

**Robust Control  
Systems with Uncertain Physical Parameters**

---

# Communications and Control Engineering Series

Editors: B.W. Dickinson · A. Fettweis · J.L. Massey · J.W. Modestino  
E.D. Sontag · M. Thoma

CCES published titles include:

*Sampled-Data Control Systems*  
J. Ackermann

*Interactive System Identification*  
T. Bohlin

*The Riccati Equation*  
S. Bittanti, A.J. Laub and J.C. Willems (Eds.)

*Nonlinear Control Systems*  
A. Isidori

*Analysis and Design of Stream Ciphers*  
R.A. Rueppel

*Sliding Modes in Control Optimization*  
V.I. Utkin

*Fundamentals of Robotics*  
M. Vukobratović

*Parametrizations in Control, Estimation and Filtering Problems:  
Accuracy Aspects*  
M. Gevers and G. Li

*Parallel Algorithms for Optimal Control of Large Scale  
Linear Systems*  
Zoran Gajić and Xuemin Shen

*Loop Transfer Recovery: Analysis and Design*  
A. Saberi, B.M. Chen and P. Sannuti

*Markov Chains and Stochastic Stability*  
S.P. Meyn and R.L. Tweedie

Jürgen Ackermann

in co-operation with Andrew Bartlett, Dieter Kaesbauer,  
Wolfgang Sienel and Reinhold Steinhäuser

# **Robust Control Systems with Uncertain Physical Parameters**

---

With 222 Figures



Springer-Verlag

London Berlin Heidelberg New York

Paris Tokyo Hong Kong

Barcelona Budapest

Jürgen Ackermann, Prof. Dr.-Ing.  
Dieter Kaesbauer, Dr.techn.  
Wolfgang Sienel, Dipl.-Ing.  
Reinhold Steinhauser, Dr.-Ing.

Deutsche Forschungsanstalt fuer Luft- und Raumfahrt (DLR), Institute for  
Robotics and System Dynamics Oberpfaffenhofen, 82230 Wessling/Obb,  
Germany

Andrew Bartlett, PhD

Electrical and Computer Engineering Department, University of Michigan-Dearborn,  
MI 48121, USA

ISBN-13:978-1-4471-3367-4  
DOI: 10.1007/978-1-4471-3365-0

e-ISBN-13:978-1-4471-3365-0

**British Library Cataloguing in Publication Data**  
**A catalogue record for this book is available from the British Library**

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms of licences issued by the Copyright Licensing Agency. Enquiries concerning reproduction outside those terms should be sent to the publishers.

© Springer-Verlag London Limited 1993  
Softcover reprint of the hardcover 1st edition 1993

The publisher makes no representation, express or implied, with regard to the accuracy of the information contained in this book and cannot accept any legal responsibility or liability for any errors or omissions that may be made.

Typesetting: Camera ready by authors

69/3830-543210 Printed on acid-free paper

# Preface

This book is organized in four parts.

## *Part I: Introduction to Some Practical Problems of Robust Control*

In the first part, consisting of Chapters 1 through 3, some control engineering examples are introduced that illustrate the origin of the problems dealt with in the further parts.

The first example that we consider is a crane. It has an uncertain load mass, for which only the lower bound (empty hook) and the upper bound (maximum load of the crane) is known. Similarly, we know the lower and upper bounds for the rope length and for the crab mass.

The second example is four-wheel steering of cars with two actuator signals, i.e. front and rear steering angle. Uncertain parameters are velocity, mass, and adhesion between tire and road surface. The uncertain car steering model is further extended to model an automatic car steering problem for an intelligent vehicle-highway system with the same uncertain parameters.

Finally, a flight control problem for an experimental aircraft – an F4-E with canards – is introduced. This aircraft is aerodynamically stable but badly damped in supersonic flight and aerodynamically unstable with a single right half plane pole in subsonic flight. Uncertain parameters are velocity and flight altitude. In this example, data are given only for four representative operating points.

In all these examples the parametrization by an uncertain real parameter vector  $q$  is physically motivated; this is in contrast to mathematically motivated parametrizations that can be found frequently in the literature (e.g. norm bounded uncertainties of the matrices in state-space models, interval matrices, complex parameters, frequency domain uncertainties, overbounding by interval polynomials). The selection and presentation of the material of this book is oriented at the usefulness of the various methods for problems with physically motivated uncertainties. As a rule we give preference to nonconservative methods and illustrate their application by one of the examples of Chapter 1.

In Chapter 2, feedback control system structures are discussed under the aspects of choice of sensors for robust observability, state and output feedback, integral control, relative degree, and bandwidth of controllers. For the example of the four-wheel steering car introduced in Chapter 1 a robustly decoupling controller and two gain scheduling controllers for velocity-independent yaw eigenvalues and steering dynamics of the car are derived. As a rule we exploit knowledge of the plant model structure as much as possible in the choice of the controller structure and try to solve the problem with a low order controller. This is in contrast to design approaches (like the  $H_\infty$  approach) that yield the controller order. Usually, such controller orders are high, i.e. many additional eigenvalues of the closed-loop are introduced that must be monitored in the robustness analysis.

The free design parameters  $\mathbf{k}$  in the assumed controller structures enter together with the uncertain plant parameters  $\mathbf{q}$  into the closed-loop characteristic polynomial  $p(s, \mathbf{q}, \mathbf{k}) = \sum a_i(\mathbf{q}, \mathbf{k})s^i$ . The polynomials can now be classified by the kind of coefficient functions  $a_i(\mathbf{q}, \mathbf{k})$ . Interesting cases are interval polynomials, affine, multilinear, or polynomial coefficient functions  $a_i(\mathbf{q}, \mathbf{k})$ .

In Chapter 3, control system specifications are discussed from the point of view that unsatisfactory time responses give clues which eigenvalues should be shifted. As a result of this step, a region  $\Gamma$  in the complex  $s$ -plane is obtained where the closed-loop eigenvalues should be located in the design process. A polynomial is called “ $\Gamma$ -stable” if all its roots are located in  $\Gamma$ . The problem of robustness analysis is then: Is  $p(s, \mathbf{q}, \mathbf{k}^*)$  (for a fixed controller  $\mathbf{k}^*$ )  $\Gamma$ -stable for all admissible plant parameter values  $\mathbf{q}$ ? The problem of robust controller synthesis is: Find a  $\mathbf{k} = \mathbf{k}^*$ , such that  $p(s, \mathbf{q}, \mathbf{k}^*)$  is  $\Gamma$ -stable for all admissible  $\mathbf{q}$ . In view of necessary tradeoffs with other design requirements it is desirable to find also an answer to the more general question: Find a set  $K$  such that  $p(s, \mathbf{k}, \mathbf{q})$  is  $\Gamma$ -stable for all  $\mathbf{k} \in K$  and all admissible plant parameters  $\mathbf{q}$ . Then  $\mathbf{k}$  may be chosen from  $K$  such that also other design requirements are satisfied. We are interested for example in small feedback gains to satisfy actuator constraints.

The aim of Part I is to acquaint the reader with some plant models with physically motivated uncertain parameters, feedback structures for these systems, and specifications for their respective closed-loop systems. Part I should motivate the more practically oriented engineer to study uncertain polynomials in Part II and it should give some nontrivial examples and problems for the theoretician.

## *Part II: Stability Analysis of Polynomial Families*

In the first part of this book several practical examples of feedback control systems are given. Common attributes of the examples are:

- The systems are linear and time invariant. Thus, essential dynamic properties can be concluded from their eigenvalues, i.e. the roots of the closed-loop characteristic polynomial.

- There are uncertain parameters entering into the characteristic polynomial as uncertain plant parameters and free parameters in an assumed controller structure.

In Part II of the book we will proceed from engineering art to engineering science and analyze the stability of uncertain polynomials  $p(s, \mathbf{q}) = \sum_{i=0}^n a_i(\mathbf{q})s^i$ . Usually,  $\mathbf{q}$  ranges over a given operating domain  $Q$ , typically a “box” with constraints  $q_j \in [q_j^-; q_j^+]$  for each element  $q_j$  of  $\mathbf{q}$ . An uncertain polynomial with given  $Q$  is called a polynomial family  $\mathcal{P}(s, Q) = \{p(s, \mathbf{q}) \mid \mathbf{q} \in Q\}$ . The characteristic polynomial is taken as the only interface between Parts I and II. Other interfaces between feedback control problems and robustness theory will be treated in Part III. Part II is primarily structured into methods.

In Chapter 4 several classical stability tests are reviewed and applied to uncertain polynomials. Whenever possible, we use graphical representations of the results, because they give additional information on “closeness to instability” instead of only a yes or no answer to the robustness analysis question: Is a given polynomial family stable? A brute force approach is based on gridding  $Q$  and a display of the resulting roots in the  $s$ -plane. Less computational effort is involved in approaches that check if roots of  $p(s, \mathbf{q})$  can cross the imaginary axis of the  $s$ -plane. An algebraic solution uses the critical Hurwitz conditions. In frequency domain methods it is important to determine and analyze the “singular frequencies” separately. Graphical displays can be derived from stability boundaries in the space of parameters  $\mathbf{q}$  or from families of Mikhailov plots  $\mathcal{P}(j\omega, Q)$ . The latter approach leads to the concept of “zero exclusion from the value set”.

