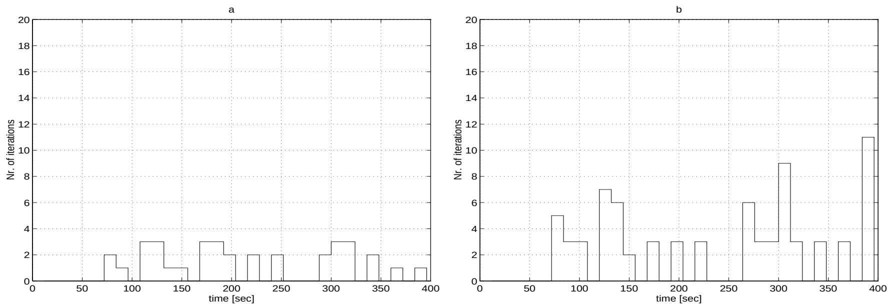

Figure 5.8. *Number of iterations required by the DGP algorithm to find the optimal solution with accuracy $\eta = 10^{-2}$ (part a) and $\eta = 10^{-3}$ (part b) when the level constraints are taken into account and u_0 is fixed.*

Usually, it is difficult to predict the effect of the design parameters of the predictive controller on the response of the closed loop system. This fact is due to the complicated form of the closed loop system and of the predictive control low (4.35). However, in the following the influence of some parameters is analysed.

Influence of the prediction horizon H_p

The influence of the prediction horizon will now be analyzed. The effect of changing the prediction horizon seems to be easier to predict.

By predicting the process output over a larger horizon, the effect of the constraints can better be taken into account. Therefore, usually, it can be expected that using a larger prediction horizon causes the performance of the closed loop system to improve.

H_p	Method	J_u	J_y	n_R
2	Ad hoc	$1.7431 \cdot 10^4$	44.8354	—
	DGP	$1.4732 \cdot 10^4$	39.7909	20
3	Ad hoc	$8.8547 \cdot 10^3$	26.5642	—
	DGP	$7.5378 \cdot 10^3$	26.4327	5
4	Ad hoc	$2.9774 \cdot 10^3$	24.3581	—
	DGP	$2.9774 \cdot 10^3$	24.3581	2
5	Ad hoc	$1.7740 \cdot 10^3$	23.0632	—
	DGP	$1.7740 \cdot 10^3$	23.0632	1

Table 5.1. *Performance criteria as a function of H_p in case of level constraints.*

Example 5.4 *Some simulations are now performed with $H_m = 1$, $H_c = 2$, $\eta = 10^{-3}$ kept constant, while H_p is varying. The design model and the other parameters shown in Example 5.1 are used. Table 5.1 illustrates the loss functions, defined on page 164, as a function of the prediction horizons. The number n_R of times when the Rosen's gradient projection algorithm is activated, in DGP scheme, is also shown. It is observed that a larger prediction horizon causes smaller input and output loss functions, that DGP method provides better results. Thus the performance of the closed loop system is improved, by using a larger H_p. It is also observed that, by increasing H_p, the number n_R of times Rosen's gradient projection algorithm is activated, in DGP scheme, is decreasing. This means that, when the prediction*

horizon H_p is increased, the number of situations where the constraints are violated in the unconstrained case decreases, and thus finally, the effect of the DGP algorithm turns out to be the same as in the ad-hoc strategy. $\diamond$

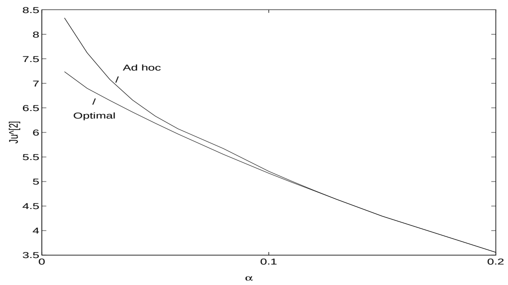

Figure 5.9. *The effect of α.*

Influence of the α

Another parameter, whose influence is interesting, is α in:

$$P_n = 1 - \alpha q^{-1} \quad \text{and} \quad P_d = 1 - \alpha$$

The simulations are done for the design parameters of the controller:

$$H_m = 1, \quad H_c = 2, \quad H_p = 3, \quad \rho = 0$$

and a fix initialization of the optimization algorithm. Parameter α is varying.

It is observed that when α is increased, the effect of DGP algorithm is the same with the effect of the ad-hoc strategy. This aspect is illustrated in Figure 5.9. The reason is that, by increasing α the system becomes slower, and therefore the unconstrained solution does not violate the constraints that much.

Influence of the ρ

The control signal weighting factor ρ is quite difficult to determine, and therefore it is typically chosen small. Usually ρ is used as a tool to decrease the controller output variance. It is observed in Table 5.2, that increasing ρ makes the controller less active: J_u and J_y become smaller. The simulations are done with the same conditions presented in Example 5.1. Notice again that DGP implies better performance than ad-hoc strategy.

It should also be noted that the number of moments where DGP is activated, is again decreasing when ρ is increasing, due to the fact that the closed loop system becomes slower. For numerical reasons, it may be convenient to choose ρ as a small positive value. The robustness is increased but servo and regulator performance decrease. In the case of unstable systems: increasing ρ results in an unstable closed loop system, the poles of the system appear as closed loop poles for larger ρ *[Soeterboek, 1991]*.

Influence of the initialization in the level constraints

The starting point in the optimization algorithm has a major effect on the number of iterations required to find the optimal solution with a particular accuracy.

ρ	Method	J_u	J_y	n_R
0	Ad-hoc	$8.8547 \cdot 10^3$	26.5642	—
	DGP	$7.5378 \cdot 10^3$	26.4327	5
10^{-5}	Ad-hoc	$8.8271 \cdot 10^3$	26.5768	—
	DGP	$7.5090 \cdot 10^3$	26.4107	5
10^{-4}	Ad-hoc	$8.5206 \cdot 10^3$	26.4895	—
	DGP	$7.1998 \cdot 10^3$	26.2001	4
10^{-3}	Ad-hoc	$5.3812 \cdot 10^3$	27.5310	—
	DGP	$5.3759 \cdot 10^3$	26.3745	3
10^{-2}	Ad-hoc	925.0522	22.7649	—
	DGP	925.0522	22.7649	2
10^{-1}	Ad-hoc	224.4813	28.70	—
	DGP	224.4813	28.70	0

Table 5.2. *Influence of ρ in level constraints.*

Example 5.5 *Figure 5.10 shows the number of iterations that is required for every sample by the DGP algorithm, when different types of initializations (presented on page 159) are performed. The step response simulations are carried out with the controller design parameters:*

$$H_m = 1, \quad H_c = 3, \quad H_p = 3, \quad \rho = 0$$

and accuracy $\eta = 10^{-2}$. In order to be able to compare the results, the same noise sequence is used in all simulations.

In Figure 5.10, it is obvious that the number of iterations in case the starting point is generated using the solution calculated in the preceding sample

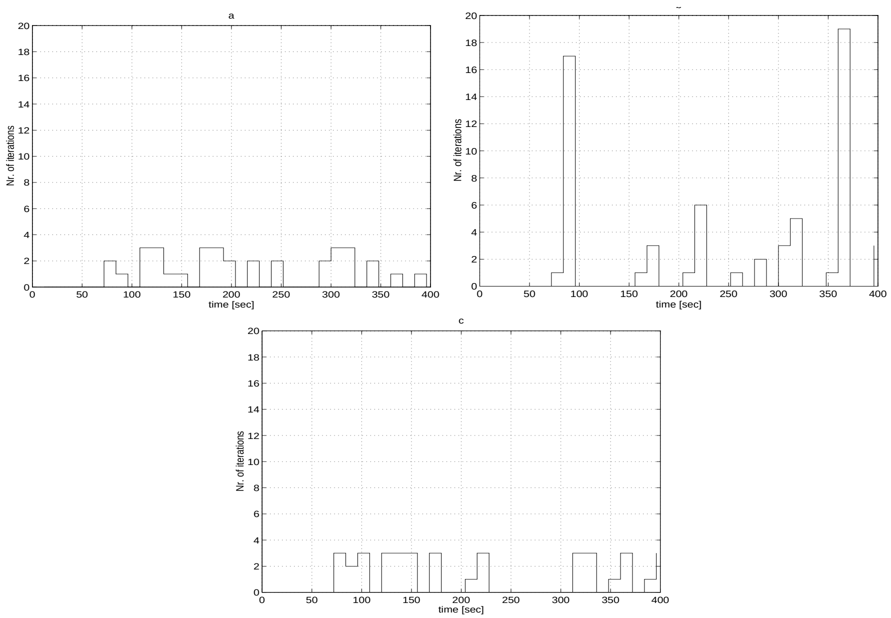

Figure 5.10. *Number of iterations required by the DGP algorithm for different types of initializations: part a – fixed initialization u_0; part b – initialization using the optimal solution in the preceding sample; part c – initialization by truncating the previous control signal.*

(part b), is much bigger than in the cases when a fixed initialization (part a) or a truncated control signal (part c) is used. If the area delimited by the plots is computed both for (part a) and (part c), it is observed that the fixed initialization requires the smallest number of situations when Rosen's optimization method is started. This fact implies a smaller computation time. $\diamond$

A heuristic method (Active Constraint Handling) to find an initialization point closer to the optimal solution of the constrained optimization problem, in the case of level constraints, is suggested in *[Soeterboek, 1991]*.

Summarizing remarks

In this section predictive control with level constraints was discussed. The following conclusions can be drawn based on simulation results:

- The performance of the closed-loop system in the presence of level constraints improves significantly when the hard constraints are taken systematically into account in the minimization of the cost function.

- A larger prediction horizon causes the performance of the closed loop system to be improved.

- By increasing the control signal weighting ρ and parameter α from $P_n = 1 - \alpha q^{-1}$, the control becomes less active, and therefore it can happen that no significant differences exist between ad-hoc and DGP strategy.

- Increasing H_p, ρ and α the number of moments where DGP is activated, is decreasing.

- Using a smaller value of the accuracy η, in finding the optimal solution, does not result in a better performance but causes the number of iterations to increase.

- The starting point in the optimization algorithm directly affects the number of iterations required in finding the optimal solution. For the considered model and controller, the fixed starting point yields a faster convergence.

5.6 Rate constraints

Rate constraints demand that the change of the control signal per sample T_s to be limited between two values:

$$\underline{\Delta u}^{min} \leq \underline{\Delta u} \leq \underline{\Delta u}^{max}$$

where at the time instant $t = kT_s$ (k positive integer), the rate of the control signal is defined as:

$$\Delta u(t) = \frac{u(t) - u(t-1)}{T_s}$$

As the rate constraints are imposed over all the control signals in the control horizon, namely $\left[\ u(t), \quad \cdots \quad , u(t+H_c-1)\ \right]$, the rate constraints can be

written according to (5.5), where:

$$\boldsymbol{A} = \left[\begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ -1 & 1 & \ddots & & 0 \\ 0 & -1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & 1 \\ \hline -1 & 0 & \cdots & 0 & 0 \\ 1 & -1 & \ddots & & 0 \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -1 \end{array}\right], \quad \underline{b} = \left[\begin{array}{c} T_s \cdot \Delta u^{max} + u(t-1) \\ T_s \cdot \Delta u^{max} \\ \vdots \\ T_s \cdot \Delta u^{max} \\ \hline -T_s \cdot \Delta u^{min} - u(t-1) \\ -T_s \cdot \Delta u^{min} \\ \vdots \\ -T_s \cdot \Delta u^{min} \end{array}\right] \tag{5.27}$$

Implementation aspects

Matrix $\boldsymbol{A}$ has dimension $[\boldsymbol{A}] = 2H_c \times H_c$, while vector $\underline{b}$ has dimension $[\underline{b}] = 2H_c \times 1$. Note that here vector $\underline{b}$ is a "dynamic" vector, i.e. some of its elements change at each sample, being dependent on the previous control signal $u(t-1)$.

In case of rate constraints, Rosen's gradient projection algorithm implies some extra calculations in comparison with the case of level constraints. The reason is that the previous simplifications in the computation of the projection matrix $\boldsymbol{P}_n$, search direction $\underline{d}_n$ and Kuhn–Tucker vector $\underline{n}_n$ do not exist anymore.

It should be mentioned that the number of FLOPs at each iteration only depends on the control horizon, not on the prediction horizon. The number of FLOPs in case of rate constraints is increased to $4H_c^2 + 17H_c - 6$ *[Soeterboek, 1991]*. However, the number of iterations and hence, the total calculation time (for a certain accuracy η in the search of the optimal solution) depends on H_p. A bigger H_p implies a greater computation time.

Influence of the optimal rate constraints control

The ad-hoc way to handle the rate constraints consists in truncating the control signal as follows:

$$\begin{aligned} u^*(k) \quad = \quad & \max[\min[u(k), \Delta u^{max}(k) \cdot T_s + u(k-1)], \\ & \Delta u^{min}(k) \cdot T_s + u(k-1)] \end{aligned} \tag{5.28}$$

The rate constraints can have a very deteriorating effect on the closed loop system in the case when the rate constraints are handled by the ad-hoc way. In particular, they can have a great influence on the stability of the closed loop system. As it will be seen in the following example, the closed loop system can even become unstable. This is a strong reason for having a controller design method which can handle constraints in a systematic way.

Example 5.6 *The influence of the way the rate constraints are handled, is analyzed in this example. The design parameters of the controller are chosen as in Example 5.1. Only constraints on the rate of the input signal are considered:*

$$\Delta u^{min} = -3.5 \ [deg/s], \quad \Delta u^{max} = 3.5 \ [deg/s]$$

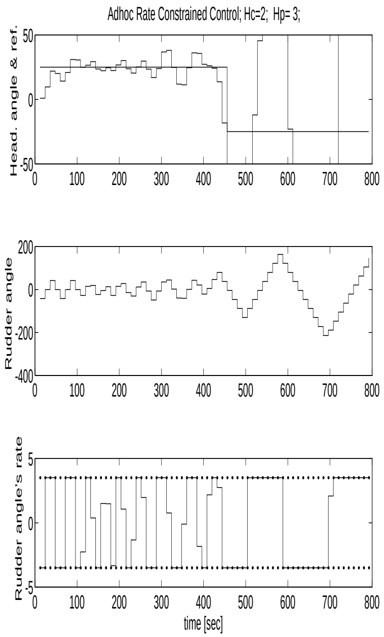

Figure 5.11. *Ad-hoc Strategy.*

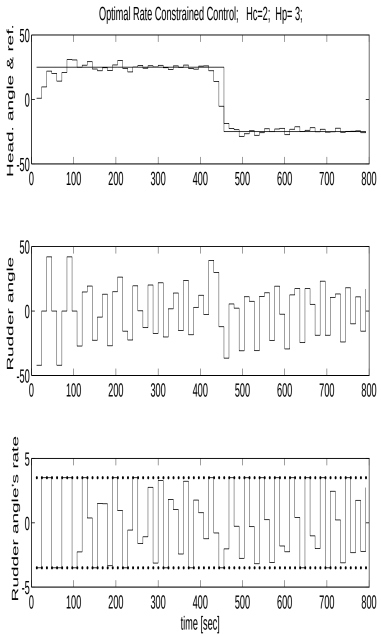

Figure 5.12. *Optimal Strategy.*

Figures 5.11 and 5.12 show the simulations for an ad-hoc and for an optimal rate constrained control, respectively. The accuracy used for finding the optimal solution is $\eta = 10^{-3}$. Figure 5.11 shows that for an ad-hoc strategy the closed loop system becomes unstable. It is however observed that if $H_p \longrightarrow 20$, as the system becomes very slow, its robustness towards instability is increased. When the DGP algorithm is used to calculate the optimal controller output sequence over the control horizon, the closed loop *system is stable, as it is shown by Figure 5.12. It is observed that the optimal control requires a big amplitude of the control signal, which in practice is not realistic. To manage this problem, both level and rate constraints in the design of the controller should be taken explicitly into account.* ◇

Influence of the prediction horizon H_p and ρ

As in the case of level constrained control, the performance of the closed loop in the presence of rate constrained control, is also improved by increasing the prediction horizon H_p.

H_p	Method	J_u	J_y	n_R
2	Ad-hoc	$2.6837 \cdot 10^6$	$4.9450 \cdot 10^8$	—
	DGP	$4.5146 \cdot 10^4$	$4.3998 \cdot 10^3$	59
3	Ad-hoc	$3.8397 \cdot 10^5$	$3.2677 \cdot 10^6$	—
	DGP	$2.0071 \cdot 10^4$	$1.2865 \cdot 10^3$	31
4	Ad-hoc	$1.3923 \cdot 10^4$	$4.3677 \cdot 10^3$	—
	DGP	$9.3058 \cdot 10^3$	$3.3719 \cdot 10^3$	8
5	Ad-hoc	$7.4451 \cdot 10^3$	$5.8038 \cdot 10^3$	—
	DGP	$6.7977 \cdot 10^3$	$5.4337 \cdot 10^3$	5
10	Ad-hoc	$5.1922 \cdot 10^3$	$1.6082 \cdot 10^4$	—
	DGP	$3.6866 \cdot 10^3$	$1.5828 \cdot 10^4$	2
20	Ad-hoc	$2.7988 \cdot 10^3$	$3.9204 \cdot 10^4$	—
	DGP	$2.7988 \cdot 10^3$	$3.9204 \cdot 10^4$	1

Table 5.3. *Performance criteria as a function of H_p in case of rate constraints.*

Example 5.7 *Some simulations are performed using all the parameter settings equal to those in the Example 5.1. The design parameters are kept constant, except H_p, during the different simulations. The results are shown in Table 5.3. A significant decrease in the loss functions J_u and J_y is observed. Again, when H_p increases, n_R decreases, and finally the performances of DGP and ad-hoc are the same.*

The influence of ρ is illustrated in Table 5.4. It is again confirmed that by increasing ρ, the control signal is more penalized and the number of moments where the DGP procedure is activated, decreases. ◇

ρ	Method	J_u	J_y	n_R
0	Ad-hoc	$3.8397 \cdot 10^5$	$3.2697 \cdot 10^6$	—
	DGP	$2.0071 \cdot 10^4$	$1.2865 \cdot 10^3$	31
10^{-5}	Ad-hoc	$7.3194 \cdot 10^4$	$1.7573 \cdot 10^4$	—
	DGP	$1.7277 \cdot 10^4$	$9.8131 \cdot 10^3$	24
10^{-4}	Ad-hoc	$4.1139 \cdot 10^5$	$5.6931 \cdot 10^6$	—
	DGP	$1.46 \cdot 10^4$	$1.2438 \cdot 10^3$	21
10^{-3}	Ad-hoc	$1.9313 \cdot 10^4$	$2.4367 \cdot 10^3$	—
	DGP	$1.7756 \cdot 10^4$	$1.9711 \cdot 10^3$	24
10^{-2}	Ad-hoc	$1.1691 \cdot 10^4$	$2.0186 \cdot 10^3$	—
	DGP	$5.5391 \cdot 10^3$	$1.471 \cdot 10^3$	7
10^{-1}	Ad-hoc	$3.9322 \cdot 10^3$	$1.4215 \cdot 10^3$	—
	DGP	$3.9322 \cdot 10^3$	$1.4215 \cdot 10^3$	1

Table 5.4. *Influence of ρ in the case of rate constraints ($H_p = 3$).*

Influence of the initialization in the rate constraints

Simulations have shown that in case of rate constraints, the initialization of the optimization algorithm requires some extra attention. Due to the fact that $\underline{b}$ is a "dynamic" vector here, depending on the previous signal $u(t-1)$, it is more difficult to control the feasibility of the starting point at every sample of the controller algorithm. By performing a set of simulations, it is noticed that only the second method, suggested on page 159, i.e. choosing the initial optimal solution from the previous sample, is robust to the infeasibility of the initial point. For the considered ship model, the other two methods can not keep $\underline{u}_0$ in the feasible region.

In case of rate constraints, neither the heuristic algorithm (ACH) proposed by *[Soeterboek, 1991]* can be used to calculate the initial control signal $\underline{u}_0$.

Summarizing remarks

In this section the rate constrained predictive control was discussed. The following conclusions can be drawn based on simulation results:

- Rate constraints, in particular, have a great influence on the stability of the closed loop system.
- The performance of the closed loop system, in the case of rate constraints are explicitly taken into account, is much better than with the ad-hoc strategy.

- Increasing the prediction horizon improves the performance of the closed loop system.
- Vector $\underline{b}$ is a "dynamic" vector, i.e. it changes in each sample.
- Increasing ρ makes the controller less active and the number of times Rosen's algorithm is activated n_R, decreases.
- The initialization of the algorithm needs some extra attention. The feasibility of the initial point can disappear. For the considered model, it was observed that using the optimal solution calculated from the preceding sample, the feasibility of the initial point was maintained.
- The computation time is bigger than in the case of level constraints.

5.7 Level and rate constraints

In this section it is assumed that both level and rate constraints exist on the controller output:

$$\begin{array}{rcr} \underline{u} & \leq & \underline{u}^{max} \\ -\underline{u} & \leq & -\underline{u}^{min} \\ \underline{\Delta u} & \leq & \underline{\Delta u}^{max} \\ -\underline{\Delta u} & \leq & -\underline{\Delta u}^{min} \end{array} \qquad (5.29)$$

The level and rate constrains (5.29) can be rewritten according to (5.5), when:

$$
\boldsymbol{A} = \left[\begin{array}{ccccc}
1 & 0 & \cdots & \cdots & 0 \\
0 & 1 & \ddots & & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & \ddots & 1 & 0 \\
0 & \cdots & \cdots & 0 & 1 \\
\hline
-1 & 0 & \cdots & \cdots & 0 \\
0 & -1 & \ddots & & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & \ddots & -1 & 0 \\
0 & \cdots & \cdots & 0 & -1 \\
\hline
1 & 0 & \cdots & \cdots & 0 \\
-1 & 1 & \ddots & & 0 \\
0 & -1 & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & 1 & 0 \\
0 & \cdots & 0 & -1 & 1 \\
\hline
-1 & 0 & \cdots & \cdots & 0 \\
1 & -1 & \ddots & & 0 \\
0 & 1 & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & -1 & 0 \\
0 & \cdots & 0 & 1 & -1
\end{array}\right], \quad
\underline{b} = \left[\begin{array}{c}
u^{max} \\
\vdots \\
u^{max} \\
\hline
-u^{min} \\
\vdots \\
-u^{min} \\
\hline
T_s \cdot \Delta u^{max} + u(t-1) \\
T_s \cdot \Delta u^{max} \\
\vdots \\
T_s \cdot \Delta u^{max} \\
\hline
-T_s \cdot \Delta u^{min} - u(t-1) \\
-T_s \cdot \Delta u^{min} \\
\vdots \\
-T_s \cdot \Delta u^{min}
\end{array}\right] \tag{5.30}
$$

In this case, matrix $\boldsymbol{A}$ has dimension $[\boldsymbol{A}] = 4H_c \times H_c$, while vector $\underline{b}$ has dimension $[\underline{b}] = 4H_c \times 1$. It should be noticed that in this case, not more than $2H_c$ constraints can be active simultaneously. As it can be observed in (5.30), vector $\underline{b}$ is again a "dynamic" vector.