In Chapter 5, more recent results on testing sets are treated. A subset  $Q_T$  of  $Q$  is called a testing set if stability of  $\mathcal{P}(s, Q_T)$  implies stability of  $\mathcal{P}(s, Q)$ . Kharitonov’s theorem solves the case of interval polynomials and the edge theorem by Bartlett, Hollot, and Huang solves the case of affine coefficient functions  $a_i(\mathbf{q})$ . For nonlinear coefficient functions, interior points of  $Q$  must also be tested; they are determined by Jacobi conditions.

The examples of the first chapters show that there is a practical need for robustness analysis tools for the case of nonlinear coefficient functions  $a_i(\mathbf{q})$ . Such a tool is treated in Chapter 6; it is the construction of value sets. The concept of zero exclusion from the value set is used in Chapter 5 only in proofs. In Chapter 6, we actually construct complex value sets  $\mathcal{P}(j\omega^*, Q)$  at a fixed frequency  $\omega^*$ , repeating this for a grid on  $\omega$  and checking the graphic display for zero exclusion. Also, a larger number of uncertain parameters can be treated provided they appear in a special structure that allows a “tree structured decomposition” of the characteristic polynomial. Then, the value set can be constructed by sequential operations involving only subsets of the parameters.

In Chapter 7 the parameter set (operating domain)  $Q$  is no longer fixed, it may be dilated by a scalar factor until it hits the stability boundary in the parameter space. Thus, the “radius of stability” can be determined and the smallest destabilizing perturbation (worst case operating condition) and the frequency at which the corresponding root set first touches the imaginary axis. For the affine case the Tsypkin-Polyak plot is used and for the multilinear and polynomial cases a finite number of candidates are

determined, the smallest one is the radius of stability. The calculation of the radius of stability for the affine case is simplified, if a “parameter ball” is blown up around a nominal operating point instead of a parameter box.

### *Part III: Robustness Analysis of Feedback Systems*

In Part II of this book, we use the characteristic polynomial as the interface between the real world of robust control problems (Part I) and the mathematical world of robust stability of uncertain polynomials and polynomial families (Part II). From classical control theory we know that there are many other aspects to be considered in analysis and design of control systems, for example actuator nonlinearities, other design requirements than just stability, and digital controller implementation. Single-loop structures can be exploited in the analysis. Some such practically important topics have been selected for Part III and are discussed under robustness aspects.

In Chapter 8, we assume a single-loop negative feedback structure with open-loop transfer function  $-g_0(s, q)$ . Then the closed-loop characteristic equation may be written as  $g_0(s, q) = -1$ . Some useful results for interval plants (interval numerator and denominator polynomials) in a loop with a compensator are summarized. It is shown when it suffices to check some extreme points for stability. Also robustness of a loop with positive real plant and compensator is treated. Nyquist value sets are constructed by tree structured decomposition of fractions. One very early robustness result is the Popov criterion for stability robustness with respect to an unknown nonlinear characteristic in a sector. This criterion is extended to uncertain plants.

In a feedback loop there are several possibilities to define a stability margin as an implicit measure of performance, e.g. distance of the Nyquist value set from the critical point  $-1$  or radius of stability in the parameter space (see Chapter 7). Chapter 9 looks at a third possibility that is suggested by the eigenvalue specifications discussed in Chapter 3. It is Gamma-stability as defined by a region  $\Gamma$  in the complex plane, where all eigenvalues must be located. The various stability tests of Part II are reviewed and modified in Chapter 9 in view of their applicability to  $\Gamma$ -stability.

Controllers are usually implemented in digital computers. The discrete-time compensator transfer function may be obtained by discretization of a continuous-time design or by design of a common sampled-data controller for some representative operating conditions. In both cases a robust stability analysis for the resulting sampled-data loop with a continuum of operating conditions is needed. The sections of Chapter 10 correspond directly to Chapters 1 to 9 and discuss the sampled-data versions of the results. Exact methods for roots in the unit circle are summarized and a useful approximation is given for the more realistic case of exponential coefficient functions  $a_i(q)$ .

## *Part IV: Some Design Tools for Robust Control Systems*

There are no general results on robust stabilizability of a plant family, not even if the continuum of plants is represented by a finite number of representatives. For the latter problem formulation of “simultaneous stabilization” two design tools are introduced that are not based on conservative estimates.

The first tool for parameter space design describes for each operating condition an admissible region in the space of controller parameters. The intersection of these regions is the set of simultaneous Gamma-stabilizers. A feasible controller may be selected from this intersection in view of small gains, gain margins and robustness with respect to sensor failures. This tool is implemented with graphics of two-dimensional cross sections. Some supporting ideas for the choices of such cross sections are discussed and illustrated by design studies for the crane, for an automatic car steering system, and a robust aircraft stabilization.

The second tool is simultaneous design by performance vector constraints. In this approach the designer steers the design process in the controller parameter space interactively to a Pareto optimal solution. Different forms of performance measures may be included in the performance vector. In the context of robust control design we are particularly interested in design tradeoffs between different representative operating conditions. This design process is illustrated on an active car steering system.

## *Appendices*

A detailed model of car steering dynamics from a control systems point of view is presented in Appendix A. Appendix B reviews some useful mathematical results on polynomials and polynomial equations.

## *General remarks*

The prerequisite for the reader is an undergraduate course in feedback control systems. We try to keep the mathematics simple. The book is suited for a graduate level course on “Robust Control”, in fact the material was selected and used for such courses at the University of California, Irvine and at the Technische Universität München and in a short course for Scandinavian Ph.D. students in Lyngby (Denmark). It was also used in continuing education courses of the Carl-Cranz Gesellschaft (CCG), Oberpfaffenhofen, for participants from industry.

For the purpose of such courses a very restrictive selection had to be made from the large and rapidly growing literature on robust control. Therefore, many important contributions and alternative approaches could not be mentioned. Some cross-references

are given in form of remarks. Remarks also indicate possible generalizations, open problems, and other supplements that are not prerequisites for understanding the following sections. The beginner should ignore all remarks, they are intended for the advanced reader.

In the examples with physical parameters, units are given in brackets, e.g. [m] for meter (to be distinguished from the symbol  $m$  for mass) or [s] for seconds (to be distinguished from the complex variable  $s$  of Laplace transforms). In calculations the units are omitted. We use the following units:

Physical Variable	Symbol	Unit
Length	$\ell$	Meter [m]
Time	$t$	Seconds [s]
Mass	$m$	Kilogramm [kg]
Moment of Inertia	$J$	[kg · m <sup>2</sup> ]
Force	$f$	Newton [N] = [kg · m/s <sup>2</sup> ]
Velocity	$v$	[m/s]
Acceleration	$a$	[m/s <sup>2</sup> ]
Angles	$\alpha, \beta, \dots$	Radian [rad]

The symbols are defined only in the context of the particular example, e.g.  $\ell$  may be the rope length of a crane in one example or it may be the wheelbase of a car in a different example. In general contexts outside such examples  $\ell$  is the (integer) number of uncertain parameters. Similarly,  $m$  may be the (integer) degree of a numerator polynomial of a transfer function,  $f$  may be a general function,  $J$  may be a Jacobian,  $v$  may be a transformed complex variable, and  $a$  may be a coefficient. We have tried to use notations that are standard in the literature, the price is the multiple use of some symbols.

*Acknowledgements*

The authors like to thank C. Hollot for his careful review of the manuscript. Also V. Utkin and T. Connolly have given very helpful corrections and suggestions.

The camera-ready manuscript was produced by the team of co-authors, special thanks go to Mrs. G. Kieselbach for typing a part of the manuscript and to Mrs. C. Bell for drawing part of the figures.

Oberpfaffenhofen  
April 1993

Jürgen Ackermann  
Andrew Bartlett  
Dieter Kaesbauer  
Wolfgang Sienel  
Reinhold Steinhauser

# Contents

<b>I Introduction to Some Practical Problems of Robust Control</b>	<b>1</b>
<b>1 Examples for Modelling of Plants with Uncertain Parameters</b>	<b>3</b>
1.1 Crane . . . . .	4
1.2 Four-Wheel Car Steering . . . . .	9
1.3 Automatic Car Steering . . . . .	15
1.4 A Flight Control Problem . . . . .	19
1.5 Notation for Uncertain Plants . . . . .	21
1.6 Exercises . . . . .	23
<b>2 Control System Structures</b>	<b>25</b>
2.1 Robust Controllability and Observability . . . . .	26
2.2 State and Output Feedback . . . . .	29
2.3 Choice of Sensors . . . . .	35
2.4 Further Aspects of Controller Structures . . . . .	40
2.5 Robust Decoupling of Car Steering Dynamics . . . . .	44
2.6 Gain Scheduling Controllers . . . . .	49
2.7 Problem Classes of Parametric Polynomials . . . . .	54
2.8 Exercises . . . . .	55
<b>3 Analysis and Design</b>	<b>57</b>
3.1 Eigenvalue Specifications . . . . .	57
3.2 Introduction to Robustness Analysis . . . . .	67
3.3 Introduction to Robust Controller Design . . . . .	71
3.4 Three Basic Rules of Robust Control . . . . .	76
3.5 Exercises . . . . .	76

<b>II Stability Analysis of Polynomial Families</b>	77
<b>4 Classical Stability Tests Applied to Uncertain Polynomials</b>	79
4.1 Root Set Problem Formulation . . . . .	80
4.2 Boundary Crossing . . . . .	87
4.3 Algebraic Problem Formulation . . . . .	90
4.4 Singular Frequencies . . . . .	99
4.5 Parameter Space Problem Formulation . . . . .	106
4.6 Frequency Plot Problem Formulation . . . . .	112
4.7 Exercises . . . . .	121
<b>5 Testing Sets</b>	123
5.1 Interval Polynomials: Kharitonov's Theorem . . . . .	124
5.2 Affine Coefficients: Edge Theorem . . . . .	135
5.3 A Warning Example . . . . .	140
5.4 Jacobi Conditions . . . . .	142
5.5 The Mapping Theorem . . . . .	147
5.6 Exercises . . . . .	153
<b>6 Value Set Construction</b>	155
6.1 Sequential Set Operations . . . . .	155
6.2 Simplification of Elementary Value Set Operations . . . . .	157
6.3 Computer-Aided Execution of Value Set Operations . . . . .	162
6.4 Color Coding of Value Sets . . . . .	164
6.5 Tree Structured Decomposition . . . . .	166
6.6 Animation of Value Sets . . . . .	174
6.7 Exercises . . . . .	176
<b>7 The Stability Radius</b>	178
7.1 Tsyplkin-Polyak Loci . . . . .	179
7.2 Affine Dependency: The Largest Hypersphere in Parameter Space . . . . .	186
7.3 Polynomial Dependency . . . . .	192
7.4 Exercises . . . . .	198

<b>III Robustness Analysis of Feedback Systems</b>	<b>201</b>
<b>8 Single-Loop Feedback Structures</b>	<b>203</b>
8.1 Interval Plants with a Compensator . . . . .	205
8.2 Positive Interval Plants with a Positive Compensator . . . . .	211
8.3 Tree Structured Transfer Functions . . . . .	216
8.4 Robustness with Respect to Sector Nonlinearities . . . . .	226
8.5 Exercises . . . . .	231
<b>9 Gamma-Stability</b>	<b>234</b>
9.1 Boundary Representation . . . . .	234
9.2 Boundary Crossing . . . . .	238
9.3 Algebraic Problem Formulation . . . . .	238
9.4 Gamma-Stability Boundaries in Parameter Space . . . . .	239
9.5 Value Sets for Gamma-Stability . . . . .	245
9.6 Gamma-Stability Radius . . . . .	248
9.7 Testing Sets . . . . .	250
9.8 Exercises . . . . .	253
<b>10 Robustness of Sampled-Data Control Systems</b>	<b>255</b>
10.1 Plant Discretization . . . . .	256
10.2 Discrete-time Controllers . . . . .	258
10.3 Eigenvalue Specifications . . . . .	259
10.4 Classical Stability Tests . . . . .	263
10.5 Testing Sets . . . . .	276
10.6 Construction of Value Sets . . . . .	280
10.7 Real Radius of Stability . . . . .	280
10.8 Single-Loop Feedback Structures . . . . .	280
10.9 Circle Stability . . . . .	284
10.10 Exercises . . . . .	288

<b>IV Some Design Tools for Robust Control Systems</b>	291
<b>11 Parameter Space Design</b>	293
11.1 Introduction to Design by Simultaneous Gamma-Stabilization . . . . .	293
11.2 Pole Region Assignment . . . . .	295
11.3 Intersections in Controller Parameter Space . . . . .	304
11.4 Selection of a Controller from the Admissible Set . . . . .	310
11.5 Case Study: Automatic Steering of a Bus . . . . .	318
11.6 Case Study: Flight Control . . . . .	330
11.7 Exercises . . . . .	338
<b>12 Design by Optimizing a Vector Performance Index</b>	340
12.1 Formulation of Design Specifications for a Finite Plant Family . . . . .	341
12.2 Concept of the Design by Optimizing a Vector Performance Index . . .	346
12.3 Case Study: Car Steering . . . . .	356
<b>V Appendix</b>	369
<b>A The Four-Wheel Car Steering Model</b>	371
A.1 Single-Track Model . . . . .	371
A.2 Linearized Single-Track Model . . . . .	375
<b>B Polynomials and Polynomial Equations</b>	378
B.1 Polynomials in One Variable . . . . .	378
B.2 Polynomials in Two Variables . . . . .	381
B.3 Some Properties of Two-dimensional Curves . . . . .	381
B.4 Two Polynomials in Two Variables . . . . .	382
B.5 Several Polynomials in Several Variables . . . . .	384
<b>Bibliography</b>	385
<b>Index</b>	399
<b>Color plates</b>	405

## **Part I**

# **Introduction to Some Practical Problems of Robust Control**

# 1 Examples for Modelling of Plants with Uncertain Parameters

A mathematical model for the plant dynamics is the basis for analysis and design of control systems. For linear time-invariant systems we commonly have a state space model of the form

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t)\end{aligned}\tag{1.0.1}$$

with  $\mathbf{u}$  the vector of input signals (manipulated variables), the state vector  $\mathbf{x}$ , and the vector of output signals  $\mathbf{y}$  (performance variables). The output signals are often comprised of those variables that are measured and hence available for feedback.

Another commonly used form of the model can be obtained by Laplace transformation of (1.0.1)

$$\begin{aligned}s\mathbf{x}(s) - \mathbf{x}(0) &= \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s) \\ \mathbf{y}(s) &= \mathbf{C}\mathbf{x}(s)\end{aligned}\tag{1.0.2}$$

Solving for  $\mathbf{x}(s)$  and premultiplying by  $\mathbf{C}$  gives the transformed output vector

$$\mathbf{y}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(s)\tag{1.0.3}$$

with the transfer function matrix

$$\mathbf{G}(s) := \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\tag{1.0.4}$$

For zero initial conditions,  $\mathbf{x}(0) = \mathbf{0}$ ,

$$\mathbf{y}(s) = \mathbf{G}(s)\mathbf{u}(s)\tag{1.0.5}$$

A state space model ( $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ) or a transfer function model  $\mathbf{G}(s)$  can be determined from different approaches. The first possibility is called input-output modelling. In this approach, the model is obtained from experiments on the plant which is treated like a “black box”. During the experimentation the signals  $\mathbf{u}(t)$  and  $\mathbf{y}(t)$  are measured. These data are then processed to derive a system model. The second approach is often called analytical modelling. In this approach, the model structure is derived from first principles of physics. A model typically depends on some parameter values that are individually measured or estimated between lower and upper bounds.

In this book, the second approach, analytical modelling, is assumed. Its advantage is that the controller can be designed before the plant is fully operational and available for experiments (i.e. cannot be drastically changed). Thus, a concurrent design of plant and controller becomes feasible. Also, safety critical experiments with the uncontrolled plant are avoided. For a space vehicle, such experiments would be impossible. Practically, we have to combine input-output modelling and analytical modelling, e.g. some submodels may be obtained by experiments with the separated subsystem or by numerical approximation as in finite element models. Other subsystem models can be obtained by analytical modelling. Eventually all subsystems are integrated in an overall system. The following examples illustrate analytical modelling of systems with uncertain parameters, the starting point for our later robustness investigations.

Many more examples for continuous system modelling can be found in [43].

## 1.1 Crane

Consider the crane of Fig. 1.1.

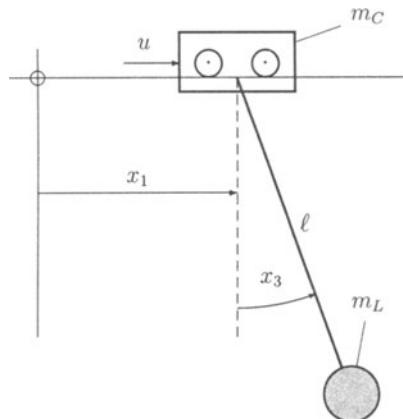


Fig. 1.1: Crane

The task of the crane or loading bridge is for example to load containers into a ship. First the load mass is only the weight of the empty hook. The hook must come to rest above the container. Sometimes a man is helping to damp the swaying pendulum and to fix the hook. In our approach the placement of the crab above the known position of the container and the damping of the pendulum shall be performed automatically. After lifting the container it is transported for some distance to the vicinity of the

hatch of the ship. This motion can be commanded by feedforward control considering the limited motor force at the crab and safety constraints. Feedback control is needed again in order to position the container above the hatch and dampen the motion. It must be almost at rest, before it can be lowered into the ship. In this application, the load mass varies widely between the weight of the empty hook and the maximum load that the crane can carry. Also, the rope length may vary (even more for a construction crane). In contrast, the mass of the crab varies very little (e.g. if a fat or a slim crane operator is sitting in the gantry).

In common design approaches a linearized model is used for controller design. It models small motions for positioning the empty hook or the container with sufficient accuracy. It is important that the controller is designed such that the assumptions made in linearization are not violated.

The input signal is the force  $u$  that accelerates the crab. The crab mass is  $m_C$ . Further parameters are rope length  $\ell$ , load mass  $m_L$ , and gravity acceleration  $g$ . The position of the crab is  $x_1$  and its velocity  $\dot{x}_1 =: x_2$ , the rope angle is  $x_3$  (in radian) and the angular velocity  $\dot{x}_3 =: x_4$ .

The following assumptions are made for simplification of modelling:

A1)

Dynamics and nonlinearity of the driving motor are neglected. This assumption makes sense only if the controller design guarantees that  $|u|$  and  $|\dot{u}|$  are not excessively large.

A2)

The crab moves along the track without friction or slip.

A3)

The rope has no mass and no elasticity.

A4)

There is no damping of the pendulum (e.g. from air drag).

A5)

The parameters are constant during each operation time interval of the loading bridge.