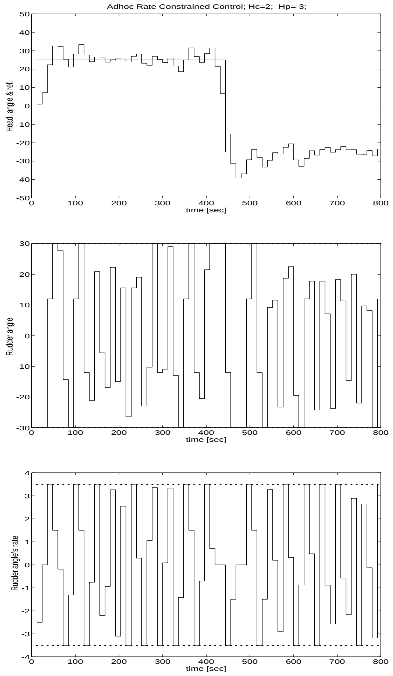

Figure 5.13. *Ad-hoc strategy.*

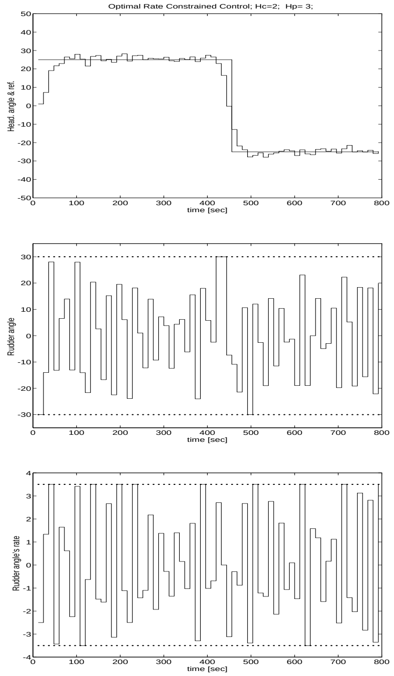

Figure 5.14. *Optimal strategy.*

Influence of the optimal level and rate constraints control

The ad-hoc way to handle both level and rate constraints consists in truncation of the control signal as following:

$$
\begin{aligned}
u^*(k) \quad = \quad & \max[\min[u(k), u^{max}(k), \Delta u^{max}(k) \cdot T_s + u(k-1)], \\
& u^{min}(k), \Delta u^{min}(k) \cdot T_s + u(k-1)]
\end{aligned}
$$

In the following example an analysis on the influence of the optimal strategy is done.

Example 5.8 *The simulations are performed for the design parameters:*

$$H_m = 1, \quad H_c = 2, \quad H_p = 3, \quad \rho = 0.0001$$

Both level and rate constraints on the controller output are now considered:

$$\begin{aligned} u^{min} &= -30 \; [deg] & u^{max} &= 30 \; [deg] \\ \Delta u^{min} &= -3.5 \; [deg/s] & \Delta u^{max} &= 3.5 \; [deg/s] \end{aligned} \tag{5.31}$$

The accuracy used for finding the optimal solution is $\eta = 10^{-2}$. Figure 5.13 and 5.14 show the performance of the closed loop system in the case of ad-hoc and optimal control, respectively. Again a better performance of the closed loop system is observed when a systematic way of handling the hard constraints is considered. ◇

Influence of the prediction horizon H_p

Increasing the prediction horizon H_p improves the performance of the system, as it is shown in Table 5.5. The simulations are performed for the same settings of the controller design parameters as in Example 5.1.

Again, it can be concluded that the bigger H_p is chosen, the smaller are J_u and the number of times the constraints are violated, i.e. number n_R of

H_p	Method	J_u	J_y	n_R
2	Ad-hoc	$2.7523 \cdot 10^4$	$2.2964 \cdot 10^4$	—
	DGP	$1.6867 \cdot 10^4$	644.7303	34
3	Ad-hoc	$2.5558 \cdot 10^4$	$1.338 \cdot 10^3$	—
	DGP	$1.4241 \cdot 10^4$	995.0585	17
4	Ad-hoc	$8.5276 \cdot 10^3$	$2.7429 \cdot 10^3$	—
	DGP	$8.0925 \cdot 10^3$	$2.6188 \cdot 10^3$	8
5	Ad-hoc	$6.2197 \cdot 10^3$	$4.7166 \cdot 10^3$	—
	DGP	$5.8309 \cdot 10^3$	$4.5559 \cdot 10^3$	5
10	Ad-hoc	$3.345 \cdot 10^3$	$1.5668 \cdot 10^3$	—
	DGP	$3.345 \cdot 10^3$	$1.5668 \cdot 10^3$	3

Table 5.5. *Performance criteria as a function of H_p in case of level and rate constraints.*

times DGP is activated. It is also observed that by increasing H_p, as the system becomes slower, J_y has a slight tendency to increase. The results from DGP strategy are significantly better than those from ad-hoc strategy.

Influence of the initialization

Simulations have shown that in case of level and rate constrained control, as in case of rate constraints, the starting point should be calculated using the second method suggested on page 159, otherwise an infeasible initial point may be generated.

Summarizing remarks

In this section the level and rate constrained predictive control was discussed. The following conclusions can be drawn based on the simulation results:

- A significant improvement in the performance of the closed loop system is observed in case when the level and rate constraints are systematically incorporated in the minimization of the criterion function.
- Better performance is achieved using a larger prediction horizon.
- The number n_R of times where DGP is activated, decreases with an increasing of H_p.
- Vector $\underline{b}$ is a "dynamic" vector, a fact which implies that the initialization of the optimization algorithm requires some precautions. As in the rate constraints case, the starting point based only on the preceding sample, provides a feasible point.

5.8 Summary

In this chapter, aspects related to the constrained predictive control were considered and illustrated by simulations with a stochastic ship model. It was presented an algorithm (DGP), based on Rosen's gradient projection method, able to take the hard constraints explicitly into account in the minimization of the cost function over the control horizon. As hard constraints, level and rate constraints on the control signal were considered.

The objective was to demonstrate the effectiveness of the new control algorithm compared to the ad-hoc strategy. As simulation examples, a stochastic Mariner Class Vessel model was used. The analysis was motivated, in particular, by the physical limitations present in the steering machine of a ship. It was shown that if the constraints are not taken into account explicitly, the performance of the system can be rather bad. Moreover, when there are rate constraints, the system can even become unstable for an ad-hoc strategy.

Due to the simplicity of the constraints, it was shown that the method DGP is simple and fast. The influence of initialization, of the accuracy and of the prediction horizon on the performance of the closed loop system were illustrated. The initialization of the algorithm DGP must take into account the fact that the initial solution must be feasible, otherwise the algorithm does not start. It was observed that the starting point of the optimization had a great effect on the number of iterations required to find the optimal solution with a particular accuracy. It was also shown that using a very small value of accuracy did not result in a better performance. It only caused a larger number of iterations. Finally, the use of a larger prediction horizon implied an improvement of the performance of the closed loop system. Thus, via a simulations set, it has been illustrated that it is highly advantageous to formulate and to solve the constrained optimization problem.

Chapter 6

Parameter estimation algorithms

In the controller design, it is generally assumed that a model of the process and of the disturbances are available. However, this is not always the case in practice. The problem of formulating a mathematical model can in principle be attacked in two ways. The first one, namely the classical modelling approach, involves an analytical/deductive method, where the model is built based on physical knowledge about the system. As it was described in Chapter 3, in order to design a controller, the high order resulting model must, in most cases, be further reduced to a low order model. The second one, is an inductive approach. It is known as the system identification method and it is based on the idea that the model is fitted to the measured

data from experiments performed on the system. The fitting is done by assigning suitable values to its parameters.

The goal of the chapter is to introduce the basic estimation procedures, in order to support the intuitive understanding of the connection between identification and control, which will be analyzed in the next chapter. The chapter gives also a survey of variants of the RLS recursive least squares estimator, able to track parameter variations. RLS is one member of a family of prediction error identification methods that are based on the minimization of prediction error functions.

6.1 Some formal definitions

Before entering the presentation of the recursive estimation methods, it is necessary to establish a suitable framework for the description. This is the objective of this section. It will be assumed that the system equation has a known linear structure.

Suppose that the true system is given by:

$$(\mathcal{S}): \qquad y(t) = G_0(q^{-1})u(t) + v(t) \tag{6.1}$$

where $G_0(q^{-1})$ is a strictly proper rational transfer function. The argument q^{-1} denotes the backward shift operator. $u(t)$ is the control input signal and $v(t)$ is an unmeasurable disturbance acting on the output $y(t)$.

For system (6.1), a parameterized model set of the following form will be considered:

$$(\mathcal{M}): \qquad y(t) = G(q^{-1})u(t) + H(q^{-1})e(t) \tag{6.2}$$

where $e(t)$ is a zero mean white noise, while $G(q^{-1})$ and $H(q^{-1})$ are proper rational transfer functions. They are dependent on the parameter vector $\underline{\theta}$, but for notational convenience, this dependence is not explicitly shown.

Equation (6.2) describes a general linear model. It can be used to provide different well known model structures, as e.g.:

- ARX - model, given by:

$$A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + e(t) \tag{6.3}$$

where:

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_{n_a} q^{-n_a} \\ B(q^{-1}) &= b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b} \end{aligned} \tag{6.4}$$

According to (6.2), in (6.3) the parameterized filters are:

$$\begin{cases} G(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} \\ \\ H(q^{-1}) = \frac{1}{A(q^{-1})} \end{cases} \tag{6.5}$$

The ARX model (6.3) can be rearranged into the well-known linear regression form:

$$y(t) = \underline{\varphi}^T(t)\,\underline{\theta} + e(t) \tag{6.6}$$

where $\underline{\varphi}^T(t)$ is the regression vector and $\underline{\theta}$ is the parameter vector. They are defined as:

$$\begin{aligned} \underline{\varphi}^T(t) &= [-y(t-1) \quad \cdots \quad -y(t-n_a) \quad u(t-d) \quad \cdots \quad u(t-d-n_b)] \\ \underline{\theta} &= \begin{bmatrix} a_1 & \cdots & a_{n_a} & b_0 & \cdots & b_{n_b} \end{bmatrix}^T \end{aligned} \tag{6.7}$$

- ARMAX - model, given by:

$$A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + C(q^{-1})e(t) \tag{6.8}$$

where $A(q^{-1})$ and $B(q^{-1})$ are defined as above, while:

$$C(q^{-1}) = 1 + c_1 q^{-1} + \cdots + c_{n_c} q^{n_c}$$

The corresponding filters of the general model (6.2) are:

$$\begin{cases} G(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} \\ \\ H(q^{-1}) = \frac{C(q^{-1})}{A(q^{-1})} \end{cases} \tag{6.9}$$

In this case the regression vector and the parameter vector become:

$$\begin{aligned}\underline{\varphi}^T(t) &= [-y(t-1)\ \cdots\ -y(t-n_a)\ u(t-d)\ \cdots\ u(t-d-n_b) \\ &\quad e(t-1)\ \cdots\ e(t-n_c)] \\ \underline{\theta} &= \begin{bmatrix} a_1 & \cdots & a_{n_a} & b_0 & \cdots & b_{n_b} & c_1 & \cdots & c_{n_c}\end{bmatrix}^T\end{aligned} \tag{6.10}$$

6.2 Recursive one step prediction error methods

Usually a model as (6.2), obtained by identification, is used for prediction. This is often inherent when the model is to be used as a basis for control system synthesis for the stochastic systems. There, the output at time t cannot be determined exactly from data available at time $t-1$, and therefore a prediction of the output, based on past data, must be done. Therefore it makes sense to determine a model parameter vector $\underline{\theta}$ so that e.g. the one step ahead prediction error:

$$\epsilon(t) = y(t) - \hat{y}(t|t-1) \tag{6.11}$$

is small. $\hat{y}(t|t-1)$ denotes the one step ahead prediction of $y(t)$, given the data up to and including time $t-1$, based on the parameter vector $\underline{\theta}$.

According to *[Ljung, 1987]*, the one step ahead predictor for $y(t)$ of the general model (6.2) is given by[1]:

$$\hat{y}(t|t-1) = H^{-1}\, G\, u(t) \ + \ [1 - H^{-1}]\, y(t) \tag{6.12}$$

The parameter vector $\underline{\theta}$ should be restricted to those values for which the predictor is asymptotically stable *[Söderström and Stoica, 1989]*, namely $\underline{\theta}$ should belong to $\mathcal{D}_\theta$, defined as:

$$\begin{aligned}\mathcal{D}_\theta \ = \ \{\ \underline{\theta}\ |\ & H^{-1} \text{ and } H^{-1}\, G \\ & \text{are asymptotically stable}\}\end{aligned} \tag{6.13}$$

In order to assure that the predictor $\hat{y}(t|t-1)$ depends only on the previous data (i.e. $u(t-1)$, $u(t-2)$, $\cdots$, $y(t-1)$, $y(t-2)$, $\cdots$), not also on $u(t)$ and $y(t)$, two other assumptions should be satisfied: $G(0) = 0$ and $H^{-1}(0) = 1$.

From (6.11) and (6.12) the one step ahead prediction error can be expressed as:

$$\epsilon(t) = \ H^{-1}\, [y(t) - G\, u(t)] \tag{6.14}$$

[1] For notational convenience the dependence with q^{-1} and $\underline{\theta}$ is dropped.

Once the prediction error $\epsilon(t)$ is available, a criterion function can be defined. The most common criterion used to identify the 'best' model in the model set (6.2) is the sum of the squares of the prediction errors:

$$J(\underline{\theta}) = \frac{1}{N}\sum_{t=1}^{N}[\epsilon(t)]^2 \tag{6.15}$$

Other types of criteria can be found in *[Söderström and Stoica, 1989]* and *[Ljung, 1987]*. The minimization of the criterion (6.15) leads to the the recursive prediction error algorithm:

$$\boxed{\begin{aligned} \epsilon(t) &= y(t) - \hat{y}(t|t-1) \\ \underline{\hat{\theta}}(t) &= \underline{\hat{\theta}}(t-1) + \underline{K}(t)\,\epsilon(t) \\ \underline{K}(t) &= \boldsymbol{P}(t)\,\underline{\psi}(t) = \frac{\boldsymbol{P}(t-1)\,\underline{\psi}(t)}{1+\underline{\psi}^T(t)\,\boldsymbol{P}(t-1)\,\underline{\psi}(t)} \\ \boldsymbol{P}(t) &= [\,\boldsymbol{I} - \underline{K}(t)\,\underline{\psi}^T(t)\,]\boldsymbol{P}(t-1) \end{aligned} \quad (6.16)}$$

where $\boldsymbol{P}(t)$ is proportional to the covariance matrix, which contains information about the confidence of the parameters and the search direction. $\underline{\psi}(t)$ represents the gradient of the prediction error, namely:

$$\underline{\psi}(t) = -\left[\frac{\partial\epsilon(t;\underline{\theta})}{\partial\underline{\theta}}\right]^T \tag{6.17}$$

and $\underline{K}(t)$ is the algorithm gain. $\underline{K}(t)$ can be seen as a linear projection of the gradient vector $\underline{\psi}(t)$ on the search direction.

It is noted that the algorithm (6.16) is very general, because it does not assume a specific model structure. Various recursive estimation schemes correspond to different model structures and choices of gradient vector $\underline{\psi}(t)$. Thus, e.g in the case of the ARX model (6.3), the prediction error (6.11) is:

$$\epsilon(t) = y(t) - \underline{\varphi}^T(t)\,\underline{\theta} \tag{6.18}$$

while the gradient of the prediction error is equal to the regression vector:

$$\underline{\psi}(t) = \underline{\varphi}(t) \tag{6.19}$$

Thus, for the ARX model the recursive prediction error algorithm (6.16) leads to the standard recursive least squares method (RLS). For the ARMAX model, based on (6.16) and on the way the prediction error is calculated, RELS (Recursive Extended Least Square) and RML (Recursive Maximum Likelihood) algorithms can be defined. In general, they are subjected to convergence problems and they are computationally burdensome (see e.g. *[Söderström and Stoica, 1989]*).

The recursive algorithm (6.16) is initialized by $\underline{\hat{\theta}}(0)$ and $\boldsymbol{P}(0)$. Their choice reflects the a priori knowledge about the parameter vector $\underline{\hat{\theta}}(0)$. Thus, if an a priori estimate of $\underline{\theta}(0)$ exists, then $\underline{\theta}(0)$ and $\boldsymbol{P}(0)$ should be chosen according to this estimate. If $\underline{\theta}$ is unknown then, one may choose:

$$\underline{\theta}(0) = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}^T \tag{6.20}$$

and

$$\boldsymbol{P}(0) = c\boldsymbol{I} \tag{6.21}$$

where c is a large positive constant and $\boldsymbol{I}$ is the identity matrix.

It is well-known that, in order to ensure the stability of the parameter estimation algorithm, the covariance matrix $\boldsymbol{P}$ should be positive definite. In computers with limited computational accuracy, a direct implementation of the presented algorithms is not recommended, because the positive definiteness of the $\boldsymbol{P}$ matrix might be lost due to the accumulated round-off errors. This problem is likely to occur if $\boldsymbol{P}$ becomes ill conditioned and especially if, the dimension of $\boldsymbol{P}$ is high. Moreover, by a direct implementation, the symmetry of the covariance matrix $\boldsymbol{P}$ is not assured, a fact which can lead to erroneous parameter estimates. Therefore, it is of crucial importance that the computations are performed in a proper numerical manner. One method for improving the robustness against round-off errors is the numerical stabilization, suggested by *[Bierman, 1977]*. All simulations presented in this thesis have been performed using this method. Another way is the $\boldsymbol{UDU}^T$ factorization of the $\boldsymbol{P}$ matrix, described in *[Bierman, 1977]*. Detailed analysis of these numerical algorithms can be found in the above-mentioned references.

6.3 Identification of the time-varying systems

The idea of the possibility of handling time-varying systems is strongly related to the realization of a self tuning controller. The identification algorithm, as a part of a self tuning controller, must be able to handle time variations in the parameters. The variation of the parameters can be slow or abrupt in time.

The ordinary recursive least squares algorithm (RLS) is known to be the most popular parametric identification method, as long as the model parameters are assumed to be time-invariant. When the parameters vary in time, RLS is not directly applicable, because the covariance matrix $\boldsymbol{P}$, and hence the gain $\underline{\mathrm{K}}$, tends to zero as the number of observations increases. This means that the new information from the measurement does not make any contribution to the parameter estimation, which makes it impossible for RLS to track the variations in the parameters. Several modifications to the ordinary RLS scheme have been proposed in the literature to prevent "the sleeping" of the estimator. Even though most of these methods are developed for slow time variation systems, they are actually able to handle parameter jumps as well *[Parkum, 1992]*.

The purpose of the next sections is to describe methods which handle the time variations by gradually discounting old information. This is the idea behind the so-called forgetting factor methods. It is important to underline that these methods are based on the idea that the parameters are so slow that they can be regarded as constant within a horizon with a reasonable length.

6.3.1 Constant forgetting factor algorithm

One method of ensuring that the RLS algorithm retains its alertness, is to introduce a forgetting or discounting factor λ. Effectively, this means the minimization of the criterion where the most recent data has the highest weights:

$$J = \sum_{k=1}^{t} \lambda^{t-k} \left[y(t) - \underline{\varphi}^T(t)\hat{\underline{\theta}}(t-1)\right]^2, \qquad 0 < \lambda \leq 1 \tag{6.22}$$

Notice that the old data for constant λ are exponentially discounted. The minimization of (6.22) leads to the following recursive formulation, see e.g. *[Ljung, 1987]*:

$$\boxed{\begin{aligned}
\epsilon(t) &= y(t) - \underline{\varphi}^T(t)\,\hat{\underline{\theta}}(t-1)\\
\underline{K}(t) &= \boldsymbol{P}(t)\,\underline{\varphi}(t) = \frac{\boldsymbol{P}(t-1)\,\underline{\varphi}(t)}{\lambda + \underline{\varphi}^T(t)\,\boldsymbol{P}(t-1)\,\underline{\varphi}(t)}\\
\hat{\underline{\theta}}(t) &= \hat{\underline{\theta}}(t-1) + \underline{K}(t)\,\epsilon(t)\\
\boldsymbol{P}(t) &= \tfrac{1}{\lambda}\left[\boldsymbol{I} - \underline{K}(t)\,\underline{\varphi}^T(t)\right]\boldsymbol{P}(t-1)
\end{aligned}} \tag{6.23}$$

The role of introducing λ is to keep the covariance matrix $\boldsymbol{P}$ away from zero, and thus to be able to handle time varying parameters. However,

this aspect is achieved as long as the input signal is persistently excited, see e.g. *[Söderström and Stoica, 1989]*. Otherwise, the $\boldsymbol{P}$ matrix increases exponentially according to:

$$\boldsymbol{P}(t) \approx \frac{\boldsymbol{P}(t-1)}{\lambda} \tag{6.24}$$

This phenomenon is known as "covariance wind up" or "blow-up", and it is due to the fact that the maximum eigenvalue of $\boldsymbol{P}(t)$ is not bounded. If $\boldsymbol{P}$ becomes very large then, even a small measurement error can lead to a very large and perhaps erroneous change of the estimates. This is very serious drawback of the constant forgetting factor algorithm. Another drawback can consist in the fact that it is impossible to utilize a priori information, e.g. that some parameters vary faster than others *[Parkum, 1992]*.

The choice of the value of λ is a trade-off between tracking ability and noise sensitivity: a smaller value of λ gives a faster discounting (i.e. faster tracking) but leads to a greater noise sensitivity; a larger value of λ provides a slower discounting (i.e. difficult to track fast parameter variations).

	Depth-to-draught ratio			
	∞	2.50	1.93	1.50
T	92.22	75.32	66.08	26.94
K	-0.102	-0.095	-0.101	-0.047
a	-0.8780	-0.8527	-0.8339	-0.6405
b	-0.0124	-0.0140	-0.0168	-0.0169

Table 6.1. *Shallow water parameters of the continuous and the discret model – $T_s = 12$ seconds – for Mariner Class Vessel at 6.17 [m/s] (12 knots).*

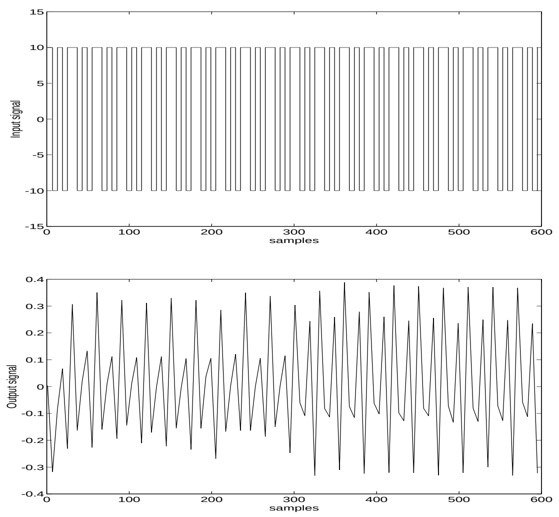

Figure 6.1. *Input and output signals during the experiment.*

Example 6.1 *In this example the constant forgetting factor method is illustrated on a ship dynamic problem. It is known that the coefficients of the maneuvering equations of surface ships are influenced by surge speed, loading, trim and water depth [Åström, 1980]. In the present example, the variation of the coefficients related with the water depth is regarded, due to the fact that, e.g. in Danish waters, the water depth can unexpectantly become very small.*

The coefficients of the Nomoto model:

$$G_{r\delta}(s) = \frac{K}{1 + s\,T} \tag{6.25}$$

for varying depth-to-draught are given in Table 6.1 (adopted from [Mort and Linkens, 1981]) and they correspond to the Mariner Class Vessel at a speed of 6.17 m/s (12 knots) for the full scale ship. The system (6.25) is sampled with a sampling period of $T_s = 12$ seconds:

$$\tilde{G}_{r\delta}(q^{-1}) = \frac{b\,q^{-1}}{1 + a\,q^{-1}} \tag{6.26}$$

The parameters a and b are also presented in Table 6.1, for different depth-to-draught ratio.