As shown in [5], the plant may be described by two nonlinear second order differential equations for the horizontal and vertical motions.

$$\begin{aligned} (m_L + m_C)\ddot{x}_1 + m_L\ell(\ddot{x}_3 \cos x_3 - \dot{x}_3^2 \sin x_3) &= u \\ m_L\ddot{x}_1 \cos x_3 + m_L\ell\ddot{x}_3 &= -m_L g \sin x_3 \end{aligned} \quad (1.1.1)$$

For deriving a state space description, the scalar differential equations must be solved for the highest derivatives  $\ddot{x}_1$  and  $\ddot{x}_3$ . Rewrite (1.1.1) as

$$\begin{bmatrix} m_L + m_C & m_L\ell \cos x_3 \\ m_L \cos x_3 & m_L\ell \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_3 \end{bmatrix} = \begin{bmatrix} m_L\ell\dot{x}_3^2 \sin x_3 + u \\ -m_L g \sin x_3 \end{bmatrix} \quad (1.1.2)$$

The determinant of the matrix on the left hand side is  $m_L\ell(m_C + m_L \sin^2 x_3)$ . It vanishes for  $m_L = 0$  or  $\ell = 0$ . In these two cases the system degenerates to a second order system.

It is a standing assumption in all following discussions of crane control that  $m_L > 0$  and  $\ell > 0$ . Then

$$\begin{aligned}\ddot{x}_1 &= f_1(x_3, x_4, u) \\ \ddot{x}_3 &= f_2(x_3, x_4, u)\end{aligned}\tag{1.1.3}$$

with

$$\begin{aligned}f_1(x_3, x_4, u) &:= \frac{u + (g \cos x_3 + \ell x_4^2)m_L \sin x_3}{m_C + m_L \sin^2 x_3} \\ f_2(x_3, x_4, u) &:= -\frac{u \cos x_3 + (g + \ell x_4^2 \cos x_3)m_L \sin x_3 + g m_C \sin x_3}{\ell(m_C + m_L \sin^2 x_3)}\end{aligned}$$

With the state vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_3 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} \text{crab position} \\ \text{crab velocity} \\ \text{rope angle} \\ \text{rope angle rate} \end{bmatrix}\tag{1.1.4}$$

the nonlinear state space model is written as

$$\dot{\mathbf{x}} = f(\mathbf{x}, u) = \begin{bmatrix} x_2 \\ f_1(x_3, x_4, u) \\ x_4 \\ f_2(x_3, x_4, u) \end{bmatrix}\tag{1.1.5}$$

This state space model can be used for simulations of the crane motion by numerical integration. For this purpose, numerical values for the parameters  $g$ ,  $\ell$ ,  $m_L$ , and  $m_C$ , an initial condition for the state  $\mathbf{x}(0)$ , and an input function  $u(t)$  must be chosen.

The nonlinear model (1.1.5) is linearized for small deflection angle  $x_3$  and small angular velocity  $x_4$ . Setting

$$\cos x_3 \approx 1, \sin x_3 \approx x_3, \sin^2 x_3 \approx 0, x_4^2 \approx 0$$

it is easy to derive the following linear state space model.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}u \\ \mathbf{A} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 0 \\ b_2 \\ 0 \\ b_4 \end{bmatrix}\end{aligned}\tag{1.1.6}$$

$$\begin{aligned} a_{23} &= \frac{m_L}{m_C}g, & b_2 &= \frac{1}{m_C} \\ a_{43} &= -\frac{(m_L + m_C)g}{m_C\ell}, & b_4 &= -\frac{1}{m_C\ell} \end{aligned}$$

The characteristic polynomial of  $\mathbf{A}$

$$\begin{aligned} p_A(s) &= \det(s\mathbf{I} - \mathbf{A}) \\ &= s^2(s^2 - a_{43}) \\ &= s^2[s^2 + (1 + m_L/m_C)g/\ell] \end{aligned} \quad (1.1.7)$$

yields the parameter independent eigenvalues

$$s_{1,2} = 0$$

and the parameter dependent eigenvalues

$$s_{3,4} = \pm j\sqrt{1 + m_L/m_C}\sqrt{g/\ell}$$

For the vector of transfer functions from  $u$  to the state vector  $\mathbf{x}(s) = \mathbf{g}(s)u(s)$  we get

$$\mathbf{g}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} = \frac{1}{m_C\ell p_A(s)} \begin{bmatrix} s^2\ell + g \\ s(s^2\ell + g) \\ -s^2 \\ -s^3 \end{bmatrix} \quad (1.1.8)$$

Consider now three output variables: Load position  $y_L$ , crab position  $y_C$ , and rope angle  $y_R$ , for which we write the respective transfer functions.

a) Output load position (horizontal)  $y_L = x_1 + \ell \sin x_3 \approx x_1 + \ell x_3$

$$g_L(s) = [1 \ 0 \ \ell \ 0] \mathbf{g}(s) = \frac{g}{s^2[m_C\ell s^2 + (m_L + m_C)g]} \quad (1.1.9)$$

b) Output crab position  $y_C = x_1$

$$g_C(s) = [1 \ 0 \ 0 \ 0] \mathbf{g}(s) = \frac{s^2\ell + g}{s^2[m_C\ell s^2 + (m_L + m_C)g]} \quad (1.1.10)$$

This transfer function has parameter dependent zeros at  $\pm j\sqrt{g/\ell}$ . The poles and zeros are shown in Fig. 1.2.

c) Output rope angle  $y_R = x_3$

$$g_R(s) = [0 \ 0 \ 1 \ 0] \mathbf{g}(s) = \frac{-1}{m_C\ell s^2 + (m_L + m_C)g} \quad (1.1.11)$$

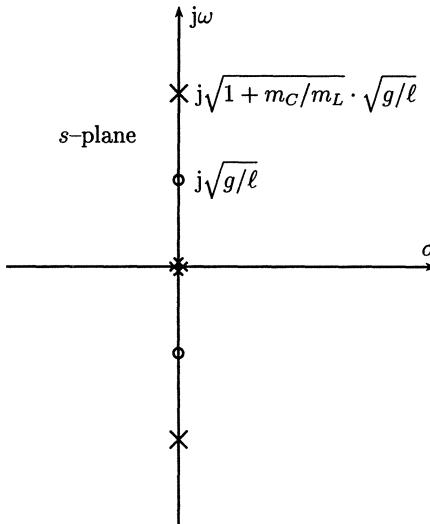


Fig. 1.2: Poles and zeros of the transfer function  $g_C(s)$  from force  $u$  to crab position  $x_1$ .

This transfer function is second order instead of fourth order because the double pole  $s^2 = 0$  of the subsystem crab is canceled by a double zero  $s^2 = 0$ . The cancellation occurs because the crab subsystem is not observable from the rope angle  $x_3$ . This can also be seen from the state space model (1.1.6). This model has the canonical form for separation of observable and unobservable subsystems as given by Kalman [101] and Gilbert [78]

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & | & 0 & 0 \\ 0 & 0 & | & a_{23} & 0 \\ - & - & - & - & - \\ 0^* & 0^* & | & 0 & 1 \\ 0^* & 0^* & | & a_{43} & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ b_2 \\ - \\ 0 \\ b_4 \end{bmatrix} u \quad (1.1.12)$$

$$y_R = \begin{bmatrix} 0^* & 0^* & | & 1 & 0 \end{bmatrix} \mathbf{x}$$

Crucial are the zeros with an asterisk, they indicate that the zero input dynamics of the crab (states  $x_1$  and  $x_2$ ) with arbitrary initial conditions have no influence on the deflection angle  $x_3$ . The transfer function (1.1.11) describes only the

controllable and observable subsystem

$$\begin{aligned} \begin{bmatrix} \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ a_{43} & 0 \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ b_4 \end{bmatrix} u \\ y_R &= [1 \quad 0] \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} \end{aligned} \tag{1.1.13}$$

We will come back to the loading bridge in later chapters. There we will consider that the load mass  $m_L$  and the rope length  $\ell$  are parameters with large uncertainty. Also the crab mass  $m_C$  may be treated as an uncertain parameter. The gravitational acceleration is assumed fixed at  $g = 10 \text{ [m} \cdot \text{s}^{-2}\text{]}$ .

In principle the parameters  $m_L$ ,  $m_C$ , and  $\ell$  can be measured before each new operation of the crane. However this would not be practical. In the context of robust control the uncertain parameters are treated as fixed but unknown quantities for which only lower and upper bounds are known.

## 1.2 Four-Wheel Car Steering

The second example for modelling of a plant with uncertain parameters is four-wheel car steering (4WS). In addition to conventional front wheel steering with a steering angle  $\delta_f$  also rear wheel steering with an angle  $\delta_r$  is assumed as shown in Fig. 1.3. A pair of wheels on the same axle is steered by the same angle.

The manipulated variables of the system are the rear and front steering angles  $\delta_r$  and  $\delta_f$ . State variables are as explained in Appendix A,

- $\beta$  sideslip angle between the vehicle center line and the velocity vector  $\vec{v}$  at the center of gravity (CG),
- $r$  yaw rate.

The lateral acceleration  $a_f$  of the front axle is a performance variable for the steering behavior. The vehicle mass  $m$  and the adhesion between tires and road surface are uncertain parameters. Also the velocity  $v = |\vec{v}|$  is considered as an uncertain parameter.

*Remark 1.1.* If acceleration and braking of the car are also modelled, then  $v$  is another state variable. We assume here however that  $v$  is unknown but constant.  $\square$

Fig. 1.4a shows a feedforward structure for commanding front and rear steering angles from the steering wheel angle  $\delta_S$ . This is the most commonly used control structure for 4WS cars. The front wheel steering is unchanged with a mechanical connection from  $\delta_S$  to  $\delta_f$  via the steering column and gear. The motor for steering the rear wheels is controlled via a prefilter  $F_r$ . The prefilter may be a velocity-scheduled gain, it may be also a dynamic filter. Fig. 1.4b illustrates a control structure with underlying feedback

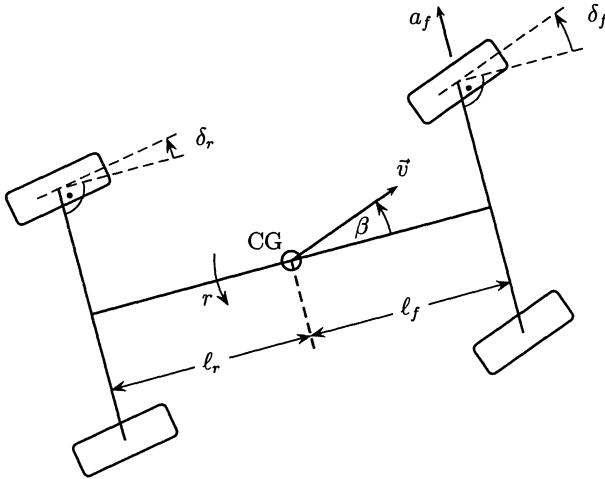


Fig. 1.3: Car with front and rear wheel steering

of the yaw rate to the rear wheels. This structure has been implemented with a vibration gyro measuring the yaw rate  $r$ , see [91].  $H_r$  is a dynamic compensator. Here the rear wheels are not only steered by the driver via his steering wheel command  $\delta_S$ , but also by yaw motion of the car. If for example a side-wind disturbance  $d$  excites a yaw motion, then the automatic control system can react immediately. The driver would need some reaction time to recognize the effect of side-wind on his car and to decide about countermeasures. A disturbance  $d$  also occurs by braking with asymmetric road-tire contact ( $\mu$ -split braking).

Even more design flexibility is offered by feeding back the measured yaw rate also to the front wheels as shown in Fig. 1.4c. Active control of the front wheels has been studied for example in [102, 59]. In Chapter 2 a specific controller  $H_f(s)$  for robust decoupling of the lateral mode (involving  $a_f$ ) and the yaw mode (involving  $r$ ) will be introduced. A velocity-scheduled controller  $H_r(s, v)$  for velocity-invariant yaw eigenvalues will be derived. Also additional accelerometer feedback to front wheel steering will be used in order to obtain a velocity-invariant steering transfer function. All these controller structures are based on a detailed analysis of the car steering model.

For the system ‘‘steering dynamics’’ in Fig. 1.4, dynamic models of different degrees of complexity are known in the automotive literature, e.g. [59, 150, 130, 183]. In this book the classical Riekert-Schunck model [145] is used. The details of this model are described in Appendix A. The resulting linearized equations, see (A.2.7), are

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (1.2.1)$$

where

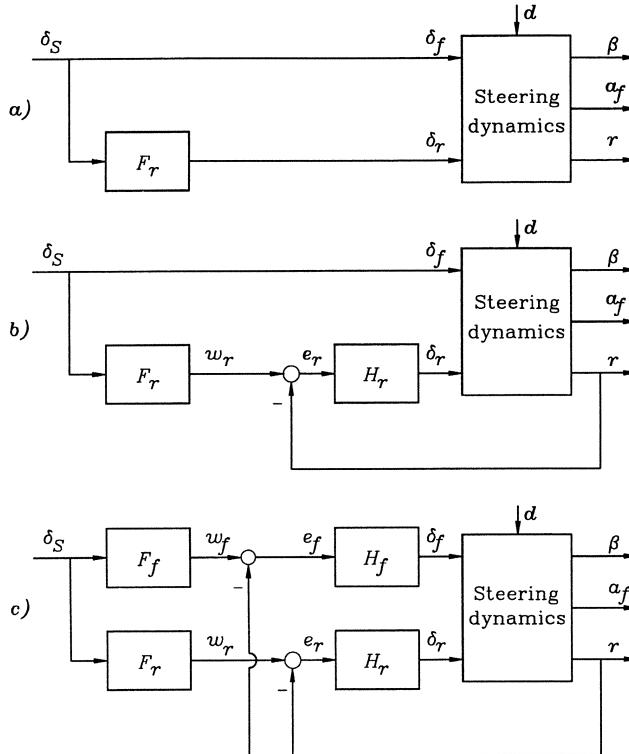


Fig. 1.4: Controller structures for 4WS: a) Feedforward structure, b) Yaw rate feedback to rear wheel steering, c) Yaw rate feedback to rear and front wheel steering.

$$\begin{aligned}
 a_{11} &= -(c_r + c_f)/\tilde{m}v \\
 a_{12} &= -1 + (c_r \ell_r - c_f \ell_f)/\tilde{m}v^2 \\
 a_{21} &= (c_r \ell_r - c_f \ell_f)/\tilde{J} \\
 a_{22} &= -(c_r \ell_r^2 + c_f \ell_f^2)/\tilde{J}v \\
 b_{11} &= c_f/\tilde{m}v \\
 b_{12} &= c_r/\tilde{m}v \\
 b_{21} &= c_f \ell_f/\tilde{J} \\
 b_{22} &= -c_r \ell_r/\tilde{J}
 \end{aligned}$$

The cornering stiffnesses ( $c_r$  for the rear wheels,  $c_f$  for the front wheels) are empirically determined tire parameters entering into the side force equations as explained in Appendix A. The distances  $\ell_r$  and  $\ell_f$  from the center of gravity to the rear and front axles sum up to the wheelbase  $\ell = \ell_r + \ell_f$ , see Fig. 1.3. The vehicle mass  $m$  is normalized by

a road adhesion factor  $\mu$ , i.e.  $\tilde{m} = m/\mu$  is a virtual mass, see Appendix A.2. Similarly the moment of inertia  $J$  is normalized as  $\tilde{J} = J/\mu$ . The characteristic polynomial of the system matrix in (1.2.1) is

$$\begin{aligned} p_A(s) &= (s - a_{11})(s - a_{22}) - a_{12}a_{21} \\ &= a_0 + a_1 s + s^2 \end{aligned} \quad (1.2.2)$$

with the parameter dependent coefficients

$$\begin{aligned} a_0 &= \frac{c_f c_r \ell^2}{\tilde{m} \tilde{J} v^2} + \frac{c_r \ell_r - c_f \ell_f}{\tilde{J}} \\ a_1 &= \frac{c_r + c_f}{\tilde{m} v} + \frac{c_r \ell_r^2 + c_f \ell_f^2}{\tilde{J} v} \end{aligned}$$

The system is stable for  $a_0 > 0$ ,  $a_1 > 0$ . Two cases must be distinguished

- a) for  $c_r \ell_r - c_f \ell_f \geq 0$ , the system is stable,
- b) for  $c_r \ell_r - c_f \ell_f < 0$  the system is stable, if the “critical speed”  $v_{crit}$  is not exceeded, where

$$v_{crit}^2 := \frac{c_f c_r \ell^2}{\tilde{m} (c_f \ell_f - c_r \ell_r)} \quad (1.2.3)$$

Typical vehicle data correspond to case a).

If the characteristic polynomial  $p_A(s)$  is stable, then it may be written in terms of damping  $D$  and natural frequency  $\omega_0$  as

$$p_A(s) = \omega_0^2 + 2D\omega_0 s + s^2 \quad (1.2.4)$$

where  $\omega_0$  and  $D$  depend on the physical parameters as follows

$$\omega_0^2 = \frac{c_f c_r \ell^2 + \tilde{m} v^2 (c_r \ell_r - c_f \ell_f)}{\tilde{m} \tilde{J} v^2} \quad (1.2.5)$$

$$D = \frac{\tilde{J} (c_r + c_f) + \tilde{m} (c_r \ell_r^2 + c_f \ell_f^2)}{2 \sqrt{\tilde{m} \tilde{J} [c_r c_f \ell^2 + \tilde{m} v^2 (c_r \ell_r - c_f \ell_f)]}} \quad (1.2.6)$$

For large velocities  $v$  the damping  $D$  goes to zero and the eigenvalues approach the imaginary axis at

$$\omega_\infty = \lim_{v \rightarrow \infty} \omega_0 = \sqrt{\frac{c_r \ell_r - c_f \ell_f}{\tilde{J}}} \quad (1.2.7)$$

The term  $c_r \ell_r - c_f \ell_f$  in (1.2.7) is positive for typical vehicle data. With decreasing velocity  $v$ , the damping  $D$  becomes larger than one, i.e. the pair of complex eigenvalues meets at the real axis and branches into a pair of real eigenvalues. For  $v \rightarrow 0$ , two real eigenvalues go to minus infinity. Fig. 1.5 illustrates the velocity dependence of the yaw eigenvalues.

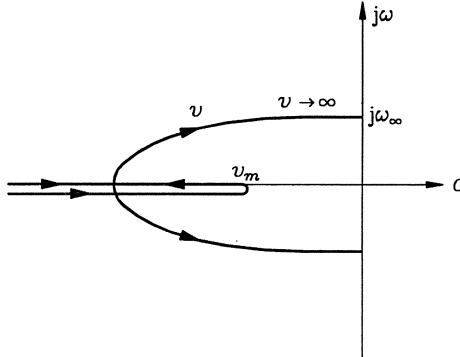


Fig. 1.5: Velocity dependence of yaw eigenvalues.