The model description, which is wanted to be estimated, is assumed to have the correct structure, i.e. (6.26).

An on-line identification experiment of 600 samples is performed, and the input-output signals are as shown in Figure 6.1. The input is a square wave of 10 samples and amplitude 10. Identical noise[1] and input sequences are used in all simulations. The RLS algorithm with constant forgetting factor is applied, for different values of λ. The initial conditions in all simulations are chosen as: $\underline{\theta}(0) = \underline{0}$ and $\boldsymbol{P}(0) = 1000\boldsymbol{I}$.

For the first 300 samples, it is assumed that the depth-to-draugth ratio for the ship is infinite (see Table 6.1). At the moment sample 300, a set point change occurs, namely the depth-to-draught ratio decreases to 1.5 (see Table 6.1).

Figure 6.2 illustrates that a small value of λ gives a faster adaptation but also a stronger variation of the estimates. The situation is opposite when λ

[1] *$e(t)$ is zero mean white noise with variance 1.*

is close to 1. This compromise in choosing λ is also shown in Figure 6.3, where the variation of the estimates and the tracking ability are plotted as function of the forgetting factor λ.

The variation of the parameters (measured e.g. for the sample interval $50 \leq t \leq 300$) can mathematically be expressed by:

$$V(\underline{\hat{\theta}}) = \sum_{t=50}^{300} (\underline{\hat{\theta}} - \overline{\underline{\theta}})^T (\underline{\hat{\theta}} - \overline{\underline{\theta}}) \tag{6.27}$$

where

$$\overline{\underline{\theta}} = \frac{1}{251} \sum_{t=50}^{300} \underline{\hat{\theta}}$$

The tracking ability TA can be measured by the length of the time interval from the parameter jump takes place, until 90% of the step height in the parameters is reached [Parkum, 1992], i.e. until:

$$TA(i) = arg \min_i \frac{\underline{\tilde{\theta}}^T \underline{\tilde{\theta}}}{\underline{\Delta\theta}^T \underline{\Delta\theta}} \leq 0.1, \qquad i = t - 300$$

where $\underline{\theta}^0$ denotes the real value of the parameters, $\underline{\tilde{\theta}} = \underline{\theta}^0(301) - \underline{\hat{\theta}}(301)$ denotes the estimation error, while $\underline{\Delta\theta} = \underline{\theta}^0(301) - \underline{\theta}^0(300)$ is the parameter change at $t = 301$. Figure 6.3 illustrates the compromise between the sensibility of the estimates on one hand and the ability to track the time variations on the other. ◇

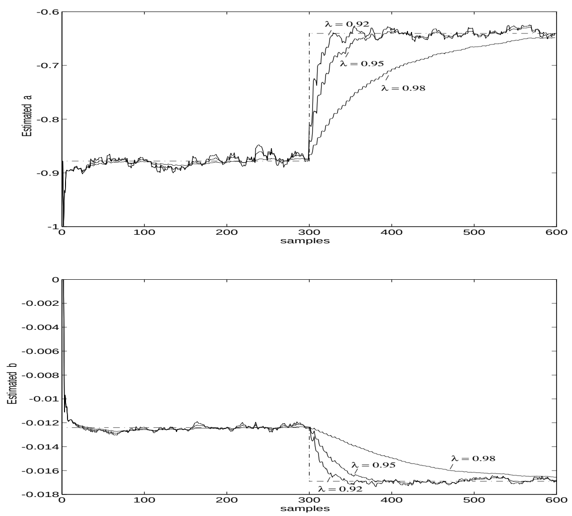

Figure 6.2. *Estimated $\hat{a}$ and $\hat{b}$ parameters for different values of λ. Dash-dotted line denotes the real values of the parameters.*

Example 6.2 *In order to illustrate the "covariance wind up" phenomena, the system from Example 6.1 is further used. However, this time it is assumed that the depth-to-draught ratio for the ship remains constant to infinity. Thus the coefficients remain constant: $a = -0.8780$, $b = -0.0124$. The input signal is the same used in Example 6.1, with the exception that, between sample 200 and sample 600, the input signal is identical to 1. The constant forgetting factor algorithm is applied for $\lambda = 0.96$.*

The result is shown in Figure 6.4. It is observed that after sample 200, as it was expected, the "covariance wind up" phenomena appears. The constant input, which is applied after sample 200, makes only one direction of the parameter space to be excited. Thus the corresponding eigenvalue decreases.

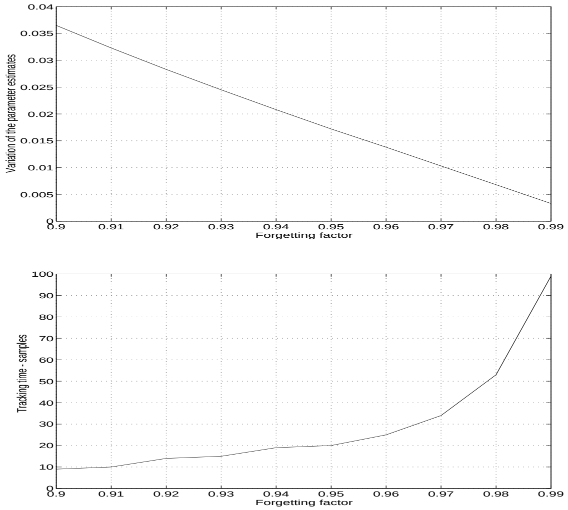

Figure 6.3. *Variation of the estimates and the tracking ability versus the forgetting factor.*

The perpendicular direction will not be excited and this eigenvalue increases exponentially. Therefore the variation of the $\hat{a}$ estimate becomes larger and larger. This is reflected in the trace of $\boldsymbol{P}$, which starts to grow almost exponentially. ◇

6.3.2 Variable forgetting factor algorithm

To overcome the problems of $\boldsymbol{P}$ blowing up, a variable forgetting factor in the identification algorithm can be introduced. The basic idea is to reduce the degree of forgetting in periods with low excitation. A method for

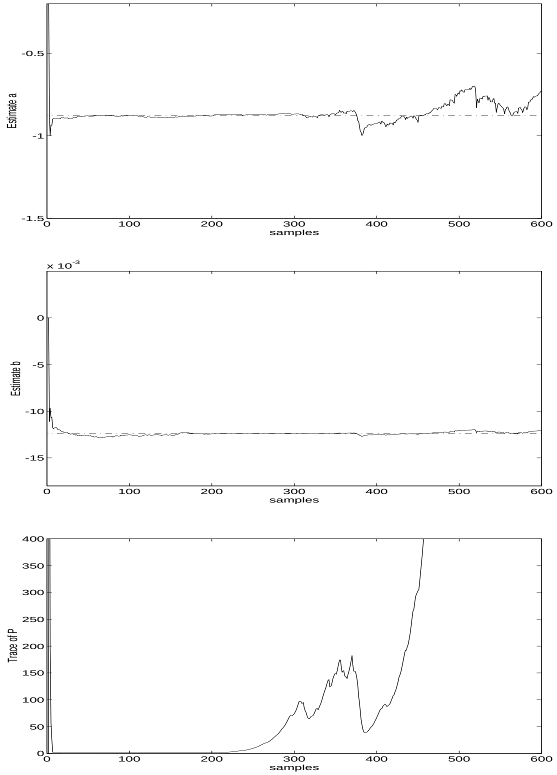

Figure 6.4. *Estimated parameters and the trace of the covariance matrix $\boldsymbol{P}$ when the input is not persistently excited. Dash-dotted line denotes the real parameter values.*

doing this was proposed in *[Fortescue et al., 1981]*, in which the forgetting factor λ is varied so that the criterion (6.22) is kept constant at a constant value J_0. By keeping J constant, the amount of forgotten information will correspond to the amount of new information obtained at each sample, loosely speaking. This aspect is important especially for adaptive control, where the cost function J often is an indirect measure of the excitation level. A small value of J indicates that the parameters are good, and this leads to poor excitation, i.e. a low information content of the data. A big value of J indicates large prediction errors, which provide a good excitation and a high level of information. Hence, keeping J constant to J_0, the forgetting factor can be expressed as:

$$\lambda(t) \approx 1 - \frac{1}{2J_0} \; \frac{\epsilon^2(t)}{1 + \varphi^T(t)\, \boldsymbol{P}(t-1)\, \varphi(t)} \tag{6.28}$$

J_0 can be thought of as a tuning parameter. See *[Poulsen, 1985]* for further details.

The idea of the algorithm is thus that the forgetting factor is chosen based on the prediction error $\epsilon(t)$. When the parameters are close to their correct values then, $\epsilon(t)$ is small compared to its expectation and the forgetting factor λ should be close to one $\lambda(t) \approx 1$, whereas large values of $\epsilon(t)$ forces $\lambda(t)$ away from one. Thus, the risk of the "covariance wind up" can be reduced.

Simulation studies in *[Parkum, 1992]* show that the algorithm can be suitable as long as the time variation of the parameters is not too slow. Otherwise a "covariance wind up" may still occur.

6.3.3 Constant trace algorithm

Another way to overcome the blowing up problems of $\boldsymbol{P}(t)$ is to keep the trace of $\boldsymbol{P}(t)$ constant at each iteration:

$$tr\{\boldsymbol{P}(t)\} = tr\{\boldsymbol{P}(t-1)\} \tag{6.29}$$

By applying this demand, algebraic computations leads to the following expression of the forgetting factor (see e.g. *[Parkum, 1992]*):

$$\lambda(t) = 1 - \frac{1}{tr\{\boldsymbol{P}(t-1)\}} \cdot \frac{\underline{\varphi}^T(t)\, \boldsymbol{P}^2(t-1)\, \underline{\varphi}(t)}{1 + \underline{\varphi}^T(t)\, \boldsymbol{P}(t-1)\, \underline{\varphi}(t)} \tag{6.30}$$

Hence, keeping the trace of $\boldsymbol{P}(t)$ constant, the eigenvalues of $\boldsymbol{P}(t)$ are bounded from above but not from below. Even if the trace of $\boldsymbol{P}(t)$ is kept constant, some eigenvalues of $\boldsymbol{P}(t)$ may converge to zero *[Parkum, 1992]*, leading to a complete loss of tracking ability in the corresponding directions.

The selection of the constant value of the trace is a compromise between tracking speed and noise sensitivity. A small trace gives a slow tracking, while a high trace will cause the parameter estimates to oscillate around the true values due to a too large gain (keeping the trace constant will counteract the decrease of the gain in situations with large excitation). One way of solving this problem is to use the constant trace algorithm in conjunction with the constant forgetting factor algorithm. The constant trace is used only in low excitation periods. By doing this the best qualities of both algorithms can be utilized.

6.3.4 SF1 algorithm

An attractive algorithm, which ensures an upper bound, as well as a lower bound of the matrix $\boldsymbol{P}(t)$, is the SF1 algorithm (Selective Forgetting Version 1) proposed by *[Parkum, 1992]*. It is slightly more computationally demanding compared to the ordinary RLS algorithm.

The method has a close parallel to the EFRA (Exponential Forgetting and Reseting Algorithm) *[Salgado et al., 1988]*, but it is less computational expensive. It can be viewed as a modification to the Kalman-filter, which guarantees the diagonal elements of the covariance matrix to remain bounded within a specified interval $[\alpha_{min}, \alpha_{max}]$:

$$\boxed{\begin{aligned}
\epsilon(t) &= y(t) - \underline{\varphi}^T(t)\,\underline{\hat{\theta}}(t-1) \\
\underline{K}(t) &= \boldsymbol{P}(t)\,\underline{\varphi}(t) = \frac{\boldsymbol{P}(t-1)\,\underline{\varphi}(t)}{(1+\underline{\varphi}(t)^T\,\boldsymbol{P}(t-1)\,\underline{\varphi}(t))} \\
\underline{\hat{\theta}}(t) &= \underline{\hat{\theta}}(t-1) + \underline{K}(t)\,\epsilon(t) \\
\boldsymbol{P}(t) &= \frac{\alpha_{max}-\alpha_{min}}{\alpha_{max}}\,[\boldsymbol{I} - \underline{K}(t)\,\underline{\varphi}^T(t)]\,\boldsymbol{P}(t-1) + \alpha_{min}\boldsymbol{I}
\end{aligned} \qquad (6.31)}$$

In *[Parkum, 1992]* it is shown that, if the initial values are set to:

$$\begin{aligned}
\underline{\hat{\theta}}(0) &= \underline{0} \\
\boldsymbol{P}(0) &= a\boldsymbol{I}, \quad \text{where} \quad 0 < a \leq \alpha_{max}
\end{aligned} \qquad (6.32)$$

then SF1 algorithm has the property:

$$\alpha_{min}\boldsymbol{I} \leq \boldsymbol{P}(t) \leq \alpha_{max}\boldsymbol{I}, \qquad t \geq 0, \qquad (6.33)$$

where the lower bound α_{min} ensures that the algorithm gain does not converge to zero, while the upper bound α_{max} is a guarantee against blowing up. α_{min} and α_{max} are the only tuning parameters of the method. Empirically it has been found that a good choice of α_{max} is $\alpha_{max} = p\,\alpha_{min}$, where $10 \leq p \leq 1000$, while typically $0.001 \leq \alpha_{min} \leq 0.1$. Simulation studies in *[Parkum, 1992]* illustrate that the algorithm handles slow parameter variations very well.

6.4 Summary

The purpose of this chapter was to present some forgetting factor methods discussed in the literature. The list of forgetting methods suggested in the literature is very long, and only few examples were given in this chapter. The basic principle of the forgetting factor algorithms which have been discussed here is to regard the parameters as almost constant within a certain time horizon. Old information is gradually discounted, and consequently only a limited amount of information is used by the estimator.

As the recursive least squares RLS is the most popular parametric identification method used in adaptive control, the chapter was mainly focused on variants of the ordinary recursive least squares formulation: Constant forgetting factor algorithm, Variable forgetting factor algorithm, Constant trace algorithm, SF1 algorithm. When model parameters are changing, it is important to keep the eigenvalues of the variance matrix $\boldsymbol{P}_t$ bounded from above and below. Thus the robustness towards time variations is improved by keeping $\boldsymbol{P}_t$ bounded from below and the robustness towards lack of excitation is improved by bounding $\boldsymbol{P}_t$ from above. This property is ensured only by SF1 and EFRA. As SF1 is more simple than the EFRA algorithm and gives similar performances, SF1 is more practically attractive.

Throughout this chapter, it was assumed that the model was of correct structure and correct order. However, in the presence of unmodelled dynamics, the recursive least square algorithms may cause the parameter estimates to drift, a fact which leads to significant problems. These consequences are less serious in the case when data are filtered by a low pass filter. This will be pursued in Chapter 7. Another way to reduce the effect of unmodelled dynamics is to introduce a dead zone in the parameter estimation algorithm, presented in details in *[Middleton et al., 1988]*.

Chapter 7

Predictive control relevant prefiltering

During the last years, the intimate interplay between the parameter estimation algorithm and the control algorithm became a major research activity in the adaptive control community. Indeed, adaptive control combines on-line identification with a control law.

Most conventional adaptive control systems are based on the certainty equivalence principle *[Åström and Wittenmark, 1989]*, where the true parameters are substituted by the parameter estimates, neglecting the uncertainty of the parameter estimates. In these studies, the effort is usually spent either on the control design with only a little attention paid to the

properties of the parameters, or on the parameters estimation with tiny consideration given to the way the controller affects the parameter estimator.

Most of the available theory for convergence of adaptive controllers is for the case when the estimated model is of the same complexity or higher than the true model. The reason is that, for a long time, the identification strategy has been viewed as a search for "the true system" rather than as a methodology whose aim is to produce an "approximation of reality". In the eighties, the effort shifted from the concept of identifying the true system to that of "approximating" the true system and of characterizing this approximation in terms of bias and variance of the transfer function estimates. Thus the identification task has been turned into tuning the model towards the objective for which the model is to be used. One of the main initial contributors is this engineering view of identification is Lennart Ljung in *[Ljung, 1987]*.

Usually, the estimated model is unable to completely match the true process, due to the fact that even though most real processes are of high order, the controller, for practical reasons, must be designed using a low order model. The inevitable model error should be adapted to the control objective, i.e. an improved estimation must be obtained in those frequency ranges which are relevant to the controller. Thus, the controller, designed on the estimated model, becomes robust with respect to the mismatch between the system and the model.

The control relevant identification viewpoint affects all stages of system identification: input-signal design, model structure selection, parameter estimation and model validation. These aspects have been emphasized in a number of interesting papers, as in *[Wahlberg and Ljung, 1986]*, *[Ljung, 1987]*, *[Mohtadi, 1988b]*, *[Åström, 1993]*, *[Gevers, 1991]*, *[Gevers, 1993]* and *[Hakvoort et al., 1994]*.

This chapter focuses on parameter estimation, in particular on the development and analysis of a control-relevant prefilter. Predictive control is considered. The basic theoretical elements and notations for identification are reviewed. The effects of prefiltering in identification and of the observer polynomial in the control law are analyzed and supported by some illustrative simulations on a Mariner Class Vessel model. An iterative procedure is also evaluated on examples.

7.1 The basic theory elements and notations

The identification framework is adopted from *[Ljung, 1987]*.

7.1.1 Bias and variance

Consider a discrete-time representation of a linear, time invariant SISO system with noise:

$$(\mathcal{S}): \qquad y(t) = G_0(q^{-1})\, u(t) + v(t) \tag{7.1}$$

where $G_0(q^{-1})$ is a strictly proper transfer function, $u(t)$ is the control input and $v(t)$ is an unmeasurable disturbance acting on the output $y(t)$. The

disturbance term $v(t)$ is assumed to be a zero mean stationary stochastic process with rational spectral density $\Phi_v(\omega)$. It can be represented by:

$$v(t) = H_0(q^{-1})\xi(t) \tag{7.2}$$

where $\xi(t)$ is a sequence of independent random variables with zero mean value and variance σ_0^2. Thus:

$$\Phi_v(\omega) = \sigma_0^2\, |H_0(e^{-j\omega})|^2 \tag{7.3}$$

Let us consider the model:

$$(\mathcal{M}): \qquad y(t) = G(q^{-1};\underline{\theta})\, u(t) + H(q^{-1};\underline{\theta})\, e(t) \tag{7.4}$$

where $\underline{\theta}$ is a parameter vector, which can be determined by a recursive estimator and $e(t)$ is zero mean white noise with variance σ^2. $G(q^{-1};\underline{\theta})$ and $H(q^{-1};\underline{\theta})$ are strictly proper transfer functions. The estimates of these transfer functions are in the following denoted by:

$$\hat{G}_N(e^{-j\omega}) = G(e^{-j\omega}, \hat{\underline{\theta}}_N), \qquad \hat{H}_N(e^{-j\omega}) = H(e^{-j\omega}, \hat{\underline{\theta}}_N) \tag{7.5}$$

where $\hat{\underline{\theta}}_N$ is the estimate of $\underline{\theta}$ obtained from N data.

An important aspect is that the true system $\mathcal{S}$ is not assumed to be exactly representable within the chosen model set $\mathcal{M}$, a fact which means that bias errors exist.

In practice, it is important to keep in mind that the model can never give a true and exact description of the system in question. Therefore, as the model structure is of lower complexity than the true system, the error on the estimated transfer function consists of two contributions, which are typically affected by the design variables in different ways:

- Asymptotic Bias Error $B(e^{-j\omega})$ – is an error independent of the number of data N. It is mostly affected by the model set (the unmodelled dynamics) and therefore it can theoretically be made arbitrarily small by selecting increasingly sophisticated model structures. For example, in control and adaptive control, typically rather simple models are used, which may give large bias.

- Random Error $R(e^{-j\omega}, \underline{\hat{\theta}}_N)$ – is an error due to noise and the finite amount of data N. This error stems from the fact that if an identification experiment is repeated with the same input, one will not obtain the same model. In general, the random error decreases with an increasing amount of data and input signal power, while it increases with the number of estimated parameters *[Ljung, 1987]*.

Thus the total model error between the true transfer function and the estimated model can be expressed as:

$$\begin{aligned}\Delta G(e^{-j\omega}) &= G_0(e^{-j\omega}) - \hat{G}_N(e^{-j\omega}) \\ &= \underbrace{G_0(e^{-j\omega}) - E\{\hat{G}_N(e^{-j\omega})\}}_{B(e^{-j\omega})} + \underbrace{E\{\hat{G}_N(e^{-j\omega})\} - \hat{G}_N(e^{-j\omega})}_{R(e^{-j\omega}, \underline{\hat{\theta}}_N)}\end{aligned} \tag{7.6}$$

Theoretically it is desirable that this error is kept small. But actually, in practice, depending on the intended use of the model (simulation, prediction, control), this error will not necessary be small for all frequencies. *[Ljung, 1987]* presents some weighting functions applied to this error for different applications.

The mean square error for (7.6) can be formed:

$$E\{|\Delta G(e^{-j\omega})|^2\} = |B(e^{-j\omega})|^2 + \boldsymbol{P}(\omega, \underline{\hat{\theta}}_N) \tag{7.7}$$

where:

$$\boldsymbol{P}(\omega, \underline{\theta}_N) = E\{|R(e^{-j\omega}, \underline{\hat{\theta}}_N)|^2\}$$

is the variance of the random error. For models of sufficiently high order, *[Ljung, 1987]* provides an approximation of $\boldsymbol{P}(\omega, \underline{\hat{\theta}}_N)$ as:

$$\boldsymbol{P}(\omega, \underline{\hat{\theta}}_N) \approx \frac{n}{N}\,\Phi_v \begin{bmatrix} \Phi_u(\omega) & \Phi_{u\xi}(\omega) \\ \Phi_{u\xi}(-\omega) & \sigma_0^2 \end{bmatrix}^{-1} \tag{7.8}$$

where $\approx$ indicates proportionality, n is the order of the model $G(e^{-j\omega}, \underline{\theta})$, N the number of data, Φ_v the noise spectrum, Φ_u the input spectrum, $\Phi_{u\xi}$ the cross spectrum between the input and the white noise source $\xi(t)$. This formula applies to both open loop identification ($\Phi_{u\xi} = 0$) and closed loop identification ($\Phi_{u\xi} \neq 0$). Most identification applications include an estimate of $\boldsymbol{P}(\omega, \underline{\hat{\theta}}_N)$ based on the assumption that the bias error is zero. However recently, *[Hjalmarsson and Ljung, 1992]* has provided an estimate for $\boldsymbol{P}(\omega, \underline{\hat{\theta}}_N)$ in the case when the bias is nonzero, an assumption which is more realistic for practical applications.

Since the main goal of the present chapter is to investigate the role of the prefilter in the interconnection between identification and control, random error is not considered. The effects of the variance can be eliminated by using infinitely long data sets or noise-free data. Analysis of variance of random error can be found in *[Hjalmarsson et al., 1994]*.

7.1.2 Prediction error method

In this section, due to their importance, some relevant aspects of the prediction error method from *[Ljung, 1987]* are given.

The basic way to estimate the parameter vector $\underline{\theta}$ in (7.4) is to form a predictor and to minimize the sum of the squared prediction errors with respect to $\underline{\theta}$. In the following, a k-step predictor is derived, in order to outline the role of the number of steps-ahead, used in prediction. To do so, some notations are necessary. Thus, the noise model can be written as:

$$H(q^{-1};\underline{\theta}) = \bar{H}_k(q^{-1};\underline{\theta}) + q^{-k}\tilde{H}_k(q^{-1};\underline{\theta}) \tag{7.9}$$

where:

$$\begin{aligned} \bar{H}_k(q^{-1};\underline{\theta}) &= \textstyle\sum_{j=0}^{k-1} h(j;\underline{\theta})q^{-j} \\ q^{-k}\tilde{H}_k(q^{-1};\underline{\theta}) &= \textstyle\sum_{j=k}^{\infty} h(j;\underline{\theta})q^{-j} \end{aligned} \tag{7.10}$$

Notice that for $k=1$, $\bar{H}_k(q^{-1};\underline{\theta}) = 1$.

In order to get the k-step ahead prediction, *[Wahlberg and Ljung, 1986]* introduces the following definition:

$$W_k(q^{-1}) \triangleq 1 - q^{-k}\tilde{H}_k(q^{-1};\underline{\theta})\, H^{-1}(q^{-1};\underline{\theta}) = \frac{\bar{H}_k(q^{-1};\underline{\theta})}{H(q^{-1};\underline{\theta})} \tag{7.11}$$

Based on (7.9) and (7.11), for given $u(s)$ $(s \leq t)$ and $y(s)$ $(s \leq t-k)$, the k-step ahead prediction of model (7.4) is:

$$\begin{aligned} \hat{y}(t|t-k;\underline{\theta}) &= W_k(q^{-1};\underline{\theta})\, G(q^{-1};\underline{\theta})\, u(t) \\ &+ \tilde{H}_k(q^{-1};\underline{\theta})\, H^{-1}(q^{-1};\underline{\theta}) y(t-k) \end{aligned} \tag{7.12}$$

Taking (7.1), (7.11) and (7.12) into account, the k-step ahead prediction error $\epsilon_t(k;\underline{\theta})$ becomes:

$$\begin{aligned} \epsilon_t(k;\underline{\theta}) &= y(t) - \hat{y}(t|t-k;\underline{\theta}) \\ &= \left\{ [G_0(q^{-1}) - G(q^{-1};\underline{\theta})]\, u(t) + v(t) \right\} \tfrac{\bar{H}_k(q^{-1};\underline{\theta})}{H(q^{-1};\underline{\theta})} \end{aligned} \tag{7.13}$$

The signal $\epsilon_t(k;\underline{\theta})$ can be filtered with a stable linear filter $L(q^{-1})$, which yields the filtered prediction error:

$$\epsilon_t^f(k;\underline{\theta}) = L(q^{-1})\,\epsilon_t(k;\underline{\theta}) \tag{7.14}$$

As shown in *[Wahlberg and Ljung, 1986]*, for SISO systems, filtering of the prediction error is equivalent to filtering of the input $u(t)$ and of the output $y(t)$ with the filter $L(q^{-1})$ before they are used in identification:

$$u^f(t) = L(q^{-1})\,u(t), \qquad y^f(t) = L(q^{-1})\,y(t) \tag{7.15}$$

or to replacing the noise model $H(q^{-1};\underline{\theta})$ by $H(q^{-1};\underline{\theta})/L(q^{-1})$.

An estimate of $\underline{\theta}$ is obtained by minimizing the quadratic norm of the filtered prediction error with respect to $\underline{\theta}$:

$$J(\underline{\theta}) = \sum_{t=1}^{N} [\epsilon_t^f(k;\underline{\theta})\,]^2 \tag{7.16}$$

For notational convenience, the dependence on q^{-1} will not always be explicitly mentioned in the following. A short-hand notation, for example $G(\underline{\theta})$ for $G(q^{-1},\underline{\theta})$, will be used instead.

In the study of the robustness concept in the adaptive framework, when the model arises as the result of an identification algorithm, it is important to be able to characterize the bias error over the frequency range. It is known that the frequency domain provides a useful framework for the analysis

of different processes and models regardless of their order. Therefore it is advantageous to analyze the distribution of the filtered prediction error over the frequency range.

7.1.3 Bias in open loop identification

Assume first that the data are collected while the process operates in open loop. In such case, the signals $u(t)$ and $v(t)$ are mutually uncorrelated.

By applying Parseval's identity to (7.16) and taking (7.13) into account, the following expression for $\bar{J}(\underline{\theta})$ is obtained:

$$\bar{J}(\underline{\theta}) = \int_{-\pi}^{\pi} \left\{ |G_0 - G(\underline{\theta})|^2\,\Phi_u + \Phi_v \right\}\,|L|^2 \left|\frac{\bar{H}_k(\underline{\theta})}{H(\underline{\theta})}\right|^2 d\omega \tag{7.17}$$

where Φ_u and Φ_v are the spectral densities of the control signal $u(t)$ and of the noise $v(t)$, respectively. It can be noted that the minimization of (7.17) constitutes a compromise between the weighted transfer function modelling error and the weighted noise spectrum.

Under mild conditions on the data (see *[Ljung, 1987]*), the estimate $\underline{\theta}$ converges to $\underline{\theta}^*$ as $N \rightarrow \infty$, where:

$$\underline{\theta}^* = \arg\min_{\underline{\theta}} \bar{J}(\underline{\theta}) \tag{7.18}$$

This means that the model $G(\underline{\theta})$ will converge to G_0 if the number of data N tends to infinity.

Expression (7.17) is the frequency domain representation for the asymptotic bias distribution *[Ljung, 1987]*. It is useful because it shows that, in the situation where some restricted complexity model structure has been chosen for G_0, one can still manipulate the frequency distribution of the plant-model mismatch to a certain extent by playing with the design variables Φ_u and L. However, one can notice that the filter L is a more flexible tool than Φ_u, due to the fact that every time a new Φ_u is required a whole new set of experiments have to be performed.

In *[Gevers, 1991]* it is pointed out that, in the case of undermodelling, different inputs used for identification may lead to very different parameter estimates. Thus a controller based on an estimated model may even destabilize an open loop stable plant. The reason is that the model fits to the true system in a poorly chosen frequency range or the control design parameters are not properly chosen.

In the case when both the input and the noise parts of the model depend on the parameter vector $\underline{\theta}$, the interpretation of the distribution of the bias (7.17) is somewhat complicated. However, it is possible to understand the bias distribution in the situation when one of the mentioned parts is fixed, e.g. independent of $\underline{\theta}$. For example, if the noise model is fixed (H notated as H^F), the noise spectrum does not influence the minimization of the cost function (7.17), and therefore it can be written as:

$$\bar{J}(\underline{\theta}) = \int_{-\pi}^{\pi} |G_0 - G(\underline{\theta})|^2 \, \Phi_u |L|^2 \left| \frac{\bar{H}_k^F}{H^F} \right|^2 d\omega \tag{7.19}$$

If $k = 1$ ($\bar{H}_k^F = 1$) the asymptotic bias distribution is affected by the weighting function $\Phi_u \left|\frac{L}{H^F}\right|^2$. Additionally, if the input signal Φ_u is white noise and $L = 1$ (no prefiltering of data), then:

$$\bar{J}(\underline{\theta}) = \int_{-\pi}^{\pi} |G_0 - G(\underline{\theta})|^2 \frac{1}{|H^F|^2} d\omega \tag{7.20}$$

Thus $\bar{J}(\underline{\theta})$ depends on H^F and, as H^F is a low-pass filter, in the minimization of the criterion (7.20) the high frequencies will be emphasized. This supports the observation that the one step ahead prediction error method focuses more on the high frequencies. Otherwise, if $k \to \infty$, then $\bar{H}_k^F \to H^F$, a fact which implies that all the frequencies are equally weighted and that, the input spectrum distributes the bias in (7.19), when $L = 1$. The presence of prefilter L in (7.19) also can be used to weight the bias distribution over the frequency range. Thus, in the presence of structural model-plant mismatch, the low frequency behavior of a model can be improved by using a prefilter ($L \neq 1$) or multi-step predictions ($k \neq 1$).

The multi-step prediction error method is analyzed in details in *[Ljung, 1987]* and *[Söderström and Stoica, 1989]*. Even though the multi-step prediction error method can improve the robustness towards unmodelled dynamics, it has some practical deficiencies. It requires a large amount of computations, related especially to the gradients (see e.g. *[Lauritsen, 1994]*), it does not guarantee the convergence of the parameter estimates and it is quite sensitive to the initial conditions *[Söderström and Stoica, 1989]*. Therefore, in the present chapter, the attention is guided towards the analysis of the effect of prefiltering against unmodelled dynamics.

7.2 Why prefilter ?

The main function of a prefilter in adaptive control is to force the estimated model to retain those plant characteristics that are most significant with regard to user's control requirements. Thus, incorporating the control requirements in the parameter estimation problem, it becomes possible to obtain improved models over the frequency band of importance to the control problem.

In practice, an ad-hoc method is often used to compromise improved robustness of the controller in the presence of unmodelled dynamics against the ability to reject disturbances. Some few guidelines for an ad-hoc choice of a prefilter exist. The design of a prefilter is normally based on knowledge of the frequency content of the signal. Thus, a high-pass filtering is used to eliminate d.c. offsets, load disturbances, while a low pass filter eliminates irrelevant high frequency noise. A sensible choice of the prefilter is often a low-pass or band-pass filter with an upper cut-off frequency around the closed loop bandwidth. In *[Middleton et al., 1988]* it is suggested that the prefilter's cut-off frequency should be selected based on the desired closed loop bandwidth. The rule of thumb, governing the design of the prefilter, is that the upper frequency should be about twice the desired system bandwidth and the lower frequency should be about one tenth the desired bandwidth. The selection of sampling rate will often result in a change of prefilter (see e.g. *[Thygesen, 1993]*).

The effect of the relevant prefiltering has been preliminary analyzed and discussed for instance in *[Wahlberg and Ljung, 1986]*, *[Mohtadi, 1988b]*, *[Rivera et al., 1992]* and *[Lauritsen, 1994]*.

The following example illustrates how the prefilter affects the parameter estimation.

Example 7.1 *Consider the continuous time transfer function from rudder angle to yaw rate for a Mariner Class Vessel:*

$$G_{o,\,r\delta}(s) = \frac{-0.1835(1+18.55s)}{(1+7.91s)(1+116.28s)} \tag{7.21}$$

If the sampling period used for discretization is $T_s = 12$ seconds, then the discrete transfer function is:

$$G_{o,\,r\delta}(q^{-1}) = \frac{-0.0223q^{-1} + 0.0209q^{-2}}{1 - 1.1214q^{-1} + 0.1979q^{-2}} \tag{7.22}$$

Two deductive alternative low order approximations of the transfer function (7.22), discussed in Chapter 3, are:

$$\begin{array}{llcl} \textit{Nomoto approx.}: & G^N_{r\delta}(q^{-1}) & = & \frac{-0.0197\,q^{-1}}{1-0.8926\,q^{-1}} \\ \textit{BTA approx.}: & G^B_{r\delta}(q^{-1}) & = & \frac{-0.0216\,q^{-1}}{1-0.8753\,q^{-1}} \end{array} \tag{7.23}$$

These approximations can only be used at low frequencies. However, they are mentioned in this example to get an idea about the parameters of the reduced model, which this time, is intended to be determined using an inductive technique, e.g. an identification method.

Consider an RLS identification experiment of 500 samples, where the input signal is a zero mean white noise with unity variance.

The model to be estimated is a first order model:

$$G_{r\delta}(q^{-1}) = \frac{b\,q^{-1}}{1+\ a\,q^{-1}} \tag{7.24}$$

Notice that there is a mismatch between the assumed model structure and the underlying system (7.22). In order to emphasize the effects due to undermodelling, the simulation is noise free.

In case data are not prefiltered, the estimated model is:

$$\bar{G}_{r\delta}(q^{-1}) = \frac{-0.0278\,q^{-1}}{1-0.3174\,q^{-1}} \tag{7.25}$$

Traditionally, ad-hoc prefiltering is performed using an infinite impulse response (autoregressive) prefilter, namely $F(q^{-1}) = \frac{1}{L(q^{-1})}$. *In order to improve the low frequency estimation, a low pass prefilter* $F(q^{-1})$ *should be selected, for example:*

$$F(q^{-1}) = (1-0.95\,q^{-1}) \tag{7.26}$$

Using this filter, the following low order transfer function approximation (7.22) is obtained:

$$\bar{G}_{r\delta}(q^{-1}) = \frac{-0.0272\,q^{-1}}{1-0.8159\,q^{-1}} \tag{7.27}$$

It is observed that the parameters of the estimated model (7.27) are more close to those of Nomoto/BTA approximations than (7.25). Bode diagrams

of the transfer functions (7.22), (7.25), (7.27) are compared in Figure 7.1. As it was expected, in case data are not prefiltered, the estimator tends to emphasize high frequencies, being bad at low frequencies, while in case of using a low-pass prefilter for the data, a much better estimate of the transfer function for low frequencies is obtained. ◇

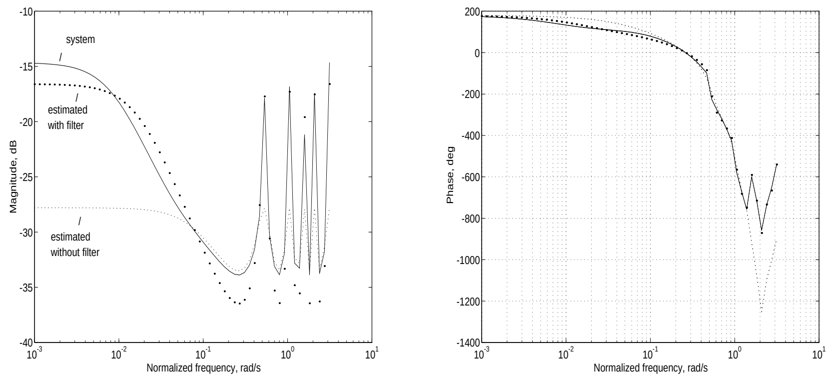

Figure 7.1. *The amplitude and phase characteristics of the system, and of the estimated models.*

7.3 Bias distribution in closed loop identification

As was pointed out, expression (7.17) requires independence between the input signal $u(t)$ and the noise sequence $v(t)$. Now consider data that are collected on the true system when some controller is operating. Thus, in

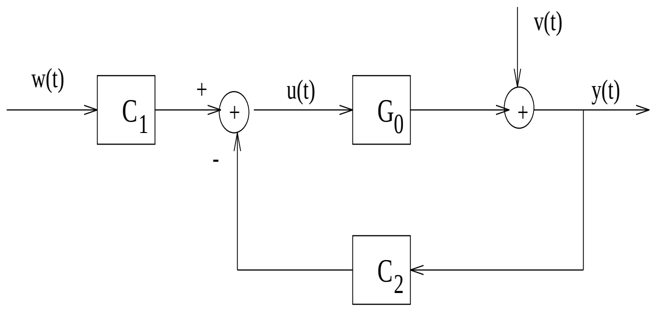

Figure 7.2. *Closed loop system.*

this section a closed loop performance criterion is defined as a measure of the model's capability to describe the controlled operation of the plant.

Let us assume that the plant is controlled by the two fixed components controller (C_1, C_2), which is assumed to be known:

$$u(t) = C_1\, w(t) - C_2\, y(t) \tag{7.28}$$

where C_1 and C_2 are transfer functions in the shift operator and $w(t)$ is the reference which is supposed to be independent from $v(t)$.

The theory discussed in this section does not make any assumptions about a specific controller design method. Many different methods can be used to obtain the controller polynomials, as e.g. the LQG controller *[Hakvoort et al., 1994]*, the pole placement controller *[Åström, 1993]*, *[Voda and Landau, 1994]* and the long range predictive controller *[Shook et al., 1991]*, *[Lauritsen, 1994]*.

The closed loop configuration is shown in Figure 7.2. The feedback system is driven by the external reference signal $w(t)$ and the external disturbance $v(t)$, which are assumed to be uncorrelated. According to Figure 7.2, the input signal $u(t)$ and the output signal $y(t)$ satisfy, respectively:

$$\begin{aligned} u(t) &= \tfrac{C_1}{1+G_0C_2}\, w(t) - \tfrac{C_2}{1+G_0C_2}\, v(t) \\ y(t) &= \tfrac{G_0C_1}{1+G_0C_2}\, w(t) + \tfrac{1}{1+G_0C_2}\, v(t) \end{aligned} \tag{7.29}$$

Replacing $u(t)$ in the filtered form of (7.13) yields:

$$\begin{aligned} \epsilon_t^f(k;\underline{\theta}) &= [(G_0 - G)u(t) + v(t)]\tfrac{\bar{H}_k}{H}\, L \\ &= [(G_0 - G)[\tfrac{C_1}{1+G_0C_2}\, w(t) - \tfrac{C_2}{1+G_0C_2}\, v(t)] + v(t)]\tfrac{\bar{H}_k}{H}\, L \\ &= [\tfrac{(G_0-G)C_1}{1+G_0C_2}\, w(t) + \tfrac{1+GC_2}{1+G_0C_2}\, v(t)]\, \tfrac{\bar{H}_k}{H}\, L \end{aligned}$$

The asymptotic parameter estimate is then given by:

$$\underline{\theta}^* = \arg\min_{\underline{\theta}} \bar{J}_{cl}(\underline{\theta}) \tag{7.30}$$

where:

$$\begin{aligned} \bar{J}_{cl}(\underline{\theta}) &= \int_{-\pi}^{\pi} \left|\tfrac{1}{1+G_0C_2}\right|^2 \left[\, |G_0 - G(\underline{\theta})|^2 |C_1|^2 \Phi_w \right. \\ &+ \left. |1 + G(\underline{\theta})C_2|^2 \Phi_v \right]\, |L|^2 \left|\tfrac{\bar{H}_k(\underline{\theta})}{H(\underline{\theta})}\right|^2\, d\omega \end{aligned} \tag{7.31}$$

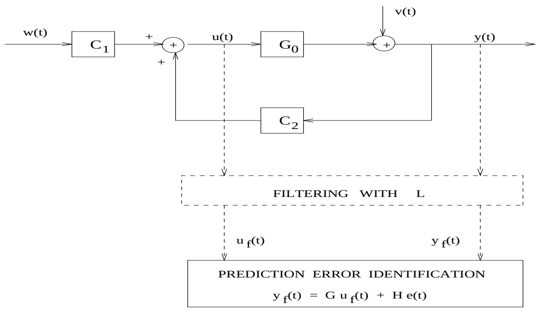

Figure 7.3. *Identification strategy scheme.*

It is interesting to observe that the model fit $(G_0 - G(\underline{\theta}))$ is influenced by the controller, by the reference w and in particular by the gain $\frac{1}{1+G_0C_2}$. The model $G(\underline{\theta})$ will approximate G_0 only if the signal spectrum $|C_1|^2\Phi_w$ dominates the noise spectrum Φ_v within the closed loop bandwidth. It can be noticed that the prefilter L again may be used to shape the modelling fit globally.

Note that there are major differences between (7.17) and (7.31). However, in the last years, several authors have paid attention to the problem of finding an identification criterion relevant for the controller purpose. In the literature, different ideas about the matching the criterion function (7.17) to the criterion (7.31), exist. For example, in *[Hakvoort et al., 1994]*, the identification criterion is defined to be compatible to the control objective when the control criterion is actually minimized by the identification procedure. Thus an optimal identification strategy (see Figure 7.3) is defined, based on a iterative designed filter. The design of the optimal prefilter can be made iteratively based on the estimated model and on the design parameters of the controller (see *[Hakvoort et al., 1994]*, for details). The interpretation

of this scheme is that it includes a weight on the model fit. This weight can be used to get a good fit at those frequencies which are important for the controller. In *[Shook et al., 1991]*, for example, the attention is focused on making the parameter estimation as a dual of the control law, in the predictive adaptive control. This idea is investigated, exemplified and supported by simulations on a linear deterministic Mariner Class Vessel model in the next section. It is based on the fact that the model obtained by identification should be able to provide the best predictions in the range required by controller.

7.4 Derivation of predictive control relevant prefilter

Typically, in adaptive control some version of the RLS recursive least squares identification scheme is used (see Chapter 6). However, if the standard RLS method is used with a predictive controller then, in practice, due to the inevitable structural model-plant mismatch, a number of modifications must be made to insure a robust adaptation and control. The reason is that, while a standard RLS method provides a model which is able to provide only one good ahead step prediction, the predictive controller requires a model of the system, which supplies the best possible output predictions over the entire prediction horizon. This means that the control and the identification objectives are not compatible, the identification does not describe sufficiently well the relevant closed-loop properties of the system. As it was demonstrated in *[Rohrs et al., 1985]* and *[Hakvoort et al., 1994]*, this aspect may even imply that the controller destabilizes the system.

Unlike minimum variance MV or generalized minimum variance controllers GMV, the long range predictive controller (GPC) is less sensitive to high frequency noise caused by model-plant mismatch *[Shook et al., 1992]*. The reason is that the controller considers many predictions in the near future. These predictions used by GPC must come from a model, which have the ability to give good predictions further ahead, and thus a good fit at low frequencies. In order to describe a suitable estimation algorithm for an adaptive GPC controller, an adaptive control objective at time t, proposed in *[Shook et al., 1992]*, can be defined by:

$$J_{ac} = \sum_{k=H_m}^{H_p} [y(t+k) - w(t+k)]^2 \tag{7.32}$$

where the future unknown outputs can be expressed as:

$$y(t+k) = \hat{y}(t+k|t;\underline{\theta}) + \epsilon_{t+k}(k;\underline{\theta}) \tag{7.33}$$

where $\epsilon_{t+k}(k;\underline{\theta})$ is the k-step prediction error. The criterion can be rewritten as:

$$J_{ac} = J_{GPC} + J_{ID} + J_{cross}$$

where:

- J_{GPC} is the usual GPC criterion:

$$J_{GPC} = \sum_{k=H_m}^{H_p} [\hat{y}(t+k|t;\underline{\theta}) - w(t+k)]^2$$

- J_{ID}, the identification objective, is a sum of squared k-step prediction errors:

$$J_{ID} = \sum_{k=H_m}^{H_p} [\epsilon_{t+k}(k;\underline{\theta})]^2 = \sum_{k=H_m}^{H_p} [y(t+k) - \hat{y}(t+k|t;\underline{\theta})]^2$$

- J_{cross} is a cross term, which combines the effects of the identification and the control:

$$J_{cross} = 2 \sum_{k=H_m}^{H_p} [\hat{y}(t+k|t;\underline{\theta}) - w(t+k)]\, \epsilon_{t+k}(k;\underline{\theta})$$

According to *[Shook et al., 1992]*, under weak conditions, the expected value of J_{cross} can be considered to be zero. The average of this cross term is assumed to be zero asymptotically, as the number of observations tends to infinity. Therefore J_{cross} can be neglected.

It is thus observed that in GPC adaptive control, the role of the identification scheme is to minimize the second moment of the prediction errors over the horizon used by the controller, not only over the first ahead step.