Note that Fig. 1.5 is not a standard root locus because the parameter  $v$  enters nonlinearly. At some velocity  $v_m$  one of the real eigenvalues has a maximum.

The transfer functions for (1.2.1) are

$$\begin{bmatrix} \beta(s) \\ r(s) \end{bmatrix} = \begin{bmatrix} s - a_{11} & -a_{12} \\ -a_{21} & s - a_{22} \end{bmatrix}^{-1} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} \delta_f(s) \\ \delta_r(s) \end{bmatrix}$$

$$= \frac{1}{a_0 + a_1 s + s^2} \begin{bmatrix} n_{11}(s) & n_{12}(s) \\ n_{21}(s) & n_{22}(s) \end{bmatrix} \begin{bmatrix} \delta_f(s) \\ \delta_r(s) \end{bmatrix} \quad (1.2.8)$$

with

$$n_{11}(s) = c_f \left( \frac{s}{\tilde{m}v} - \frac{\ell_f}{\tilde{J}} + \frac{c_r \ell_r \ell}{\tilde{J} \tilde{m} v^2} \right)$$

$$n_{12}(s) = c_r \left( \frac{s}{\tilde{m}v} + \frac{\ell_r}{\tilde{J}} + \frac{c_f \ell_f \ell}{\tilde{J} \tilde{m} v^2} \right)$$

$$n_{21}(s) = \frac{c_f}{\tilde{J}} \left( \ell_f s + \frac{c_r \ell}{\tilde{m}v} \right)$$

$$n_{22}(s) = \frac{-c_r}{\tilde{J}} \left( \ell_r s + \frac{c_f \ell}{\tilde{m}v} \right)$$

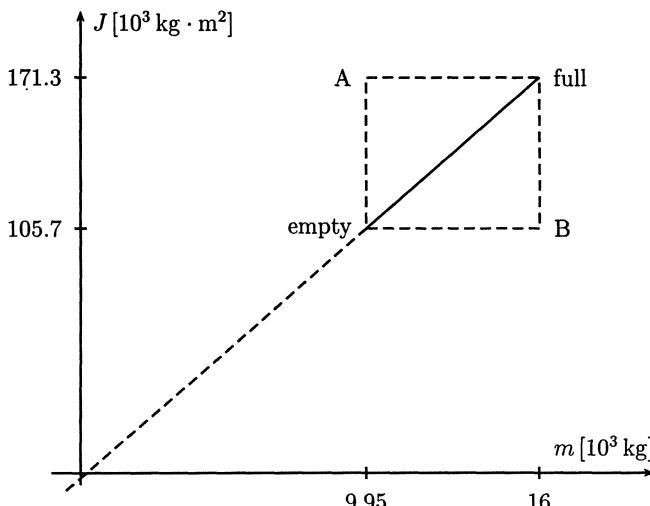
In the state space model (1.2.1) and in the transfer function model (1.2.8) the following seven parameters have been used:  $c_f, c_r, \tilde{m}, \tilde{J}, v, \ell_f$ , and  $\ell_r$ . Note that these uncertain parameters are not mutually independent. For example, the car has a fixed wheelbase

$$\ell = \ell_r + \ell_f \quad (1.2.9)$$

Also, the moment of inertia  $J$  and the mass  $m$  are not independent. Daimler-Benz gives the data of Table 1.1 for a city bus [49].

	Mass $m$ [kg]	Moment of inertia $J$ [kg · m <sup>2</sup> ]
Bus empty	9950	105700
Bus full	16000	171300

Table 1.1: Data for the bus O 305

Fig. 1.6: Relation between moment of inertia  $J$  and mass  $m$  for the bus O 305

In the plane of parameters  $m$  and  $J$  in Fig. 1.6 the given uncertainty intervals are depicted by a box with dashed lines. Obviously the operating conditions A and B in Fig. 1.6 cannot occur and should be not contained in an uncertainty model. Assume that the passengers are uniformly distributed over the bus, then  $J$  and  $m$  vary only along the straight line between the vertices *empty* and *full*. In other words, there is only one uncertain parameter  $m$  entering also into a known function  $J(m)$ . The dashed box in Fig. 1.6 “overbounds” the uncertainty.

We wish to avoid such overbounding and postulate the **basic rule number one of robust control**:

*Require robustness of a control system only for physically motivated parameter values and not with respect to arbitrarily assumed uncertainties of the mathematical model.*

*Remark 1.2.* The straight line connecting the *empty* and *full* cases in Fig. 1.6 (almost) goes through the origin. Its slope is  $i^2$ , where  $i$  is called the “inertial radius”, see [130];

thus,

$$J = i^2 m \quad (1.2.10)$$

For the bus O 305 we have  $i = 3.29$  [m]. The inertial radius is usually scaled by the wheelbase and for this case  $\ell = 5.60$  [m] and  $i/\ell = 0.59$ . For good maneuverability the bus has a short wheelbase in relation to the vehicle length. For the 31 passenger cars studied in [130] the scaled radius of inertia is in the range  $i/\ell \in [0.43; 0.53]$ .  $\square$

The uncertainty in  $\tilde{m} = m/\mu$  and  $\tilde{J} = J/\mu$  also captures the uncertainty in the road friction coefficient  $\mu$ , see (A.2.6). Thus, the virtual mass  $\tilde{m} = m/\mu$  may vary by a factor ten, for the velocity the factor  $v_{max}/v_{min}$  may be even larger, depending on the minimum speed  $v_{min}$  for which a robust control system has to operate. (Note that the car is not controllable for  $v = 0$ .)

### 1.3 Automatic Car Steering

Steering a car by hand means that the driver plans a path by preview and controls the lateral deviation of the vehicle from the planned path by the steering wheel. In an automatic steering system, this path following is automated. A guiding wire in the street may play the role of the planned path. The magnetic field from the guiding wire is measured by a sensor at the front end of the vehicle in order to determine the lateral deviation from the guiding wire. The deviation is kept small by feedback control via the steering motors. The reference trajectory may also be marked by magnetic nails in the street or it may be calculated from the data of a TV camera [55]. Automatic steering systems for city busses have been developed in Germany by Daimler-Benz [49] and MAN [162]. The potentially money saving idea is that such busses can drive on separate lanes that can be narrow and built more cheaply (tunnels, bridges).

Also, in the vicinity of big cities, separate lanes for specially equipped passenger cars are under discussion. Here the automatic steering system is combined with automatic distance control. The idea is to form car platoons of say 20 cars driving in a distance of 1 [m] from each other over a narrow lane. This is a possible alternative to building more and more lanes [170].

In order to study automation of car steering, the steering model must be extended. The extended model must include not only velocities, but also the vehicle heading and the lateral position of the sensor with respect to the reference path. For simplicity this extended model will only be derived using a linear model that is valid for small deviations from a stationary circular path. It is assumed that the reference path consists of circular arcs. Fig. 1.7 shows the transition from an arc with radius  $R_1$  and center  $M_1$  to an arc with radius  $R_2$  and center  $M_2$ . At the transition point the tangent to the path is continuous. There is, however, a step change in the reference input from  $R_{ref} = R_1$  to  $R_{ref} = R_2$ . For straight path segments the radius is  $R_{ref} = \infty$ . It is more convenient to introduce the curvature  $\rho_{ref} := 1/R_{ref}$  as the input that generates the

reference path. The curvature is defined positive for left cornering, see Fig. 1.7, and negative for right cornering.

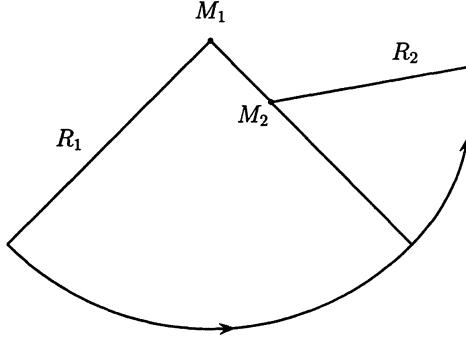


Fig. 1.7: The reference path is comprised of circular arcs

The vehicle motion in circular cornering will now be modelled for small deviations from a stationary circular path, see Fig. 1.8. The radial line from the center M passing through the center of gravity CG of the vehicle intersects a unique point  $z_{ref}$  on the desired path. The distance from the reference point  $z_{ref}$  to the CG is the deviation  $y_{CG}$ . Fig. 1.8 shows an inertially fixed coordinate system  $x_0, y_0$  and a car body fixed coordinate system  $x, y$  which is rotated by the yaw angle  $\psi$ . The tangent to the path at  $z_{ref}$  – denoted by  $\vec{v}_t$  – is rotated by a reference yaw angle  $\psi_t$  with respect to  $x_0$ .

A model for the rate of change of  $y_{CG}$  will now be developed. The component of the car velocity  $\vec{v}$  that is perpendicular to  $\vec{v}_t$  is equal to the rate of change of  $y_{CG}$ . This perpendicular component is given by  $v \sin(\beta + \Delta\psi)$  where  $\beta$  is the car sideslip angle and  $\Delta\psi := \psi - \psi_t$  is the angle between the tangent to the path at  $z_{ref}$  and the centerline of the car, see Fig. 1.8.