As it was pointed out in the previous sections, improved models over the frequency band of importance for the control problem can be obtained by prefiltering data before they are used in identification. Traditionally, based on the guidelines presented in the Section 7.2, the ad-hoc prefiltering is performed using an infinite impulse response (e.g. autoregressive) filter. However, it is more appealing to design an optimal finite impulse response filter L in order to make the estimation objective function J_{ID} dual to the control objective function J_{GPC}. In order to do this, one should focus on an estimation algorithm which would attempt to model the predictions used by the controller. Therefore, a measurable long range predictive identification LRPI cost function, can be defined as the sum of squares of k-step ahead prediction:

$$J_{LRPI} = \sum_{t=1}^{N} \sum_{k=H_m}^{H_p} \epsilon_{t+k}(k;\underline{\theta})^2 \tag{7.34}$$

where $\epsilon_{t+k}(k;\underline{\theta}) = y(t+k) - \hat{y}(t+k|t;\underline{\theta})$.

Implementation of LRPI using various techniques (including Newton-Raphson or Gauss-Newton) is discussed in details in *[Shook et al., 1991]*. However, the implementation of LRPI can also be performed through pre-filtering. To do this, the identification cost function (7.34) is analyzed in terms of the frequency domain implications of long range prediction, as in (7.17). Using Parseval's theorem in the limit as the number of data tends to infinity, LRPI cost function (7.34) can be transformed from time to frequency domain. For this reason, let us consider the system of the general linear structure:

$$A(q^{-1})y(t) = q^{-d}\frac{B(q^{-1})}{A_1(q^{-1})}\,u(t) + \frac{C(q^{-1})}{A_2(q^{-1})}\,\xi(t) \tag{7.35}$$

and the estimated model as:

$$\hat{A}(q^{-1})y(t) = q^{-d}\frac{\hat{B}(q^{-1})}{\hat{A}_1(q^{-1})}\,u(t) + \frac{C(q^{-1})}{A_2(q^{-1})}\,e(t) \tag{7.36}$$

Note, that knowledge of d and of noise model $C(q^{-1})/A_2(q^{-1})$ is assumed. Only fixed noise model is considered. For notational convenience, the dependence on q^{-1} of the polynomials, will be dropped in the following.

The k-step prediction of the output (see Chapter 4) can be expressed as:

$$\hat{y}(t+k|t) = \frac{q^{-d}\hat{B}A_2\mathcal{E}_k u(t+k) + \mathcal{F}_k\hat{A}_1 y(t)}{\hat{A}_1 C} \tag{7.37}$$

where:

$$C = \hat{A}A_2\mathcal{E}_k + q^{-k}\mathcal{F}_k \tag{7.38}$$

Some straight forward calculations on the k-step prediction error yield:

$$\epsilon_{t+k}(k;\underline{\theta}) = \left[y(t+k) - \frac{\hat{B}}{\hat{A}\hat{A}_1}u(t+k-d)\right]\frac{\mathcal{E}_k\hat{A}A_2}{C}$$

$$= \left[\left(\frac{B}{AA_1} - \frac{\hat{B}}{\hat{A}\hat{A}_1}\right)u(t+k-d) + \frac{C}{AA_2}\,\xi(t+k)\right]\frac{\mathcal{E}_k\hat{A}A_2}{C}$$

The long range prediction identification cost function (7.34) can be then expressed as:

$$\begin{aligned} J_{LRPI}(\underline{\theta}) &= \textstyle\sum_{t=1}^{N}\sum_{k=H_m}^{H_p}\Big\{\Big[\Big(\frac{B}{AA_1} - \frac{\hat{B}}{\hat{A}\hat{A}_1}\Big)u(t+k-d) \\ &+ \textstyle\frac{C}{AA_2}\xi(t+k)\Big]\frac{\mathcal{E}_k\hat{A}A_2}{C}\Big\}^2 \end{aligned} \tag{7.39}$$

Applying Parseval's formula, it means that in open loop the minimization of J_{LRPI} is asymptotically ($t \to \infty$) equivalent to the minimization of:

$$\begin{aligned} \bar{J}_{LRPI}(\underline{\theta}) &= \int_{-\pi}^{\pi} \sum_{k=H_m}^{H_p} \left\{ \left| \frac{B}{AA_1} - \frac{\hat{B}}{\hat{A}\hat{A}_1} \right|^2 \Phi_u \right. \\ &+ \left. \left| \frac{C}{AA_2} \right|^2 \Phi_\xi \right\} \left| \frac{\mathcal{E}_k \hat{A} A_2}{C} \right|^2 d\omega \end{aligned} \quad (7.40)$$

It should be observed that the only term which depends on the prediction horizon (required to be within summation) is $\mathcal{E}_k$. Therefore (7.40) can be rewritten as:

$$\begin{aligned} \bar{J}_{LRPI}(\underline{\theta}) &= \int_{-\pi}^{\pi} \left\{ \left| \frac{B}{AA_1} - \frac{\hat{B}}{\hat{A}\hat{A}_1} \right|^2 \Phi_u \right. \\ &+ \left. \left| \frac{C}{AA_2} \right|^2 \Phi_\xi \right\} \left| \frac{\hat{A} A_2}{C} \right|^2 d\omega \sum_{k=H_m}^{H_p} |\mathcal{E}_k|^2 \end{aligned} \quad (7.41)$$

In Section 7.1.3, it was shown that the cost function of the least squares estimation ($k = 1$) with a prefilter L is given by:

$$\bar{J}_{LS}(\underline{\theta}) = \int_{-\pi}^{\pi} \left\{ |G_0 - G(\underline{\theta})|^2 \Phi_u + \Phi_v \right\} \left| \frac{L}{H(\underline{\theta})} \right|^2 d\omega \quad (7.42)$$

If all parts of the cost functions (7.41) and (7.42) are the same (same input, same model, same actual process), i.e.:

$$G_0 = q^{-d} \frac{B}{AA_1} \qquad G(\underline{\theta}) = q^{-d} \frac{\hat{B}}{\hat{A}\hat{A}_1}, \qquad \Phi_v = |\frac{C}{AA_2}|^2 \Phi_\xi \quad (7.43)$$

then the LS cost function (7.42) is identical to the LRPI cost function (7.41) if the following condition holds for all frequencies:

$$|L|^2 = \sum_{k=H_m}^{H_p} |\mathcal{E}_k|^2 \quad (7.44)$$

or equivalently:

$$\boxed{L(q)L(q^{-1}) = \sum_{k=H_m}^{H_p} \mathcal{E}_k(q)\mathcal{E}_k(q^{-1}) \quad (7.45)}$$

Thus the k-step prediction method asymptotically corresponds to a one step ahead prediction error method with a prefilter determined by (7.45). This prefilter is a finite impulse response filter of order $H_p - 1$ and it can easily be found using a standard spectral factorization routine such as that due to Peterka (see e.g. *[Mohtadi, 1988a]*).

Note that C and A_2 are typically design parameters of the controller, so therefore they are imposed by the requirements of the controller. Then based on (7.38) and (7.45), it can be observed that the prefilter $L(q^{-1})$ is model and controller dependent.

The advantage of using an optimal prefilter $L(q^{-1})$ is that the design of the low pass filtering is done automatically. The bandwidth of the prefilter is selected to tune the identification to provide a model with the best predictions in the range required by the controller. Therefore the optimal prefilter can been seen as a promising alternative to the traditional use of ad-hoc prefilters in estimation.

7.5 An iterative design scheme

Looking at the expressions (7.38) and (7.45), it is observed that the prefilter is systematically built in terms of the estimated model structure ($\hat{A}$) and of the controller (H_p, H_m, A_2, C). The optimal prefilter L is not known beforehand and therefore it needs to be determined iteratively. Thus each step, the prefilter is calculating using the previous model and then it is applied to the current data.

There are many variants of the iterative design schemes, e.g. *[Gevers, 1991]*, *[Hakvoort et al., 1994]*, *[Åström and Nilsson, 1994]*. In these schemes, models are fitted in the closed loop with the current controller dictating the emphasis given to different frequency bands.

An iterative design scheme of identification and control is shown in Figure 7.4. It starts by identifying a model from open loop data. The model is then used in the design of the predictive controller. The control signal $u(t)$ is used for collecting a new set of data from the plant G_0, now acting in closed loop. At each step, the prefilter L is designed, based on the previous model and the design parameters of the controller. The iterative aspect is essential, because a model is needed in the controller design and knowledge of the controller is needed in order to identify a good model.

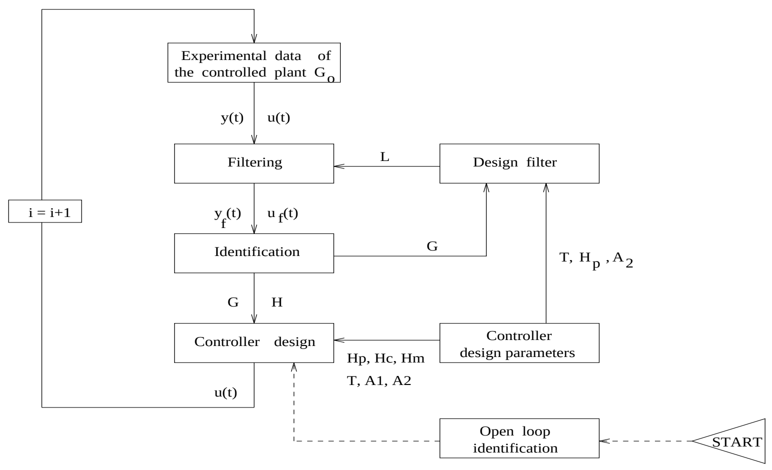

Figure 7.4. *Iterative scheme of identification and control design.*

Until now, it is unknown under what conditions the convergence of the iterative scheme may be guaranteed. In *[Åström and Nilsson, 1994]*, for a pole placement control, it was observed the fact that, the control and identification criteria are compatible, does not necessarily inply that a procedure for iterative estimation and control design converges. A such analysis for the GPC could be a subject for future research.

7.6 Related problems

As it was pointed out in *[Åström and Nilsson, 1994]*, the problem of iterative identification and control design is closely related to other problems such as:

- <u>Model reduction</u> – which is a deductive approach for getting a reduced order model (see Chapter 3), while the iterative design strategy of combining identification and control is an inductive procedure. A low order approximation of a high order model can be identified in the frequency range interesting for the controller. Thus the iterative design procedure gives a reduced order model, which is the most optimal reduced order model for the design of the controller.

- <u>Self tuning control</u> – An iterative design scheme can be also viewed as a self tuning control scheme, where the estimator and the controller are duals to each other. It is well-known that in the case of Minimum Variance Self Tuning Controller *[Åström and Wittenmark, 1973]* and Generalized Minimum Variance scheme *[Clarke and Gawthrop, 1975]*, *[Clarke and Gawthrop, 1979]*, the identification and control objectives are compatible. This is however not the case for other types of self tuning controllers, a fact which may imply catastrophic consequences in the case when unmodelled dynamics are present. These situations can be handled by an iterative design scheme, because an iterative optimal designed prefilter is applied to data before identification in order to ensure compatibility between the identification and the control criteria.

7.7 Illustrative examples

In the present section, the effect of the optimal designed prefilter on the open loop and closed loop identifications is shown by illustrative simulations on a linear model of the Mariner Class Vessel. The identification is performed

on-line. The disturbance model is used as a tool for tuning the regulator behaviour and the stability robustness of the closed loop system. Therefore, in the simulations T will denote C polynomial, as it was pointed out on page 118.

7.7.1 Open loop simulations – analysis

Some properties of the optimal prefilter are illustrated in the following set of examples with open loop identification. The design and analysis of the prefilter are done. The system description and the first order model structure from the simulation Example 7.1 are considered. A low order model is used in order to emphasize the effects due to the undermodelling. Consider on-line noise free identification experiment of 1000 samples ($T_s = 12$ seconds) such that, the variance effects can be neglected. As input signal a zero mean white noise sequence with variance one is used. The observer polynomial is $T = 1 - 0.85q^{-1}$, $A_2 = 1 - q^{-1}$, while the other parameters are the same as in Example 7.1, unless specified.

Example 7.2 *This first example illustrates the connection between the optimal finite impulse response prefilter $L(q^{-1})$ and the traditionally ad-hoc infinite impulse response filter $F(q^{-1})$ (as it was used in Example 7.1). For each H_p, an optimal prefilter $L(q^{-1})$ is found. Thus once $L(q^{-1})$ is available the appropriate ad-hoc filter $F(q^{-1})$ can be obtained via a simple least squares solution to the equation:*

$$\frac{1}{F(q^{-1})} \approx L(q^{-1}) \tag{7.46}$$

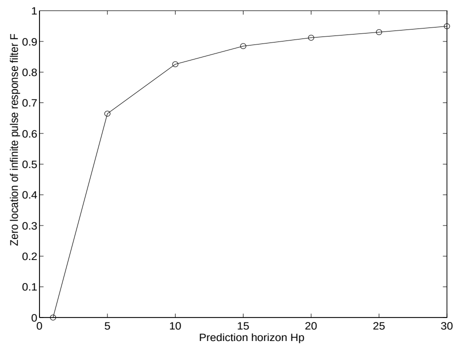

Figure 7.5. *Variation of the infinite impulse response filter with the prediction horizon H_p.*

Figure 7.5 illustrates the variation of the zero location of the first order polynomial $F(q^{-1}) = 1 - f\,q^{-1}$ with the prediction horizon H_p. For example, when $H_p = 30$, an improved low frequency model is obtained (as it will be seen later), and the estimated ad-hoc prefilter is $F(q^{-1}) = 1 - 0.949\,q^{-1}$. This confirms the choice of the ad-hoc prefilter used in Example 7.1. ◇

Example 7.3 *An open loop identification experiment is now performed for different values of prediction horizons H_p. The step responses of the second order plant (7.22) and of the first order estimated models, for different values of H_p, are shown in Figure 7.6.*

It is observed that the result of the one-step-ahead estimation algorithm ($H_p = 1$) is a model that gives totally misguiding predictions further in the future. This can in turn lead to a controller with very bad performance or even an unstable closed loop (see e.g. [Rohrs et al., 1985] *and*

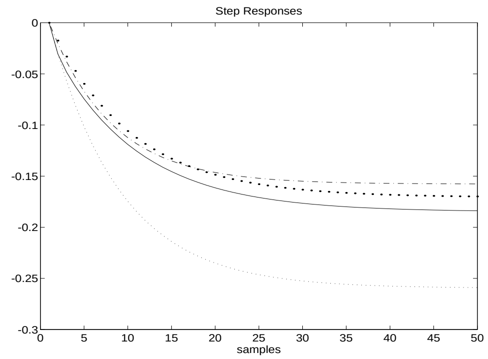

Figure 7.6. *Step responses as function of H_p: $H_p = 1$, dotted line; $H_p = 15$, dash-dotted line; $H_p = 30$, pointed line; By continuous line, the plant.*

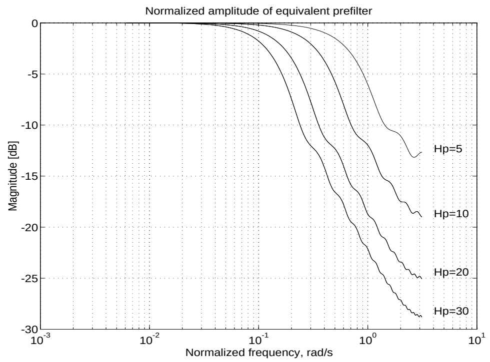

Figure 7.7. *Amplitude gain of optimal prefilters as function of H_p.*

[Gevers, 1991]). Note that, as H_p increases the estimated response matches more closely the step response of the plant. However, because of the undermodelling, there will always be some discrepancy between the estimated

and the true system. It must be stressed that, using a prefilter the effect of unmodelled dynamics may be reduced but not totally removed. Hence, there is still a need for an observer polynomial in the control design. However, it will not be as critical as with standard RLS estimation. The amplitude gain of the optimal prefilters are shown in Figure 7.7. The gain has been normalized to give a DC-gain of 0 dB. The optimal prefilter is a low pass filter. It is observed that the prefilter $L(q^{-1})$ depends significantly on the prediction horizons. Its normalized cut-off frequency decreases when H_p increases.

The influence of the observer polynomial $T(q^{-1})$ on the prefilter $L(q^{-1})$ can also be analyzed. For example, keep the prediction horizon constant as $H_p = 20$, and vary the zero location of $T(q^{-1})$ between 0.75 and 0.95. The amplitude gain of the prefilter is shown in Figure 7.8. It seems that $L(q^{-1})$ slightly depends on the observer polynomial $T(q^{-1})$. ◇

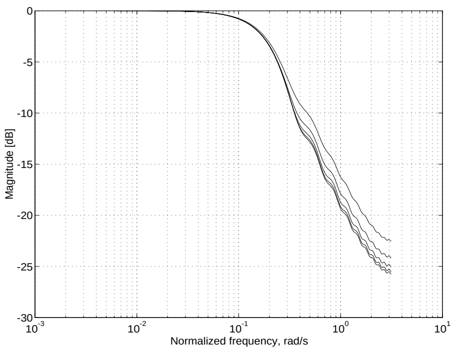

Figure 7.8. *Amplitude gain of the optimal prefilters as function of the zero location of the observer polynomial T. Zero location= 0.75, 0.8, 0.85, 0.9, 0.95 from bottom to top.*

Example 7.4 *In order to get an idea about the influence of the prediction horizon H_p and of the observer polynomial $T(q^{-1})$ on the estimated parameters in the first order model:*

$$G(q^{-1}) = \frac{b\, q^{-1}}{1 + a\, q^{-1}} \tag{7.47}$$

a set of on-line identification experiments is done on the second order system (7.22) for different values of H_p and of the zero location z of the first order observer polynomial:

$$T(q^{-1}) = 1 - \; z\, q^{-1}$$

The optimal prefilter $L(q^{-1})$ is computed for each situation.

To judge the estimated parameters as a function of both H_p and z, the coefficients of the Nomoto model (presented on page 226):

$$G^N_{r\delta}(q^{-1}) = \frac{-0.0197\, q^{-1}}{1 - 0.8926\, q^{-1}} \tag{7.48}$$

are used for comparison. In this example, Nomoto coeffiecients are regarded as "optimal" due to the well-known fact that, Nomoto model is a proper low order approximation of the system, in the frequency range interesting for controller, i.e. for low frequencies.

Figure 7.9 shows the variation of the estimated parameters $\hat{a}$ and $\hat{b}$ for different values of H_p and z. The proper Nomoto parameters ($a = -0.8926$,

$b = -0.0197$) are illustrated by the constant shaded planes, independent of H_p and z. It is observed that these constant planes are intersected by the planes related to the variation of parameters in the first order model. The experiment gives an idea of how to choose the prediction horizon H_p and zero location z of the observer polynomial $T(q^{-1})$ in order to get a good low frequency representation. In both graphics in Figure 7.9, an analysis of the intersections can be done. It is observed that a unique set of H_p and z, which simultaneously satisfies both graphics, cannot be chosen. They must be selected as a compromise which approximatively satisfies both intersections. In this example, it can be observed that an optimal range for H_p is 20–30, while for z it is 0.85 – 0.95. ◇

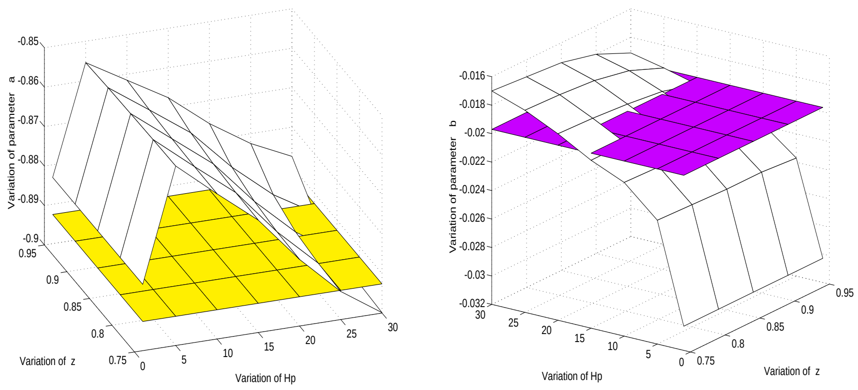

Figure 7.9. *Variation of the estimated parameters $\hat{a}$ and $\hat{b}$ as a function of H_p and zero location z of the observer polynomial T.*

The analysis in *[Rohrs et al., 1984]* has demonstrated that the effect of unmodelled dynamics can also be diminished by a proper choice of the sampling period. It has been observed that when a large sampling period

(which still satisfies Shannon's theorem) is used, the poles and the zeros of the unmodelled dynamics are so close to each other that their effects are almost cancelled. Thus, in discrete systems, the sampling interval can be viewed as an additional design parameter, which can increase or decrease the tolerance to unmodelled dynamics.

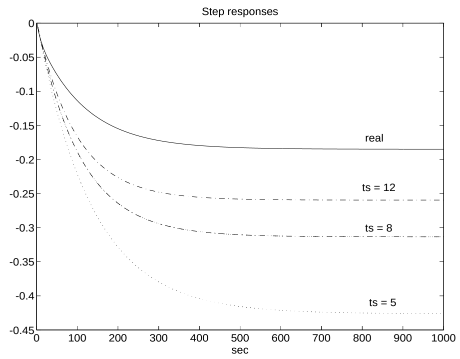

Figure 7.10. *Influence of the sampling period on the step response.*

Example 7.5 *The influence of the sampling interval on the effect of the unmodelled dynamics, in the case of a ship model, is analyzed. In Figure 7.10, the step responses of the reduced estimated model (7.47), for different sampling periods T_s, are compared with that for the true process (7.22). In Example 7.3 it was illustrated that by using a large H_p the effects of unmodelled dynamics are significantly reduced. Thus, in order to outline the influence of sampling period, alone on the unmodelled dynamics, $H_p = 1$ is used in this example. In Figure 7.10, it is observed that by using an increased sampling interval T_s, the effect of the unmodelled dynamics decreases and* *thus the step response of the estimated model becomes closer to the real one.* ◇

Example 7.6 *Now, summarizing the conclusions from the previous examples, a final open loop simulation is done using some "optimal" design parameters, e.g.* $H_p = 25$, $T(q^{-1}) = 1 - 0.95\, q^{-1}$ *and* $T_s = 12$ *seconds. As input is used again a zero mean white noise with variance 1. The on-line estimated parameters are illustrated in Figure 7.11. It is observed that, the estimates are close to the Nomoto coefficients in (7.48).* ◇

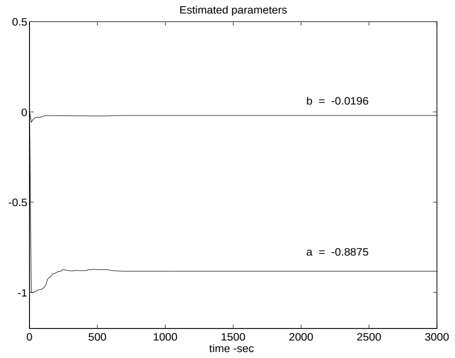

Figure 7.11. *Estimated parameters when* $H_p = 25$, $T = 1 - 0.95\, q^{-1}$ *and* $T_s = 12$ *seconds.*

7.7.2 Closed loop simulations – analysis

In order to illustrate the effects of unmodelled dynamics in self tuning control and how they can be decreased, some closed loop simulation examples

for the steering of a Mariner Class Vessel are presented in this section. The considered self tuning controller is based on recursive least squares estimation with prefiltered data and the generalized predictive controller (described in Chapter 4). The certainty equivalence principle is used.