With the linearization  $\sin(\beta + \Delta\psi) \approx \beta + \Delta\psi$  the deviation  $y_{CG}$  changes according to

$$\dot{y}_{CG} = v(\beta + \Delta\psi) \quad (1.3.1)$$

Actually, the sensor S is not mounted in the CG but in a distance  $\ell_s$  in front of the CG with  $\ell_s \ll R_{ref}$ . The measured displacement  $y$  from the guiding wire now changes both with  $\dot{y}_{CG}$  and under the influence of the yaw rate  $r = \dot{\psi}$ . Taking this into account, the rate of change of the measured displacement is

$$\dot{y} = v(\beta + \Delta\psi) + \ell_s r \quad (1.3.2)$$

Determination of  $\dot{y}$  requires knowledge of three variables  $\beta, r$  and  $\Delta\psi$ . The variables  $\beta$ , and  $r$  are given by the basic car model (1.2.1). The angle  $\Delta\psi$  will be obtained by integrating its derivative

$$\begin{aligned} \Delta\dot{\psi} &= \dot{\psi} - \dot{\psi}_t \\ &= r - r_{st} \end{aligned}$$

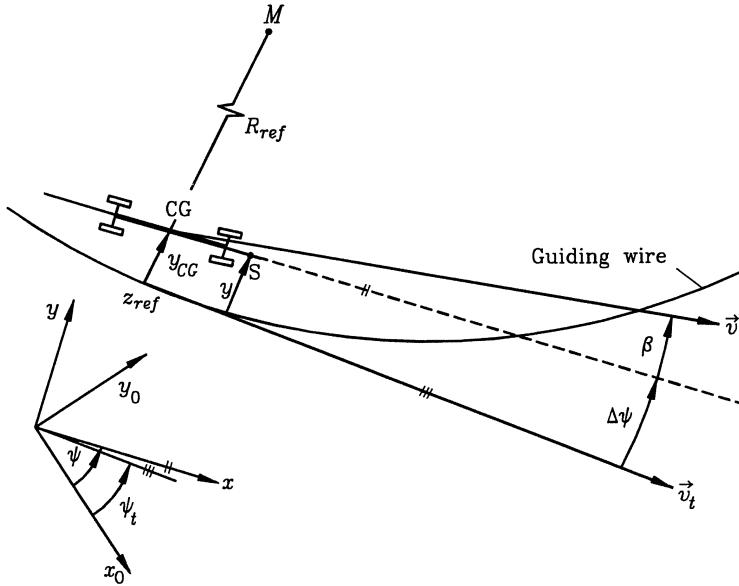


Fig. 1.8: Vehicle heading and measured displacement  $y$  from the guiding wire in stationary circular cornering

The term  $r_{st}$  is the yaw rate of the path tangent, i.e.  $r_{st} = v/R_{ref} = v\rho_{ref}$  in stationary circular cornering. Hence,

$$\Delta\dot{\psi} = r - v\rho_{ref} \quad (1.3.3)$$

Combining (1.2.1), (1.3.2), and (1.3.3), the extended state space model is obtained as

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \\ \Delta\dot{\psi} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ v & \ell_s & v & 0 \end{bmatrix} \begin{bmatrix} \beta \\ r \\ \Delta\psi \\ y \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ v \\ 0 \end{bmatrix} \rho_{ref} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (1.3.4)$$

In addition to the eigenvalues of (1.2.1) there is a double eigenvalue at  $s^2 = 0$ . The reference curvature  $\rho_{ref}$  appears as an additional input to the system. The transition to a new curvature corresponds to a step input in  $\rho_{ref}$ . Fig. 1.9 illustrates (1.3.4) in the form of a block diagram.

The transfer functions from the steering angles  $\delta_f, \delta_r$  to the measured displacement from the guiding wire  $y$  satisfy

$$y(s) = \frac{1}{(a_0 + a_1 s + s^2)s^2} \begin{bmatrix} n_f(s) & n_r(s) \end{bmatrix} \begin{bmatrix} \delta_f(s) \\ \delta_r(s) \end{bmatrix} \quad (1.3.5)$$

with  $a_0$  and  $a_1$  given in (1.2.2) and

$$n_f(s) = v[sn_{11}(s) + n_{21}(s)] + \ell_s sn_{21}(s)$$

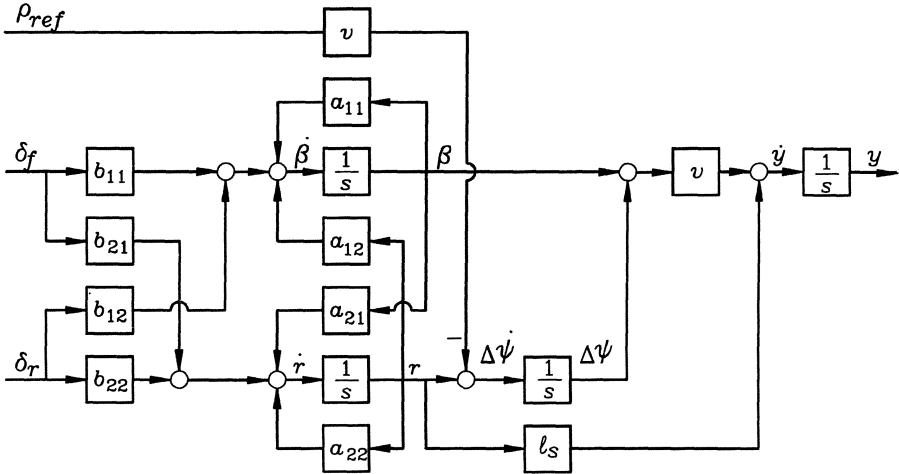


Fig. 1.9: Block diagram of automatic steering with reference curvature  $\rho_{ref}$

$$n_r(s) = v[s n_{12}(s) + n_{22}(s)] + \ell_s s n_{22}(s)$$

given by (1.2.8) and (1.3.4).

In later chapters we will come back to the design of a robust automatic steering controller for a bus with front wheel steering only ( $\delta_r \equiv 0$ ). The front wheel steering angle  $\delta_f$  is generated by an integrating actuator, i.e.

$$\dot{\delta}_f = u_f \quad (1.3.6)$$

Thus, the plant transfer function will be

$$g(s) = \frac{y(s)}{u_f(s)} = \frac{n_f(s)}{(a_0 + a_1 s + s^2)s^3} \quad (1.3.7)$$

$$n_f(s) = \frac{c_f}{m} \left[ \left( 1 + \frac{\ell_s \ell_f}{i^2} \right) s^2 + \frac{c_r \ell (\ell_r + \ell_s)}{i^2 \tilde{m} v} s + \frac{c_r \ell}{i^2 \tilde{m}} \right] \quad (1.3.8)$$

with  $a_0$  and  $a_1$  are given in (1.2.2). Multiplication of numerator and denominator by  $i^2 \tilde{m}^2 v^2$  yields

$$g(s) = \frac{c_f v (e_0 + e_1 s + e_2 s^2)}{(d_0 + d_1 s + d_2 s^2) s^3} \quad (1.3.9)$$

$$e_0 = c_r \ell v$$

$$e_1 = c_r \ell (\ell_r + \ell_s)$$

$$e_2 = (i^2 + \ell_s \ell_f) \tilde{m} v$$

$$d_0 = c_f c_r \ell^2 + (c_r \ell_r - c_f \ell_f) \tilde{m} v^2$$

$$d_1 = [(c_f + c_r) i^2 + (c_f \ell_f^2 + c_r \ell_r^2)] \tilde{m} v$$

$$d_2 = i^2 \tilde{m}^2 v^2$$

Note that the numerator and denominator coefficients of the plant transfer function (1.3.9) contain the terms  $v, v^2, \tilde{m}v, \tilde{m}v^2$ , and  $\tilde{m}^2v^2$ .

## 1.4 A Flight Control Problem

Models of aircraft dynamics and their linearization can be found in standard books on aircraft dynamics and control, e.g. [61, 126, 161]. When treated as a rigid body, an aircraft has six degrees of freedom of motion: three coordinates of the center of gravity position and three rotation coordinates for roll, yaw, and pitch. In stationary flight, two subsystems may be considered as decoupled:

1. the longitudinal motion in pitch, altitude, and longitudinal position,
2. the lateral motion in roll, yaw, and sideward position.

Consider the first subsystem. It is characterized by a slow oscillation of the CG around its trajectory (phugoid) and a much faster pitch motion of the fuselage around its CG. The latter is called the “short period longitudinal mode”. Both modes are weakly coupled. We consider here only the second mode. It can be measured by a gyro and an accelerometer and it can be controlled by the elevator rudder.

Consider the fighter aircraft F4-E of Fig. 1.10.