The following transfer function from rudder angle to heading angle ($T_s = 12$ seconds) is considered :

$$G_{o,\psi\delta}(q^{-1}) = \frac{-0.2053\, q^{-1} - 0.0434\, q^{2} + 0.0791\, q^{-3}}{(1 - \; q^{-1})(1 - 1.1214 q^{-1} + 0.1979 q^{-2})} \tag{7.49}$$

As a reduced order model for (7.49) a second order model is chosen:

$$G_{\psi\delta}(q^{-1}) = \frac{b\, q^{-1}}{(1 - q^{-1})(1 + a\, q^{-1})} \tag{7.50}$$

Note, that there is a mismatch between the real system (7.49) and the model (7.50). This aspect is common for the most real processes, because for practical reasons, controllers are designed based on low order models.

The simulations are conducted for a period of 250 samples. The first 50 samples are used for an open loop identification, where as input signal is applied a white noise sequence for ensuring the persistence of excitation property. The others 200 samples are exploited for closed loop identification.

The following parameters are considered for the controller:

$$\begin{array}{lcllcl} H_m & = & 1 & H_c & = & 2 \\ P_n & = & 1 & P_d & = & 1 \\ Q_n & = & 1 & Q_p & = & 1 \\ H_p & = & 25 & \phi & = & 1-\ q^{-1} \\ \rho & = & 10^{-4} & T & = & 1-0.95\,q^{-1} \end{array} \tag{7.51}$$

The initial values of the RLS estimator were chosen as:

$$\begin{array}{lcl} \underline{\hat{\theta}} & = & \begin{bmatrix} 0 & 0 \end{bmatrix}^T \\ \boldsymbol{P}(0) & = & 10\,\boldsymbol{I} \end{array} \tag{7.52}$$

where $\boldsymbol{I}$ denotes the identity matrix.

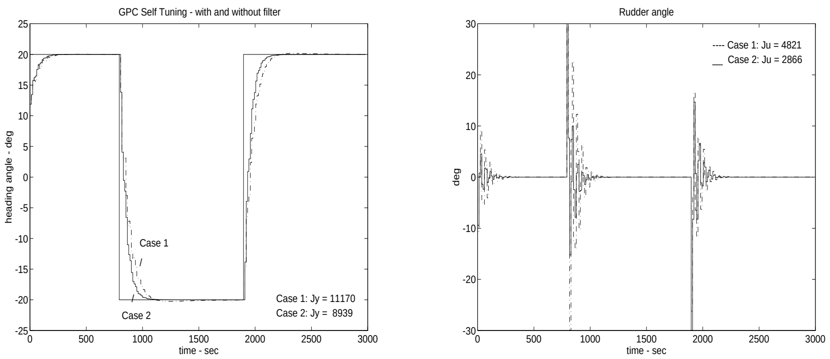

Figure 7.12. *GPC self tuning controller for Case 1 and Case 2.*

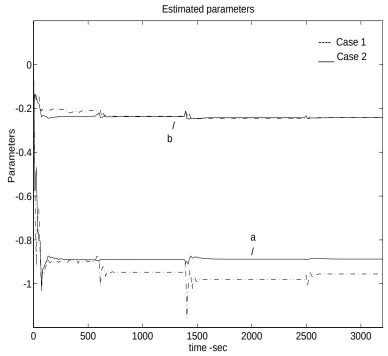

Figure 7.13. *Estimated model parameters.*

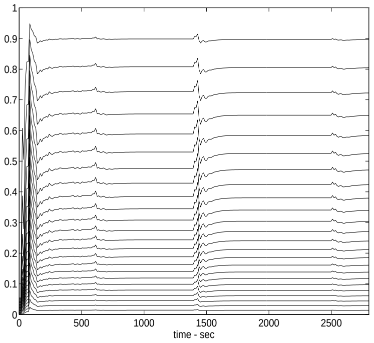

Figure 7.14. *Prefilter coefficients.*

Example 7.7 *It is known that a long range predictive controller is not very sensitive to high frequency noise, neither actual noise nor caused by model - plant mismatch. The high frequency sensitivity is reduced because the controller considers many predictions in the near future. However, in adaptive predictive control, due to the incompatibility of the identification and control cost functions, a standard RLS method can be catastrophical.*

In the present example two cases are studied:

- *Case 1: where the identification is performed without prefilter.*
- *Case 2: where the identification is performed with an optimal prefilter.*

To measure the quality of tracking and control performance, respectively, the following loss functions are defined:

$$
\begin{aligned}
J_y &= (\underline{y}-\underline{w})^T\,(\underline{y}-\underline{w}) = \sum_{i=50}^{250}(y_i-w_i)^2 \\
J_u &= \underline{u}^T\,\underline{u} = \sum_{i=50}^{250}u_i^2
\end{aligned}
\tag{7.53}
$$

where y_i is the heading angle, u_i is the rudder angle, and w_i is the reference signal at sample i. The reason for discounting the first 50 samples in the loss functions is that, only the self tuning control phase is wanted to be investigated.

Figure 7.12 illustrates that the performance of the control is improved in Case 2: J_u and J_y are smaller than in Case 1. The closed loop system in Case 2 is faster than in Case 1, plus that it requires a smaller input energy. It should be mentioned that Figure 7.12 illustrates only the closed loop responses, namely the last 200 samples simulation.

The convergence of the parameter estimates is shown in Figure 7.13. It is observed an improvement of convergence rate in the case when the prefiltering of data is used. The figure shows that there exists a difference for the estimate $\hat{a}$ between Case 1 and Case 2. It is observed that, for Case 1 the identified model can even become unstable (as it is the moment of the first changing of the reference). This is not the situation of Case 2. However, for both cases, GPC controller keeps the closed loop stable.

The coefficients of the optimal prefilter $L(q^{-1})$, are plotted in Figure 7.14. It should be noted that in Figure 7.13 and Figure 7.14, the situation from the open-loop identification is also presented, namely the first 50 samples. The convergence speed for the parameters in the prefilter is limited by the convergence speed of the model parameters, since the polynomials $\mathcal{E}_k$ depend on the estimated model through the Diophantine equation (7.38).

The step responses of the real system (7.49) and of the estimated reduced order models (7.50), with and without prefilter, are shown in Figure 7.15. This figure helps to illustrate that the model from Case 2 (with prefilter) provides a better prediction of the step response of the actual system.

The results of the simulations can be found in Tables 7.1 and 7.2, where the values of the loss functions J_y and J_u for different $T(q^{-1})$ polynomials and prediction horizons H_p, respectively, for Case 1 and Case 2 are shown. It is noticed that the influence of the observer polynomial $T(q^{-1})$ and of the prediction horizon H_p on the controller performances is similarly in closed loop simulations with that from open loop simulations. The cost functions J_u and J_y decreases when the zero location of $T(q^{-1})$ and the prediction horizon H_p are increased. A comparative analysis of Case 1 and Case 2 outlines the improved performances in Case 2. An additional observation is that the bigger H_p and the closer to one the zero location of $T(q^{-1})$ are, the more visible the difference between Case 1 and Case 2 is.

In order to outline the fact that a combined iterative approach of identification and control design can lead to results that are better than those obtained from open loop considerations alone, the step responses of the real system (7.49) and of the estimated models (7.50) in the open loop and in the closed loop situation are illustrated in Figure 7.16. Data are prefiltered *before identification. The observer polynomial and the prediction horizon $T(q^{-1}) = 1 - 0.95q^{-1}$ and $H_p = 25$, respectively are considered. An improvement of the response model, when the estimation is combined with the closed loop can be observed.* ◇

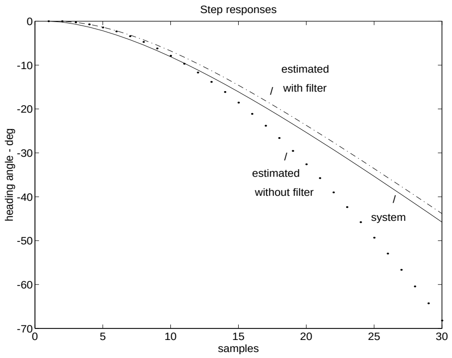

Figure 7.15. *Step responses.*

	J_y		J_u	
$T(q^{-1})$	Case 1	Case 2	Case 1	Case 2
1	$3.448 \cdot 10^4$	$1.581 \cdot 10^4$	$1.316 \cdot 10^5$	$1.429 \cdot 10^5$
$1 - 0.5\,q^{-1}$	$1.293 \cdot 10^4$	9050	6412	6129
$1 - 0.8\,q^{-1}$	$1.174 \cdot 10^4$	8953	5405	3157
$1 - 0.95\,q^{-1}$	$1.117 \cdot 10^4$	8939	4821	2866

Table 7.1. *Values of the loss functions J_y and J_u for Case 1 and Case 2, for different $T(q^{-1})$ – when $H_p = 25$.*

	J_y		J_u	
H_p	Case 1	Case 2	Case 1	Case 2
5	6850	6587	$1.805 \cdot 10^4$	$1.215 \cdot 10^4$
10	7373	6880	9376	7410
15	8508	7552	6277	4732
20	9732	8290	5536	3363

Table 7.2. *Values of the loss functions J_y and J_u for Case 1 and Case 2, for different H_p – when $T(q^{-1}) = 1 - 0.95\,q^{-1}$.*

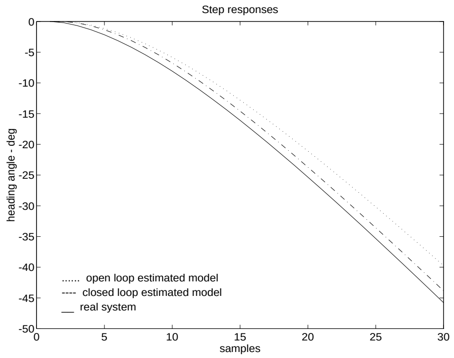

Figure 7.16. *Step responses of the models estimated in open loop and closed loop.*

Example 7.8 *As it was demonstrated in Chapter 5, one of the most attractive feature of predictive control is that it has the ability to handle the process constraints in a natural and systematic way during the design of the controller. It was shown that the ad-hoc strategy, even though it is appealing by its simplicity, does not necessarily guarantee an acceptable performance. In*

	Ad-hoc		DGP	
	Case1	Case2	Case1	Case2
J_u	$2.741 \cdot 10^4$	$2.426 \cdot 10^4$	$2.214 \cdot 10^4$	$1.89 \cdot 10^4$
J_y	$3.267 \cdot 10^4$	$3.248 \cdot 10^4$	$3.175 \cdot 10^4$	$3.084 \cdot 10^4$

Table 7.3. *Values of the loss functions J_u and J_y for Case 1 and Case 2, for Ad-hoc and DGP strategies.*

this final example, the DGP and ad-hoc control strategies (described in Chapter 5) are combined with on-line identification with and without prefiltering of data. Thus, the two cases, Case 1 and Case 2, defined in Example 7.7, are again considered. The design parameters of the controller are:

$$H_m = 1, \quad H_c = 2, \quad H_p = 5, \quad T = 1 - 0.95q^{-1}, \quad \rho = 10^{-4} \tag{7.54}$$

The others parameters are the same as defined on page 251. Both level and rate constraints on the controller output are considered:

$$\begin{array}{rclcrcl} u^{min} & = & -30\ [deg] & & u^{max} & = & 30\ [deg] \\ \Delta u^{min} & = & -3.5\ [deg/s] & & \Delta u^{max} & = & 3.5\ [deg/s] \end{array} \tag{7.55}$$

The results of the simulations can be found in Table 7.3. It is observed that the best results are obtained when DGP algorithm works togheter with prefiltering of data (i.e. DGP with Case 2). The worst situation is when the ad-hoc control strategy and no prefiltering of data (i.e. ad-hoc with Case 1) are combined. By comparing the results of Table 7.3, it can be noticed that the control loss function J_u is visible decreased from Case 1 to Case 2, from

Ad-hoc to DGP, while the output loss function J_y is not affected too much. The simulation of the Case2 with DGP is shown in Figure 7.17.

Thus, the goal of this final example was to point out the important role played by prefilter in a constrained predictive self tuning control. ◇

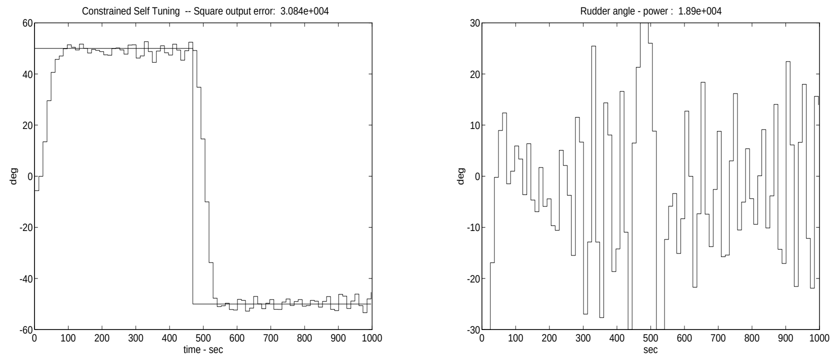

Figure 7.17. *GPC self tuning controller with DGP and prefiltering of data.*

7.8 Summary

The chapter deals with the interplay between identification and control in adaptive control, with special attention given to the long range predictive feedback design. Some theoretical results *[Ljung, 1987]* and *[Gevers, 1991]* concerned with the tuning of the identification towards the objective for which the model is to be used, are recalled.

The joint problem of identification and control design methods can be simply settled by performing identification in closed loop and by using a proper data prefilter in the estimation. This optimal filter is not known beforehand and needs to be determined iteratively. Explicit expressions for the data filter have been given for the long range predictive control design. The design prefilter is related to the design parameters of the controller and to the estimated model. In the proposed iterative identification and control scheme, each identification step uses the previously designed controller to obtain new data from the plant and to filter these data. The associated identification problem is solved by means of a filtered prediction error method.

The theoretical aspects are supported by illustrative simulations on a linear model of the Mariner Class Vessel. In order to emphasize the effects due to undermodelling, low order models and noise-free simulations are used.

This chapter has demonstrated the importance of using properly prefiltered data in the identification, as a stage of adaptive control, in the presence of structural model-plant mismatch.

Chapter 8

Conclusions

The present thesis is addressed to predictive control and identification. The work includes analysis and simulations on a stochastic model of the Mariner Class Vessel. The study was limited to ship motions in the horizontal plane only. As disturbances, wind, waves and currents are considered. The sub-conclusions of each chapter may be consulted for more detailed information. The major conclusions are summarized below.

The main objective of this thesis was to enhance insight into predictive controller design, to contribute to an improved understanding of how predictive controllers operate and why they can be successfully used in practical applications. The goal was to examine and improve the robustness of predictive controllers towards unmodelled dynamics and saturation constraints. For

these reasons, the theory of predictive control was presented in an extended fashion, based on a more general linear model structure and cost function.

In many practical situations, for the analysis and the implementation of a controller, it is often desired to approximate the high order system model by a low order model in such a way that the control relevant dynamics are preserved in the low order model. It was shown that the reduced order model can be obtained through analytical approaches (model reduction techniques) or empirical approach (system identification). Two model reduction techniques, based on singular values, were described in Chapter 3: Balanced Truncation Approximation (BTA) and Hankel Norm Approximation (HNA). They were investigated and found to have desirable properties as model reduction tools. The size of singular values express how well the lower order model approximates the system. The theoretical properties of these two methods have been studied. Simulations on a Mariner Class Vessel have demonstrated the effectiveness of these model reduction techniques for obtaining successful control design. In the studied example, it was concluded that H_∞ norm satisfied the same error bounds both for BTA and HNA. As BTA usually implies more simple calculations than HNA, BTA seems to be more appealing in practice.

Based on a low order model, obtained using the BTA method, a predictive track keeping system for a stochastic Mariner Class Vessel model was designed in Chapter 4. The synthesis of the ship's control system was derived by the minimization of the mean squares prediction errors of the ship's deviation from the desired track. The practical properties of the predictive controller, as the ability e.g. 1) to handle non-minimum phase and unstable systems, 2) to deal with pre-scheduled reference trajectories, have been illustrated via simulations. The robustness towards unmodelled dynamics was improved by a proper choice of the so-called observer polynomial.

The most attractive feature of predictive control is that, until now, it is the only methodology that can handle process constraints in a natural and systematic way during the design of the controller. The ability to cope with constraints may be an important robustness property of the controller. However, despite its practical attraction, this issue was unfortunately not often addressed in the literature. In the present thesis, the optimal constrained problem has been formulated and solved iteratively using Rosen's gradient projection method. The analysis of the optimal constrained predictive control was motivated, in particular, by the problem present in the steering machine of a ship, i.e. that the rudder angle and the speed of the rudder are physically limited. The described algorithm DPG, Dedicated Projection Gradient, has been presented and analyzed in details in Chapter 5. The effectiveness of this algorithm compared to the well-known ad-hoc strategy was illustrated via simulations on a Mariner Class Vessel linear model. The hard constraints, taken into account in the analysis, were level and rate constraints on the input control. It was shown that, opposite to DGP algorithm, the ad-hoc control strategy, based on a simple truncation of the calculated signal to the limits, may lead to poor performances or even to instability of the system. The influences of a set of design parameters on the performance of the closed-loop system were investigated, for each particular set of constraints. It was observed that the use of a larger prediction horizon caused the performance of the closed-loop system to increase and the number when DGP was activated, to decrease. The initialization point, which should be in the feasible region, had a great effect on the number of iterations required to find the optimal solution with a particular accuracy.

It was concluded that a very small value of accuracy did not necessarily result in a better performance, but only in a larger number of iterations.

In Chapter 6, a brief overview of parameter identification was given. The overview was based on the recursive least squares algorithm which is suitable in the case of on-line estimation. Further, methods of tracking slow time-varying parameters have been discussed. The discussion focused on the introduction of forgetting factors methods, where the old information is gradually discounted.

It is known that, the presence of unmodelled dynamics can cause difficulties in parameter estimation. For a long time, the identification strategy has been viewed as a search for "the true system" rather than as a methodology whose aim was to produce an "approximation of reality". However, in the last years, the identification task has been turned into tuning the model towards the objective for which the model is to be used.

In the presence of plant unmodelled dynamics it is convenient not to rely on the usual RLS method in the case when it is combined with a predictive controller. The reason is that RLS provides a model which is able to give only one good step ahead prediction, while the predictive controller requires a model, which must supply the best possible output predictions over the prediction horizon. In Chapter 7, the neccessary interplay between identification and control, especially predictive control, was investigated. The joint problem of identifcation and control design methods has been done by performing identification in closed loop and by using a proper data prefilter in the estimation. An equivalent prefilter was designed based on the idea that the identification criterion and the control criterion should be dual. An iterative scheme was proposed. A prefilter was iteratively designed, based on the design parameters of the controller and on the previous estimated model. The theoretical aspects were supported by illustrative simulations on a Mariner Class Vessel linear model. It was shown that better performances were obtained by using an optimalprefiltering of data before the estimation. The approach seems to be very practicable and an appealing alternative to the usual ad hoc prefiltering.

There are always some open investigations at the end of a research work. For example, future research might be directed towards analysis of the stability of linear systems with control saturation constraints. Another topic for future investigation can also be the study of the convergence of the constrained adaptive control and of the iterative identification/control schemes. It could be also interesting to extend the DGP algorithm, in the presence of output constraints.

Appendices

A Mariner Class Vessel 269

B Normalization 275

C Model reduction – DATA 277

D LQG controller design 281

E Curve route generation 289

F Optimal step for the unconstrained problem 293

Appendix A

Mariner Class Vessel

The simulations described in the present thesis are based on the nonlinear Blanke model for a ship from the Mariner class, a 16622 m^3 displacement, 160 meters long ship, with a designed speed of 7.71 m/s (15 knots). This class of vessels was designed in the mid-sixties and is not in use any longer. The choice was made, however, because the hydrodynamic data available is the most complete set published for a particular vessel *[Chislett and Tejsen, 1965]*.

The main data and dimensions of the Mariner Class Vessel are:

- Length over all of ship $L_{oa} = 171.8$ [m].
- Maximum beam $B = 23.17$ [m].

- Draught $T = 6.86$ [m].
- Length of the ship between perpendiculars $L = 160$ [m].
- Mass of the ship $m = 17050$ [tons].
- Displacement $W = 16622$ $[m^3]$.
- Center of gravity $X_G = 3.70$ [m].
- Nominal Speed $U = 15$ [knots] ($u_o = 7.716$ [m/s]).
- Projected front area of the ship $A_f = 416.4$ $[m^2]$.
- Projected side area of the ship $A_s = 1540.1$ $[m^2]$.
- Nominal propeller revolutions $n = 7.151$ $[rad/s]$.
- $E = \frac{m+X_{vr}}{m} = 1.25$.
- $P = \frac{X_{uu}}{m} = -2.7419 \cdot 10^{-4}$ $[m^{-1}]$.
- $G = \frac{(1-t)T_{nn}}{m} = 1.2608 \cdot 10^{-3}$ $[m/rad^2]$.
- $H = \frac{(1-t)T_{nv}(1-w)}{m} = -8.7261 \cdot 10^{-4}$ $[rad^{-1}]$.
- Water density $\rho_w = 1025$ $[kg/m^3]$.
- Air density $\rho_a = 1.23$ $[kg/m^3]$.
- Gravity acceleration $g = 9.81$ $[m/s^2]$.

The nonlinear Blanke model is:

$$\begin{aligned} \dot{u} &= E\, v\, r + P\, u^2 + G\, n^2 + H\, n\, u + \frac{1}{m} X_{ext} \\ \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} &= \boldsymbol{A}\, u_o \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \boldsymbol{B}\, u_0^2\, \delta + \boldsymbol{C} \begin{bmatrix} v^3 \\ r v^2 \end{bmatrix} + \boldsymbol{D} \begin{bmatrix} Y_{ext} \\ N_{ext} \end{bmatrix} \end{aligned} \tag{A.1}$$

The states of the considered model are:

- u - the surge speed [m/s]
- v - the sway velocity [m/s]
- r - the yaw rate [deg/s]
- ψ - the heading angle [deg]

Inputs to the system are:

- n is the propeller velocity in rad/s. In this work n is considered to be constant at the nominal value ($n = 7.151$ [rad/s]). The nominal speed is $U = 7.71$ [m/s] (15 knots).
- δ - the rudder angle [deg]
- external disturbances forces which act on the ship: X_{ext}, Y_{ext}, N_{ext}. The type of disturbances causing additional signals in the system (current, wind, wave) will be taken into account.

The matrices used in (A.1) are:

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad \boldsymbol{B} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix}$$
$$\boldsymbol{C} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ 0 & 0 \end{bmatrix}, \qquad \boldsymbol{D} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ 0 & 0 \end{bmatrix} \tag{A.2}$$

where the coefficients:

$$
\begin{aligned}
a_{11} &= \frac{(I_z - N_{\dot{r}})Y_v + (Y_{\dot{r}} - mX_G)N_v}{u_o\Delta} \\
a_{12} &= \frac{(I_z - N_{\dot{r}})(Y_r - mu_o) - (mX_G - Y_{\dot{r}})(N_r - mX_G u_o)}{u_o\Delta} \\
a_{21} &= \frac{(m - Y_{\dot{v}})N_v - (mX_G - N_{\dot{v}})Y_v}{u_o\Delta} \\
a_{22} &= \frac{(m - Y_{\dot{v}})(N_r - mX_G u_o) - (mX_G - N_{\dot{v}})(Y_r - mu_o)}{u_o\Delta} \\
b_1 &= \frac{(I_z - N_{\dot{r}})Y_\delta - (mX_G - Y_{\dot{r}})N_\delta}{u_o^2\Delta} \\
b_2 &= \frac{(m - Y_{\dot{v}})N_\delta - (mX_G - N_{\dot{v}})Y_\delta}{u_o^2\Delta} \\
c_{11} &= \frac{(I_z - N_{\dot{r}})Y_{vvv} + (Y_{\dot{r}} - mX_G)N_{vvv}}{\Delta} \\
c_{12} &= \frac{(I_z - N_{\dot{r}})Y_{rvv} + (Y_{\dot{r}} - mX_G)N_{rvv}}{\Delta} \\
c_{21} &= \frac{(N_{\dot{v}} - mX_G)Y_{vvv} + (m - Y_{\dot{v}})N_{vvv}}{\Delta} \\
c_{22} &= \frac{(N_{\dot{v}} - mX_G)Y_{rvv} + (m - Y_{\dot{v}})N_{rvv}}{\Delta} \\
d_{11} &= \frac{(I_z - N_{\dot{r}})}{\Delta}; \quad d_{12} = \frac{(Y_{\dot{r}} - mX_G)}{\Delta} \\
d_{21} &= \frac{(N_{\dot{v}} - mX_G)}{\Delta}; \quad d_{22} = \frac{(m - Y_{\dot{v}})}{\Delta}
\end{aligned} \tag{A.3}
$$

Dimensional	Value	Unit
a_{11}	$-4.5378 \cdot 10^{-3}$	$\frac{1}{m}$
a_{12}	$-3.1060 \cdot 10^{-1}$	1
a_{21}	$-1.3068 \cdot 10^{-3}$	$\frac{1}{m^2}$
a_{22}	$-1.2973 \cdot 10^{-2}$	$\frac{1}{m}$
b_1	$1.1758 \cdot 10^{-3}$	$\frac{1}{m}$
b_2	$-6.2659 \cdot 10^{-5}$	$\frac{1}{m^2}$
c_{11}	$-4.292 \cdot 10^{-3}$	$\frac{s}{m^2}$
c_{12}	1.335	$\frac{s}{m}$
c_{21}	$9.1270 \cdot 10^{-5}$	$\frac{s}{m^3}$
c_{22}	$-5.094 \cdot 10^{-2}$	$\frac{s}{m^2}$
d_{11}	$3.02 \cdot 10^{-8}$	$\frac{1}{kg}$
d_{12}	0	$\frac{1}{m\ kg}$
d_{21}	0	$\frac{1}{m\ kg}$
d_{22}	$2.17 \cdot 10^{-11}$	$\frac{1}{m^2\ kg}$

Table A.1. *Coefficients of the Blanke nonlinear model for the Mariner Class Vessel.*

$$\Delta = (m - Y_{\dot{v}})(I_z - N_{\dot{r}}) - (mX_G - N_{\dot{v}})(mX_G - Y_{\dot{r}})$$

In the Table A.1 the values of the dimensional coefficients of the model (A.1) for the Mariner Class Vessel are presented.

Nomoto model is:

$$G_{r\delta}(s) = \frac{K(1+sT_3)}{(1+sT_1)(1+sT_2)} \approx \frac{K}{(1+sT)} \tag{A.4}$$

where $T_1 = 7.91$ seconds, $T_2 = 116.28$ seconds, $T_3 = 18.55$ seconds for the considered ship. Based on these values, the Nomoto coefficients are computed as:

$$\begin{aligned} K &= K = -0.1830 \\ T &= T_1 + T_2 - T_3 = 105.6 \end{aligned} \tag{A.5}$$

Appendix B

Normalization

It is customary to normalize the models of the ship. The most commonly system used for normalization of ship models is the "prime" system, in which the length unit is the length of the ship L, the time unit is L/U, where U is the nominal ship speed, and the mass unit is $\frac{1}{2}\rho L^3$, where ρ is the mass density of water.

Prime nondimentional forces, moments , velocities and accelerations denoted by prime (see Table B.1), are defined accordingly in the following way:

- Nondimensional Force $= \frac{\text{Force}}{\frac{1}{2}\rho L^2 U^2}$
- Nondimensional Moment $= \frac{\text{Moment}}{\frac{1}{2}L^3 U^2}$
- Nondimensional Velocity $= \frac{\text{Velocity}}{U}$

- Nondimensional Angular velocity $= \frac{\text{Angular velocity}}{U} L$
- Nondimensional Acceleration $= \frac{\text{Acceleration}}{U^2} L$
- Nondimensional Angular Acceleration $= \frac{\text{Angular Acceleration}}{U^2} L^2$

When motion studies are considered, even the variables of the equations are often normalized, as for example:

$$Y' = Y_v' v' + Y_{\dot{r}}' \dot{r}' + Y_{ur}' u' v' + Y_{|v|v}' |v|' v' + Y_{|v|r}' |v|' r' \tag{B.1}$$

The dimensionless quantities are listed in the Table B.1:

$m' = m/(\frac{1}{2}\rho L^3)$	$t' = t/(L/U)$	$x_G' = x_G/L$	$u' = u/U$
$v' = v/U$	$\dot{v}' = \dot{v}/(\frac{U^2}{L})$	$r' = r/(\frac{U}{L})$	$\dot{r}' = \dot{r}/(\frac{U^2}{L^2})$
$Y_v' = Y_v/(\frac{1}{2}\rho L^3)$	$Y_{\dot{v}}' = Y_{\dot{v}}/(\frac{1}{2}\rho L^3)$	$Y_r' = Y_r/(\frac{1}{2}\rho L^3 U)$	$Y_{\dot{r}}' = Y_{\dot{r}}/(\frac{1}{2}\rho L^3 U)$
$N_v' = N_v/(\frac{1}{2}\rho L^3 U)$	$N_{\dot{v}}' = N_{\dot{v}}/(\frac{1}{2}\rho L^4)$	$N_r' = N_r/(\frac{1}{2}\rho L^4 U)$	$N_{\dot{r}}' = N_{\dot{r}}/(\frac{1}{2}\rho L^5)$
$Y_{vv}' = Y_{vv}/(\frac{1}{2}\rho L^2)$	$Y_{vr}' = Y_{vr}/(\frac{1}{2}\rho L^3)$	$N_{vv}' = N_{vv}/(\frac{1}{2}\rho L^3)$	$N_{vr}' = N_{vr}/(\frac{1}{2}\rho L^4)$
$Y_\delta' = Y_\delta/(\frac{1}{2}\rho L^2 U^2)$	$N_\delta' = N_\delta/(\frac{1}{2}\rho L^3 U^2)$	$I_x' = I_x/(\frac{1}{2}\rho L^5)$	$I_z' = I_z/(\frac{1}{2}\rho L^5)$

Table B.1. *Prime nondimentionalized coefficients.*

Another normalization, the "bis" system, was proposed by *[Norrbin, 1970]*. In this sytem the length unit is L, the time unit is $\sqrt{L/g}$, where g is the acceleration of gravity, and the mass unit is m. Details on this normalization system can be found in *[Källström, 1979]*, *[Fossen, 1994]*.

Appendix C

Model reduction – DATA

```
In the model reduction examples,
Matlab implementation has been made.

The state space high order model has the matrices:

Ah = [-0.1923  -0.0037        0        0        0        0
       1.0000        0        0        0        0        0
            0        0  -0.3602  -0.6569   0.0000   0.0000
            0        0   1.0000        0        0        0
       0.0003        0   0.0523   0.0953  -0.0103   0.0001
      -0.0006        0   0.0297   0.0542   0.0007  -0.0372
       0.0000        0  -0.0047        0   0.0000  -0.0010
            0        0        0        0        0        0


            0        0
```



```
            0        0
            0        0
            0        0
      -0.0814   0.0030
      -2.4941   0.0038
      -0.1046        0
       1.0000       0];


Bh =[    0          Wh=[0.7422        0
         0                   0        0
         0                   0   0.4133
         0                   0        0
         0                   0  -0.0600
    0.0762                   0  -0.0341
   -0.0041                   0        0
        0];                  0       0];


Ch =[ 0    0    0    0    0    0    1    0
      0    0    0    0    0    0    0    1];


Conditional number of the controllability grammian is:  1.4423e+007
Conditional number of the observabilty grammian is:  1.1999e+013

---------------------------------------------------------------------
---------------------------------------------------------------------

Minimal realzation is defined by the matrices:

Am  = [-0.1521  -0.0027  -0.0001
        1.0000        0        0
             0   1.0000       0];
Bm = [ 1
       0
```

```
      0];

Cm = [-0.0041   -0.0003    0.0000];

Conditional number of controllability grammian is:  4.3717 e+006;
Conditional number of observability grammian is:    7.7595 e+010;

----------------------------------------------------------------------
----------------------------------------------------------------------

The balanced realization is defined by the matrices:

Abs = [ -0.0138   0.0222   0.0010
         0.0222  -0.1270  -0.0100
         0.0010  -0.0100  -0.0113];

Bbs = [ 0.0480
       -0.0418
       -0.0017];

Cbs = [-0.0480   0.04184   0.0017];

Conditional number of Abs is:  14.66;
Conditional number of controllabilty/observability grammian
is: 6.6684 e+002;

Controllability and observability grammians diagonal and equal to:

 [ 0.08358   0.00000  -0.00000
   0.00000   0.00689  -0.00000
  -0.00000  -0.00000   0.00013 ];

------------------------------------------------------------------
------------------------------------------------------------------

The parameters of the polynomials of the control law in the case
```



```
when the controller is designed on the reduced order model:

R = [ 0.9038  -0.5398  -0.3640];
V = [-4.7049   6.5922  -2.4265];
S = [-1.0000   0.4007];

The parameters of the polynomials of the control law in the case
 when the controller is designed on the high order model:

R=[0.9014  8.3986  -21.7210  10.5303  8.3274  -8.1710  1.6521
   0.1065  -0.0242];

V=[37.1024  -186.9801  358.4648  -342.3435   169.7287   -38.7187
   1.6144  0.5868   -0.0543];

S=[-1.0000   0.4007];
```

Appendix D

LQG controller design

In this appendix, the design of an incremental LQG controller and the principle of gain scheduling are briefly reviewed.

LQG controller is known to have a set of practical advantages *[Grimble, 1986]*:

- LQG controller offers a guarantee of stability for all the values of the control weighting parameters, when the plant is known, even though for an unstable open loop system or non-minimum phase systems.

- A knowledge of the time delay through the plant is not necessarily required a priori, because it does not enter explicitly into the calculations.

- Stochastic disturbances to the system are taken into account in the optimal control design.

- An integral action may be introduced by appropriate choice of the cost function.

As the LQG controller belongs to the class of model-based controller design concepts, a model of the process is explicitly used to design the controller:

$$A(q^{-1})y(t) = q^{-d}\, B(q^{-1})u(t) + C(q^{-1})e(t) + l \tag{D.1}$$

where $y(t)$ is the output signal, $u(t)$ is the control signal, $e(t)$ is a zero mean white Gaussian noise sequence with constant variance, while l can be related to the mean of the disturbances or as level due to the operating point. This model can be achieved e.g. as a result of the model reduction. $A(q^{-1})$, $B(q^{-1})$ and $C(q^{-1})$ are polynomials in the backward shift operator, q^{-1}:

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a} \\ B(q^{-1}) &= b_1 + \ldots + b_{n_b} q^{-n_b} \\ C(q^{-1}) &= 1 + c_1 q^{-1} + \ldots + c_{n_c} q^{-n_c} \end{aligned} \tag{D.2}$$

The classical servo-problem formulation is to design a controller in order to ensure that the system output $y(t)$ follows the reference signal as close as possible. Assume that the desired transfer function is described as:

$$G_d(q^{-1}) = \eta\, \frac{B(q^{-1})}{P(q^{-1})} \tag{D.3}$$

where $P(q^{-1})$ denotes the polynomial which embodies the optimal closed loop pole locations and η is a constant used to correct to unity the DC – amplification, $\eta = \frac{P(1)}{B(1)}$.

The LQG control low is obtained by minimization of the criterion:

$$J = \lim_{N\to\infty} E\{\frac{1}{N}\sum_{i=1}^{N}[(y(t)-\overline{w})^2 + \rho(u(t)-\bar{u})^2]\} \tag{D.4}$$

where ρ is a positive scalar and it represents the only one tuning parameter, while $\overline{w}$, y and $\bar{u}$ represent the desired reference, the output of the system and the mean of the control signal, respectively.

It is well-known that the LQG controller implies solving of a spectral factorization and of a Diophantine equation. The implementation of the spectral factorization can be done by using the algorithm proposed by *[Mohtadi, 1988a]*. The Diophantine equation can be solved by using the Sylvester's method *[Poulsen, 1995]*.

The steady state error in the closed loop system can be eliminated by enforcing an integral action in the controller, namely by minimizing the criterion:

$$J = \lim_{N\to\infty} E\{\frac{1}{N}\sum_{i=1}^{N}[(y(t)-\overline{w})^2 + \rho\Delta u(t)^2]\} \tag{D.5}$$

where:

$$\Delta u = u(t) - u(t-1)$$

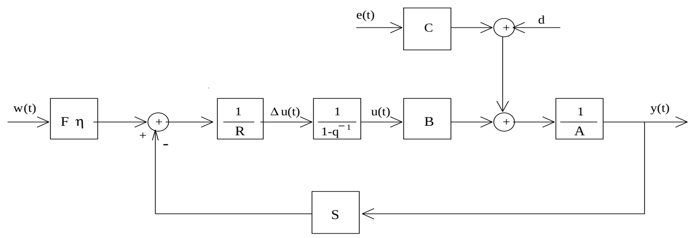

Figure D.1. *LQG controller with incremental action.*

Note that in the case when the load disturbances d is known precisely, the incremental action is not necessary. Then an additive quantity must be inserted in the control law *[Poulsen, 1985]*, in order to eliminate the load d.

Figure D.1 illustrates the LQG controller with an incremental action. The incremental algorithm proposed by *[Rasmussen and Pedersen, 1983]* is used:

1. Solve the spectral factorization:

$$P(q^{-1})P(q) = \rho A^{+}(q^{-1})A^{+}(q) + B(q^{-1})B(q) \tag{D.6}$$

   where $A^{+}(q^{-1}) = (1-q^{-1})A(q^{-1})$. The closed loop polynomial $q^{n}P(q^{-1})$ should have all zeroes inside the stability area.

2. Solve the Diophantine equation:

$$P(q^{-1})\,C(q^{-1}) = A^{+}(q^{-1})\,R(q^{-1}) + B(q^{-1})\,S(q^{-1}) \tag{D.7}$$

   where the order of the polynomials R and S are:

$$\begin{aligned} n_r &= n_b - 1 \\ n_s &= \max(n_a+1, n_p+n_c-n_b) \end{aligned} \tag{D.8}$$

3. The control law is obtained as:

$$R(q^{-1})\,\Delta\,u(t) = V(q^{-1})\,w(t) - S(q^{-1})\,y(t) \tag{D.9}$$

where the signals $u(t)$, $y(t)$, $w(t)$ are absolute values and $V(q^{-1}) = \eta\,C(q^{-1})$. The polynomial R is not necessarily monic.

DC amplification equal to one in the transfer function is assured by the condition: $\eta = \frac{P(1)}{B(1)}$. In Figure D.1 the optimal observer dynamic is $F = C$, assuming that the C polynomial has zeroes inside the unit circle *[Åström and Wittenmark, 1989]* .

The effect of the incremental action is that at low frequencies ($\omega \rightarrow 0$ and $q \rightarrow 1$) the disturbances (in particular, constant offsets) are exactly canceled. The disadvantage of the incremental control consists in a possible deterioration of the closed loop performances.

The principle of gain scheduling

In many practical cases it is known how the dynamic of a process changes with the operating conditions of the process. In these situations, to control the process, it is very popular to use a linear regulator whose parameters are changed as a function of operating conditions in a preprogrammed way. This strategy is known as Gain Scheduling Control.

Gain scheduling control is a special type of nonlinear feedback which, based on the measurements of the operating conditions of the process, is a good way to compensate for known nonlinearities. The regulator parameters are calculated at a number of operating conditions, in advance, using some suitable design method. Thus the controller is adapted in open loop as a

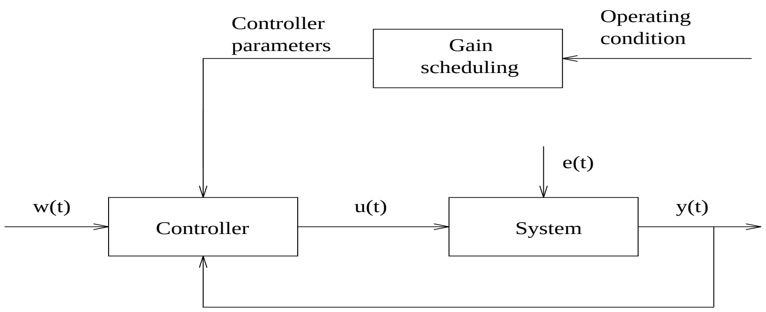

Figure D.2. *Gain scheduling strategy.*

function of operating conditions. There is not a feedback from the performance to the closed loop system. The principle of gain scheduling is illustrated in Figure D.2 (adopted from *[Åström and Wittenmark, 1989]*). There are known two types of gain scheduling: 1) model gain scheduling 2) controller gain scheduling. The last one is often used in practice.

Due to the fact that no estimation of parameters is required, the gain scheduling control has the great advantage that the parameters of the regulator can be changed very quickly when the dynamic of the process varies. However, the fact that the controller parameters are changed in open loop is a disadvantage, because there does not exist a feedback to compensate an incorrect schedule. Other drawbacks of the gain scheduling are that it is time consuming and it is not always easy to get good indication of the operation conditions.

A simple exemplification of the fact that in Gain Scheduling Control the parameters of the controller are computed in advance – as a function of the operating points – is illustrated in Figure D.4 and Figure D.3. These figures contain the parameters of a reduced order controller for different relative directions of wind and waves[1]. The surge speed is assumed constant and equal to the nominal speed $U = 7.71$ m/s, while the mean wind is constant equal to $\overline{V} = 15$ m/s. The reduced order controller is designed on a reduced order model (as it was discussed in Section 3.6), obtained in each stationary point corresponding to each operating condition. From these graphics, it can be observed that the influence of wind and waves directions on the parameters of the controller is insignificantly. Therefore, in the control simulation described in Section 3.6, a Gain Scheduling Controller is not interesting.

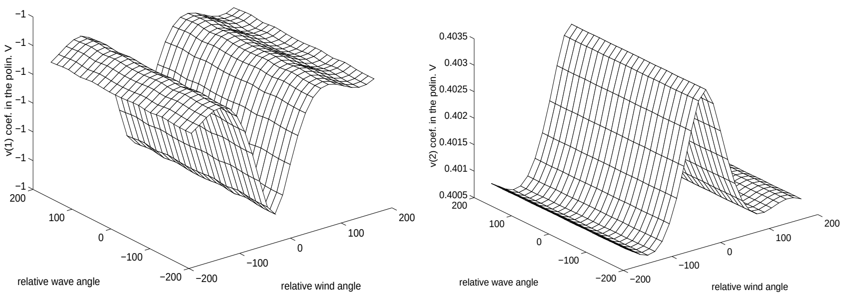

Figure D.3. *Variation of the parameters of polynomial $V(q^{-1})$ with respect to the relative direction of wind and waves.*

[1]The relative directions of wind and waves were defined in Chapter 2.

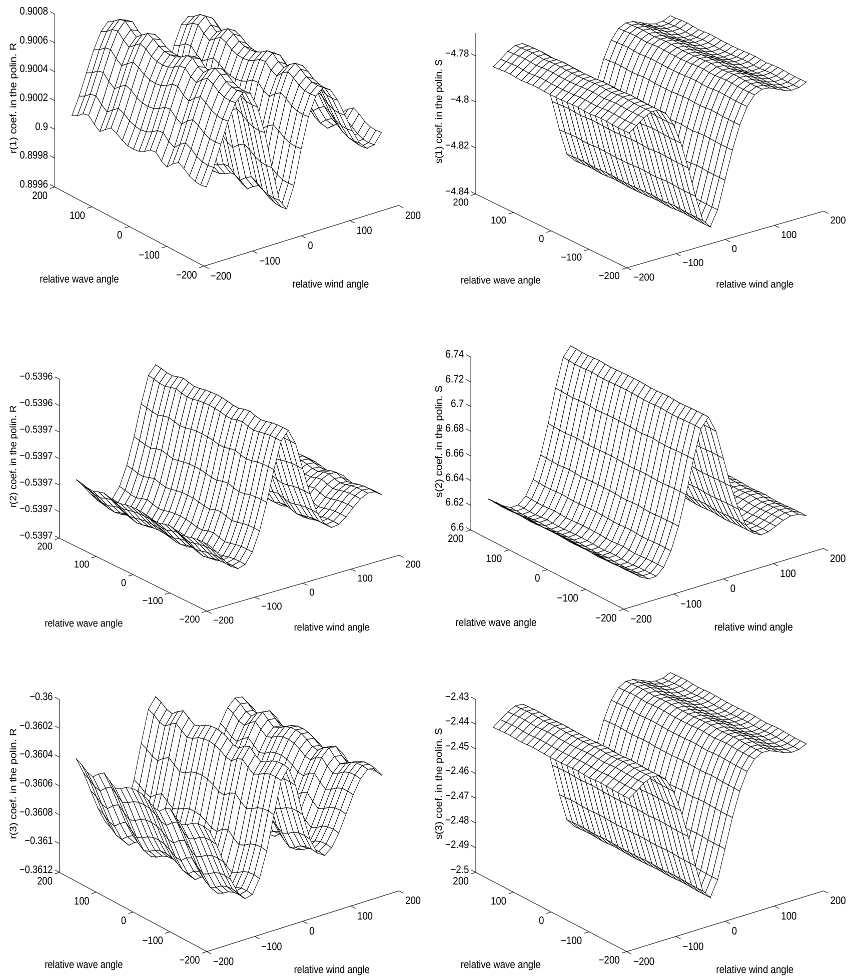

Figure D.4. *Variation of the parameters of the polynomials $R(q^{-1})$ and $S(q^{-1})$ with respect to the relative direction of wind and waves.*

Appendix E

Curve route generation

In Figure E.1 is sketched how the curve trajectory $\widetilde{MN}$ is designed when the coordinates of turning point B, the radius R and the angles Φ_1 and Φ_2 are known. The generation of the curve trajectory is uniquely determined. The tracking on the circular arcs is reduced to the tracking on small straight lines, because the circular arc can be divided into small arc lengths. They should be small enough to be able to approximate them to be tangents to the curve.

In Figure E.1 it is assumed that $\Phi_2 > \Phi_1$. The case when $\Phi_2 < \Phi_1$ is similar. $OM \perp AB$, $ON \perp BC$. Supposing that the arc is divided into n small equal pieces, then the following expressions exist:

$$\begin{aligned} \alpha &= \Phi_2 - \Phi_1 \\ \alpha_i &= \frac{\alpha}{n}. \end{aligned} \tag{E.1}$$

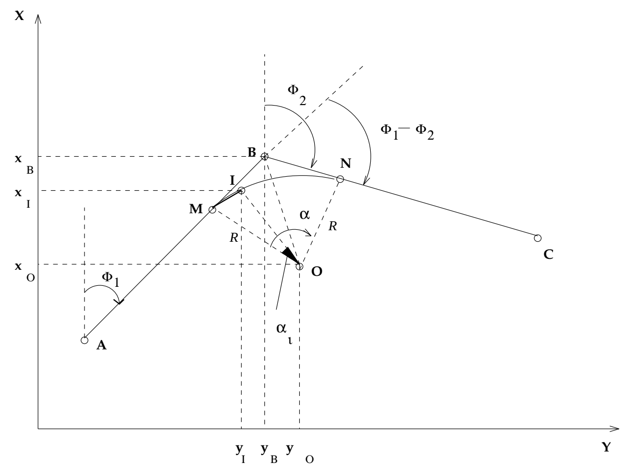

Figure E.1. *Curve route generation.*

After some simple geometrical manipulations, the coordinates of the centre of the circle are found to be:

$$\begin{aligned} X_o &= X_B - R\,\frac{\sin(\frac{\Phi_1+\Phi_2}{2})}{\cos(\frac{\Phi_2-\Phi_1}{2})} \\ Y_o &= Y_B + R\,\frac{\cos(\frac{\Phi_1+\Phi_2}{2})}{\cos(\frac{\Phi_2-\Phi_1}{2})} \end{aligned} \tag{E.2}$$

Once the coordinates of the center are known, then the coordinates of an intermediary point I on the arc, corresponding to the angle α_i, are determined by:

$$\begin{aligned} X_i &= X_o + R\sin(\Phi_1 + \alpha_i) \\ Y_i &= Y_o - R\cos(\Phi_1 + \alpha_i) \end{aligned} \tag{E.3}$$

Using these relations, the curve trajectory is obtained recursively.

Appendix F

Optimal step for the unconstrained problem

Consider the quadratic function:

$$f(\underline{x}) = \frac{1}{2}\,\underline{x}^T\,\boldsymbol{Q}\,\underline{x} - \underline{x}^T\,\underline{b} \tag{F.1}$$

In order to compute the gradient of (F.1), the following derivative formula is used:

$$\partial(\underline{x}^T\,\boldsymbol{A}\,\underline{y})/\partial\underline{v} = \underline{x}^T\,\boldsymbol{A}\,(\partial\underline{y}/\partial\underline{v}) + \underline{y}^T\,\boldsymbol{A}^T\,(\partial\underline{x}/\partial\underline{v}) \tag{F.2}$$

Thus:

$$\begin{cases} \frac{1}{2}(\underline{x}^T\,\boldsymbol{Q}\,\underline{x})/\partial\underline{x} & = & \frac{1}{2}\,\underline{x}^T\boldsymbol{Q} + \frac{1}{2}\,\underline{x}^T\boldsymbol{Q}^T\, = \,\underline{x}^T\boldsymbol{Q} \\ \partial(\underline{x}^T\,\underline{b})/\partial\underline{x} & = & \underline{b}^T \end{cases} \tag{F.3}$$

As $\boldsymbol{Q}$ is a symmetric matrix, then the expression of the gradient of (F.1) (column vector) can be written as:

$$\underline{\nabla f} = \boldsymbol{Q}\,\underline{x} - \underline{b} \tag{F.4}$$

The optimal step size $\lambda^*_{opt,n}$ in:

$$\underline{x}_{n+1} = \underline{x}_n + \lambda^*_{opt,n}\,\underline{d}_n$$

is determined thus $f(\underline{x}_{n+1})$ is minimized:

$$f(\underline{x}_n + \lambda_n\,\underline{d}_n) = \frac{1}{2}\,(\underline{x}_n + \lambda_n\,\underline{d}_n)^T\,\boldsymbol{Q}\,(\underline{x}_n + \lambda_n\,\underline{d}_n) - (\underline{x}_n + \lambda_n\,\underline{d}_n)^T\,\underline{b}$$

Differentiating $f(\underline{x}_n + \lambda^*_{opt,n}\,\underline{d}_n)$ with respect to $\lambda^*_{opt,n}$, by using (F.2), it is obtained that:

$$\begin{aligned} \frac{\partial f}{\partial \lambda_n} &= \frac{1}{2}\,(\underline{x}_n + \lambda_n\,\underline{d}_n)^T\,\boldsymbol{Q}\,\underline{d}_n + \frac{1}{2}\,(\underline{x}_n + \lambda_n\,\underline{d}_n)^T\,\boldsymbol{Q}\,\underline{d}_n - \underline{d}_n^T\,\underline{b} \\ &= (\underline{x}_n + \lambda_n\underline{d}_n)^T\boldsymbol{Q}\,\underline{d}_n - \underline{d}_n^T\,\underline{b} \end{aligned} \tag{F.5}$$

Setting the $\frac{\partial f}{\partial \lambda_n}$ to zero and using (F.4), the resulting equation can be solved for $\lambda^*_{opt,n}$:

$$\lambda^*_{opt,n} = -\frac{(\underline{\nabla f})^T_n \; \underline{d}_n}{\underline{d}^T_n \; \boldsymbol{Q} \; \underline{d}_n}$$

Bibliography

[Abkowitz, 1964] Abkowitz, M. (1964). Lectures on ship hydrodynamics - steering and maneuverability. *Report*, No. Hy-5.

[Abkowitz, 1980] Abkowitz, M. (1980). Measurement of hydrodynamic characteristics from ship maneuvering trails by system identification. *SNAME Transactions*, 88:283–318.

[Adamjan et al., 1978] Adamjan, V., Arov, D., and Krein, M. (1978). Hankel block matrices and related extension problems. *Amer. Math. Translations*, 2(111):113–156.

[Åge, 1971] Åge, C. (1971). Wind coefficients for nine ship models. Technical Report A-3, Hydro and Aerodynamics Laboratory, Lyngby Denmark.

[Amerongen, 1985] Amerongen, J. (1985). Applications of adaptive control to ship steering. *IFAC Identification and System Parameter Estimation.*

[Amerongen, 1982] Amerongen, J. V. (1982). *Adaptive steering of ships.* PhD thesis, Delft University of Technology, The Netherlands.

[Amerongen and Land, 1980] Amerongen, J. V. and Land, E. (1980). An adaptive autopilot for trackkeeping. *Ship Operation Automation*, 3:105–115.

[Åström, 1980] Åström, K. J. (1980). Why use adaptive techniques for steering large tankers ? *Int. J. Control*, 32(4):689–708.

[Åström, 1993] Åström, K. J. (1993). Matching criteria for control and identification. In *Preprints European Control Conference, ECC*, pages 249–251, Groningen, Holland.

[Åström and Källström, 1976] Åström, K. J. and Källström, C. G. (1976). Identification of ship steering dynamics. *Automatica*, 12:9–22.

[Åström and Nilsson, 1994] Åström, K. J. and Nilsson, J. (1994). Analysis of a scheme for iterated identification and control. In *SYSID'94*, volume 2, pages 171–176, Copenhagen, Denmark.

[Åström and Wittenmark, 1973] Åström, K. J. and Wittenmark, B. (1973). On self tuning regulators. *Automatica*, 9:185–199.

[Åström and Wittenmark, 1989] Åström, K. J. and Wittenmark, B. (1989). *Adaptive Control.* Addison Wesley.

[Bazaraa and Shetty, 1979] Bazaraa, M. and Shetty, C. (1979). *Nonlinear programming. Theory and Algorithms.* John Wiley and Sons.

[Berkelom et al., 1974] Berkelom, V., Trägårdth, P., and Dellhag, A. (1974). Large tankers - wind coeffiecients and speed loss due to wind and sea. *The Royal Institution of Naval Architescts*, 117(41):41–55.

[Bierman, 1977] Bierman, G. (1977). *Factorization methods for discrete sequential estimation.* Academic Press, New York.

[Blanke, 1981] Blanke, M. (1981). *Ship propulsion losses related to automatic steering and prime mover control.* PhD thesis, Servolaboratory, Technical University of Denmark.

[Camacho, 1993] Camacho, E. F. (1993). Constrained generalized predictive control. *IEEE Transactions on Automatic Control*, 38(2):327–331.

[Chislett and Tejsen, 1965] Chislett, M. and Tejsen, J. S. (1965). Planar motion mechanism tests and full scale steering and manoeuvering predictions for a mariner class vessel. Technical Report Hy-6, Hydro and Aerodynamics Laboratory, Lyngby Denmark.

[Christensen, 1992] Christensen, A. (1992). *Models od control design-modelling and validation in ship control.* PhD thesis, Denmark Technical University.

[Christensen and Blanke, 1986] Christensen, A. and Blanke, M. (1986). A linearized state-space model in steering and roll of a high speed container ship. Technical Report 86-D-574, Servolaboratory, Tehnical University of Denmark.

[Clarke and Gawthrop, 1975] Clarke, D. W. and Gawthrop, P. J. (1975). Self tuning controller. *Proc. IEE - D*, 122(9):929–934.

[Clarke and Gawthrop, 1979] Clarke, D. W. and Gawthrop, P. J. (1979). Self tuning control. *Proc. IEE - D*, 126(6):633–640.

[Clarke et al., 1987] Clarke, D. W., Mohtadi, C., and Tuffs, P. S. (1987). Generalized predictive control-Part 1. The Basic Algorithm. *Automatica*, 23(2):137–148.

[Cutler and Ramaker, 1980] Cutler, C. and Ramaker, B. (1980). Dynamic matrix control-a computer control algorithm. *JACC*.

[Davenport, 1971] Davenport, A. C. (1971). Wind structure and wind climate. technical, Boundary Layer Wind Tunnel Laboratory, Fac. of engineering Science - Canada.

[Enns, 1984a] Enns, D. (1984a). Model reduction with balanced realizations: An error bound and a frequency weighted generalization. *Proceedings of the 23 Conference on Decision and Control*, pages 127–131.

[Enns, 1984b] Enns, D. F. (1984b). *Model reduction for control system design.* PhD thesis, Standford University.

[Fernando and Nicholson, 1983] Fernando, K. V. and Nicholson, H. (1983). On the structure of balanced and other principal representations of SISO systems. *IEEE Transaction Automatic Control*, Ac-28(2):228–230.

[Fortescue et al., 1981] Fortescue, T. R., Kershenbaum, and Ydstie, B. (1981). Implementation of self tuning regulators with variable forgetting factors. *Automatica*, 17:831–835.

[Fossen, 1994] Fossen, T. (1994). *Guidance and Control of Ocean Vehicles.* John Wiley and Sons Ltd.

[Garcia and Morshedi, 1984] Garcia, C. E. and Morshedi, A. M. (1984). Solution of the dynamic matrix control problem via quadratic programming. *Proceedings of the Conference of the Canadian Industrial Computing Society*, pages 13.1–13.3.

[Gevers, 1991] Gevers, M. (1991). Connecting identification and robust control: a new challenge. In *Proc. 9th IFAC Symp. on Identification and System Parameter Estimation*, pages 1–10, Budapest, Hungary.

[Gevers, 1993] Gevers, M. (1993). Towards a joint design of identification and control? In *2nd European Control Conference*, pages 111–151, Groningen, Netherlands.

[Gill and Murray, 1974] Gill, P. and Murray, W. (1974). *Numerical methods for constrained optimalization*. Academic Press Inc., 1 edition.

[Glover, 1984] Glover, K. (1984). All optimal hankel-norm approximations of linear multivatiable systems and their l^∞ error bounds. *Int. J. Control*, 39(6):1115–1193.

[Glover, 1987] Glover, K. (1987). Model reduction: a tutorial on hankel-norm methods and lower bounds on l2 errors. *IFAC 10th*, page 293.

[Grimble, 1986] Grimble, M. J. (1986). Controllers for LQG self tuning applications with coloured measurement noise and dynamic costing. *Proc. IEE - D*, 133(1):19.

[Hakvoort et al., 1994] Hakvoort, R. G., Schrama, R. J. P., and Van den Hof, P. M. J. (1994). Approximate identification with closed-loop performance criterion and application to LQG feedback design. *Automatica*, 30(4):679–690.

[Hansen et al., 1995] Hansen, A. D., Christensen, A., and Poulsen, N. K. (1995). Model reduction and gain scheduling control. Technical Report 12, IMM, DTU.

[Hjalmarsson et al., 1994] Hjalmarsson, H., Gevers, M., Bruyne, F., and Leblond, J. (1994). Identification for control: closing the loop gives more accurate controllers. Technical report, Department Electrical Engineering, University of Linköping.

[Hjalmarsson and Ljung, 1992] Hjalmarsson, H. and Ljung, L. (1992). Estimating model variance in the case of undermodelling. *IEEE Transactions on Automatic Control*, 37(7):1004–1008.

[Holzhüter, 1990] Holzhüter, T. (1990). A high precision track controller for ships. In *Preprints 11th IFAC*, pages 118–123, Tallinn.

[Isherhood, 1972] Isherhood, R. M. (1972). Wind resistance of merchant ships. *TRINA*, 115:327–338.

[Jensen, 1991] Jensen, M. R. (1991). *Adaptive autopiloter til skibe*. IMSOR-DTU.

[Kailath, 1980] Kailath, T. (1980). *Linear Systems*. Prentice Hall, Inc.

[Källström, 1979] Källström, C. G. (1979). *Identification and Adaptive Control Applied to Ship Steering*. Ph.D. report, Lund Institute of Technology.

[Keyser and Cauwenberghe, 1985] Keyser, R. and Cauwenberghe, A. (1985). Extended prediction self adaptive control. *Proceedings 7th IFAC Symposium on Identification and System Parameter Estimation*, pages 1255–1260.

[Keyser et al., 1988] Keyser, R., Velde, V. D., and Dumortier, F. (1988). A comparative study of self-adaptive long range predictive control methods. *Automatica*, 24(2):149–163.

[Kinnaert, 1989] Kinnaert, M. (1989). Adaptive generalized predictive controller for mimo systems. *Int. J. Control*, 50(1):161–172.

[Lauritsen, 1994] Lauritsen, M. B. (1994). Robust adaptive control. Master's thesis, IMSOR-Denmark Technical University.

[Ljung, 1987] Ljung, L. (1987). *System identification: Theory for the User.* Prentice Hall.

[Luenberger, 1984] Luenberger, D. (1984). *Linear and Nonlinear Programming.* Addison-Wesley.

[Van der Klugt, 1987] Van der Klugt, P. (1987). *Rudder roll stabilization.* PhD thesis, Delft Tecnical University, Rotterdam.

[Middleton et al., 1988] Middleton, R., Goodwin, G., and Hill, D. J. (1988). Design issues in adaptive control. *IEEE Transactions on Automatic Control*, AC-33(nr. 1):50–57.

[Mohtadi, 1988a] Mohtadi, C. (1988a). *Numerical algorithms in self-tuning control.* London,UK.

[Mohtadi, 1988b] Mohtadi, C. (1988b). On the role of prefiltering in parameter estimation and control. *Adaptive Control Strategies for Industrial Use*, 137:121–144.

[Moore, 1981] Moore, B. (1981). Principal component analysis in linear systems:controllability, observability and model reduction. *IEEE Transactions on the Automatic Control*, 26(1):17–39.

[Mort and Linkens, 1981] Mort, N. and Linkens, D. A. (1981). Self-tuning controllers for surface ship course keeping and manoeuvring. *IEE Control Engineering*, 15:296–308.

[Mustafa and Davidson, 1993] Mustafa, D. and Davidson, T. (1993). Balanced approximation can be optimal. *Systems and Control Letters*, 20:397–400.

[Nomoto, 1957] Nomoto, K. (1957). On the Steering Qualities of Ships. *International Shipbuilding Progress*, 4(35).

[Norrbin, 1970] Norrbin, N. H. (1970). Theory and observation on the use of a mathematical model for ship maneuvering in deep and confined waters. *8th Symposium on Naval Hydrodynamics.*

[Parkum, 1992] Parkum, J. E. (1992). *Recursive Identification of time varying systems.* PhD thesis, Ph.D. afhandling, nr. 57, DTU, Denmark.

[Pernebo and Silvermann, 1979] Pernebo, L. and Silvermann, L. (1979). Balanced systems and model reduction. *Proc. 18th. Conf. Decision Contr.*, 12-16:865–866.

[Poulsen, 1985] Poulsen, N. K. (1985). *Robust self tuning controllers.* Ph.D. report, IMSOR, Building 321, DK-2800 Lyngby.

[Poulsen, 1995] Poulsen, N. K. (1995). *Stokastik Adaptiv Regulering*, volume 1. IMM-DTU.

[Price and Bishop, 1974] Price, W. G. and Bishop, R. E. D. (1974). *Probabilistic Theory of Ship Dynamics.* Chapman and Hall, London.

[Rasmussen and Pedersen, 1983] Rasmussen, H. and Pedersen, T. S. (1983). Self tuning controllers applied to steam superheaters. *Proc. IASTED Symposium, Applied Control and Identification, Copenhagen*, pages 1–5.

[Reid and Moore, 1982] Reid, R. and Moore, J. (1982). A steering control system to minimize propulsion losses of high speed containerships. *Journal of Dynamic Systems, Measurement and Control*, 104. Part 1 - System Dynamics.

[Reid and Parent, 1982] Reid, R. and Parent, P. (1982). Comparison of optimal linear regulators applied to minimization of steering losses of a containership in a seaway using frequency domain representations. *Journal of Dynamic Systems, Measurement and Control*, 104.

[Richalet et al., 1978] Richalet, J., Rault, A., Testud, J. L., and Papon, J. (1978). Model predictive heuristic control: applications to industrial processes. *Automatica*, 14:413–428.

[Rivera et al., 1992] Rivera, D., Pollard, J., and Garcia, C. (1992). Control relevant prefiltering: A systematic design approach and case study. *IEEE Transaction. on Automatic Control*, 37(7):964–974.

[Robinson and Clarke, 1991] Robinson, B. D. and Clarke, D. W. (1991). Robustness effects of a prefilter in Generalised Predictive Control. *IEE Proceeding Part D*, 138:2–8.

[Rohrs et al., 1984] Rohrs, C., Valvani, L., Athans, M., and Stein, G. (1984). Some design guidelines for discrete time adaptive controllers. *Automatica*, 20(5):653–660.

[Rohrs et al., 1985] Rohrs, C., Valvani, L., Athans, M., and Stein, G. (1985). Robustness of continuous time adaptive control algorithms in the presence of unmodelled dynamics. *IEEE Transaction on Automatic Control*, 30(9):881–889.

[Rosen, 1960] Rosen, J. B. (1960). The gradient projection method for nonlinear programming. Part 1. Linear constraints. *J. Soc. of indust. Appl. Math.*, 8(1):181.

[Salgado et al., 1988] Salgado, M. E., Goodwin, G. C., and Middleton, R. H. (1988). Modified least squares algorithm incorporating exponential resetting and forgetting. *Int. J. Control*, 47:477–491.

[Shook et al., 1992] Shook, D., Mohtadi, C., and Shah, S. (1992). A control relevant identification strategy for GPC. *IEEE Transactions on Automatic Control*, 37(7):975–981.

[Shook et al., 1991] Shook, D., Mohtadi, C., and Shah, S. L. (1991). Identification for long range predictive control. *IEE Proc. D*, 138(1):75.

[Söderström and Stoica, 1989] Söderström, T. and Stoica, P. (1989). *System identification.* Prentice-Hall, Englewood Cliffs, N.J. USA.

[Soeterboek, 1991] Soeterboek, R. (1991). *Predictive Control – A unified Approach.* Prentice Hall.

[Son and Nomoto, 1981] Son, K. and Nomoto, K. (1981). On the coupled motion of steering and rollings of a high speed container ship. *J.S.N.A. Japan*, 150.

[Thygesen, 1993] Thygesen, B. G. (1993). *Modelling and Control of Systems in the Delta Domain.* PhD thesis, IMM - Technical University of Denmark, Lyngby, Ph.D. Thesis, no. 70.

[Tsang and Clarke, 1988] Tsang, T. and Clarke, D. W. (1988). Generalised predictive control with input constrains. *Proc. IEE - D*, 135(6):451.

[Voda and Landau, 1994] Voda, A. and Landau, I. (1994). Multi-step closed loop identification and control design procedure - applications. In *SYSID'94*, pages 627–632, Copenhagen.

[Wahlberg, 1987] Wahlberg, B. (1987). *On the identification and approximation of linear systems.* PhD thesis, Department of Electrical Engineering, Linköping University, Sweden.

[Wahlberg and Ljung, 1986] Wahlberg, B. and Ljung, L. (1986). Design variables for bias distribution in transfer function estimation. *IEEE Transactions on Automatic Control*, 31(2):134–144.

[Ydstie, 1984] Ydstie, B. (1984). Extended horizon adaptive control. *Proceedings 9th IFAC Symposium on Identification and System Parameter Estimation*, pages 911–915.

This page is intentionally left empty

Ph. D. theses from IMM

1. **Larsen, Rasmus.** (1994). *Estimation of visual motion in image sequences. xiv* + 143 pp.

2. **Rygard, Jens Moberg.** (1994). *Design and optimization of flexible manufacturing systems. xiii* + 232 pp.

3. **Lassen, Niels Christian Krieger.** (1994). *Automated determination of crystal orientations from electron backscattering patterns. xv* + 136 pp.

4. **Melgaard, Henrik.** (1994). *Identification of physical models. xvii* + 246 pp.

5. **Wang, Chunyan.** (1994). *Stochastic differential equations and a biological system. xxii* + 153 pp.

6. **Nielsen, Allan Aasbjerg.** (1994). *Analysis of regularly and irregularly sampled spatial, multivariate, and multi-temporal data. xxiv* + 213 pp.

7. **Ersbøll, Annette Kjær.** (1994). *On the spatial and temporal correlations in experimentation with agricultural applications. xviii* + 345 pp.

8. **Møller, Dorte.** (1994). *Methods for analysis and design of heterogeneous telecommunication networks.* Volume 1-2, *xxxviii* + 282 pp., 283-569 pp.

9. **Jensen, Jens Christian.** (1995). *Teoretiske og eksperimentelle dynamiske undersøgelser af jernbanekøretøjer.* ATV Erhvervsforskerprojekt EF 435. *viii* + 174 pp.

10. **Kuhlmann, Lionel.** (1995). *On automatic visual inspection of reflective surfaces.* ATV Erhvervsforskerprojekt EF 385. Volume 1, *xviii* + 220 pp., (Volume 2, *vi* + 54 pp., fortrolig).

11. **Lazarides, Nikolaos.** (1995). *Nonlinearity in superconductivity and Josephson Junctions. iv* + 154 pp.

12. **Rostgaard, Morten.** (1995). *Modelling, estimation and control of fast sampled dynamical systems. xiv* + 348 pp.

13. **Schultz, Nette.** (1995). *Segmentation and classification of biological objects. xiv* + 194 pp.

14. **Jørgensen, Michael Finn.** (1995). *Nonlinear Hamiltonian systems. xiv* + 120 pp.

15. **Balle, Susanne M.** (1995). *Distributed-memory matrix computations. iii* + 101 pp.

16. **Kohl, Niklas.** (1995). *Exact methods for time constrained routing and related scheduling problems. xviii* + 234 pp.

17. **Rogon, Thomas.** (1995). *Porous media: Analysis, reconstruction and percolation. xiv* + 165 pp.

18. **Andersen, Allan Theodor.** (1995). *Modelling of packet traffic with matrix analytic methods. xvi* + 242 pp.

19. **Hesthaven, Jan.** (1995). *Numerical studies of unsteady coherent structures and transport in two-dimensional flows.* Risø-R-835(EN) 203 pp.

20. **Slivsgaard, Eva Charlotte.** (l995). *On the interaction between wheels and rails in railway dynamics. viii* + 196 pp.

21. **Hartelius, Karsten.** (1996). *Analysis of irregularly distributed points. xvi* + 260 pp.

22. **Hansen, Anca Daniela.** (1996). *Predictive control and identification - Applications to steering dynamics. xviii* + 307 pp.

This page is intentionally left empty