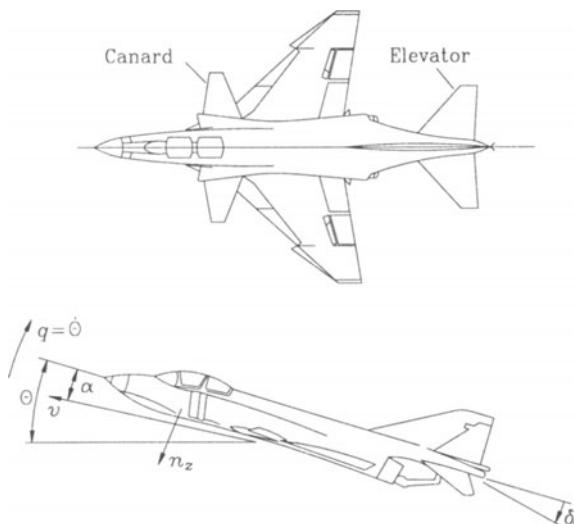


Fig. 1.10: Fighter aircraft F4-E

The F4-E is a modified military aircraft. In particular, the maneuverability was increased by additional horizontal canards. This results in a loss of longitudinal aerodynamic stability, however. The short period mode is unstable in subsonic flight and only weakly damped in supersonic flight. The equations of motion are linearized for small deviations from a stationary controlled flight (i.e. constant altitude and velocity, small angle of attack  $\alpha$ ). In flight mechanics, it is usual to take  $\alpha$  and the pitch rate  $q$  as state variables. Here we transform the equations to sensor coordinates, i.e. the normal acceleration  $x_1 = n_z$  and the pitch rate  $x_2 = q$  are introduced as states. This simplifies the design for robustness with respect to accelerometer failure [4, 68]. The actuator for the elevator is modelled as a low pass filter with the transfer function  $14/(s + 14)$ . Its state variable is  $\delta_e$ . For the state vector, we take

$$\mathbf{x} = [n_z \ q \ \delta_e]^T \quad (1.4.1)$$

and the linearized state equations

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{bu}$$

have the following structure

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & -14 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ 0 \\ 14 \end{bmatrix} \quad (1.4.2)$$

*Remark 1.3.* The model (1.4.2) is based on some simplifying assumptions.

- a) Elevator ( $\delta_e$ ) and the canard rudder ( $\delta_c$ ) are not used independently of each other for small deviations from stationary flight. The two commanded input variables are coupled by

$$\begin{aligned} \delta_{\text{ecom}} &= u \\ \delta_{\text{ccom}} &= -0.7u \end{aligned} \quad (1.4.3)$$

The factor  $-0.7$  was chosen for minimum drag. Therefore, the system (1.4.2) has only one input  $u$ .

- b) Structural vibrations are not modelled. The bandwidth for the rigid body control is limited to be below the first structural mode frequency of 85 [rad/sec] to avoid excitation of structural modes.

□

The possible flight conditions of this aircraft are represented in the Mach-altitude number diagram Fig. 1.11.

Numerical values for four representative flight conditions (FC) as indicated in Fig. 1.11 have been taken from [36] and were transformed to the state equation (1.4.2). They are listed together with the respective eigenvalues  $s_1, s_2$  in Table 1.2. The third eigenvalue of (1.4.2) is fixed at  $s_3 = -14$ .

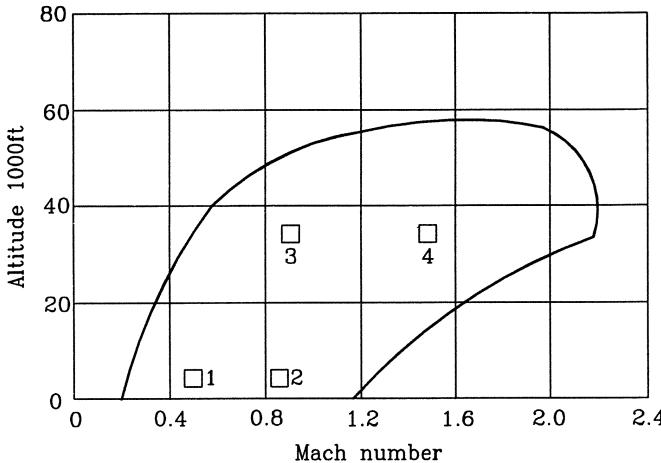


Fig. 1.11: Envelope of the possible flight conditions and four representative cases

## 1.5 Notation for Uncertain Plants

In Chapter 1, examples of plants with uncertain real parameters  $\mathbf{q} = [q_1 \ q_2 \ \dots \ q_\ell]^T$  have been introduced. For the loading bridge three uncertain parameters were considered,  $q_1 = \ell$ ,  $q_2 = m_L$ , and  $q_3 = m_C$ , for the car  $q_1 = v$ ,  $q_2 = \tilde{m} = m/\mu$  were introduced and for the flight control problem the uncertain parameters were  $q_1 = \text{Mach number}$  and  $q_2 = \text{altitude}$ .

The dependency of a state space model on the uncertain parameter vector  $\mathbf{q}$  is written in a general form

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{q})\mathbf{x}(t) + \mathbf{B}(\mathbf{q})\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{q})\mathbf{x}(t)\end{aligned}\tag{1.5.1}$$

and the notation for the  $\mathbf{q}$ -dependency of a transfer function model is

$$\mathbf{y}(s) = \mathbf{G}(s, \mathbf{q})\mathbf{u}(s)\tag{1.5.2}$$

Without further specification of  $\mathbf{q}$ , (1.5.1) and (1.5.2) are called “uncertain plant models”. If a parameter set  $Q$  is specified, then (1.5.1) or (1.5.2) with  $\mathbf{q} \in Q$  is called a “plant family”. An example for  $Q$  is the flight envelope of Fig. 1.11. In some problems  $Q$  is represented by only a finite number of operating points  $\mathbf{q}^{(j)} \in Q$ , i.e.

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}^{(j)}\mathbf{x} + \mathbf{B}^{(j)}\mathbf{u}, \quad j = 1, 2, \dots, J \\ \mathbf{y} &= \mathbf{C}^{(j)}\mathbf{x}\end{aligned}\tag{1.5.3}$$

where

$$\mathbf{A}^{(j)} := \mathbf{A}(\mathbf{q}^{(j)}), \quad \mathbf{B}^{(j)} := \mathbf{B}(\mathbf{q}^{(j)}), \quad \mathbf{C}^{(j)} := \mathbf{C}(\mathbf{q}^{(j)})\tag{1.5.4}$$

Mach Altitude [ft]	FC 1 0.5 5000	FC 2 0.85 5000	FC 3 0.9 35000	FC 4 1.5 35000
$a_{11}$	-0.9896	-1.702	-0.667	-0.5162
$a_{12}$	17.41	50.72	18.11	26.96
$a_{13}$	96.15	263.5	84.34	178.9
$a_{21}$	0.2648	0.2201	0.08201	-0.6896
$a_{22}$	-0.8512	-1.418	-0.6587	-1.225
$a_{23}$	-11.39	-31.99	-10.81	-30.38
$b_1$	-97.78	-272.2	-85.09	-175.6
$s_1$	-3.07	-4.90	-1.87	$-0.87 \pm j4.3$
$s_2$	1.23	1.78	0.56	

Table 1.2: Model data for an F4-E aircraft for four typical flight conditions. The eigenvalues  $s_1$  and  $s_2$  result from  $(s - a_{11})(s - a_{22}) - a_{12}a_{21} = 0$

Alternatively, a finite family of transfer function matrices is denoted by

$$\mathbf{G}^{(j)}(s) = \mathbf{C}^{(j)}(s\mathbf{I} - \mathbf{A}^{(j)})^{-1}\mathbf{B}^{(j)}, \quad j = 1, 2, \dots, J \quad (1.5.5)$$

Now (1.5.3) or (1.5.5) is called a “finite plant family” .

In many practical examples, it is justified to assume that the uncertain parameters  $q_i$  are piecewise constant or slowly time varying (compared to the system dynamics). The load mass of a crane is constant during a transport process. In one process it may be the mass of the empty hook, in the next process it may be maximum load mass for which the crane was built. The speed of a car is changing slowly compared to the period of the yaw motion. The cornering stiffness may change suddenly when the car drives on an icy bridge. Here we consider a new dynamical system with constant parameters and initial conditions that are inherited from the end conditions of the preceding system. We assume that parameters are constant but unknown. Frequently, the uncertainty of a parameter  $q_i$  can be described by its lower and upper bounds  $q_i^-$  and  $q_i^+$ . We write

$$q_i \in [q_i^- ; q_i^+] \quad (1.5.6)$$

The parameter  $q_i$  is then called “interval parameter”. The most common example of an operating domain is the hyperrectangle (“parameter box”)

$$Q = \left\{ \mathbf{q} \mid q_i \in [q_i^- ; q_i^+], \quad i = 1, 2, \dots, \ell \right\} \quad (1.5.7)$$

For  $\ell = 2$  parameters the box is illustrated by Fig. 1.12.

In this book we deal almost exclusively with the parameter box (1.5.7), the few exceptions are indicated accordingly.

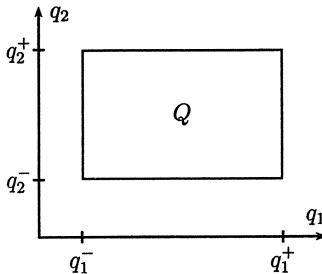


Fig. 1.12: The operating domain  $Q$  of two independent parameters  $q_1$  and  $q_2$

## 1.6 Exercises

- 1.1. a) Derive the model (1.1.1) of the crane.  
b) Now assume that the rope length  $\ell$  may vary during a transport operation. Which additional terms occur in the model?
- 1.2. Derive the state equations for the “inverted pendulum”. It corresponds to the crane, where the rope is replaced by a massless rod. The load mass  $m_L$  can be attached to the rod in any distance  $\ell$  from the joint. In the equilibrium position the load mass is vertically above the vehicle. What is the transfer function from the force  $u$  at the vehicle to the position of the vehicle?
- 1.3. Rewrite the car steering model using the lateral speed  $v_y = v \sin \beta \approx v\beta$  instead of  $\beta$  as state variable. Compare the type of coefficient functions in the  $\mathbf{A}$  and  $\mathbf{B}$  matrices with that of (1.2.1).
- 1.4. Plot the eigenvalue and zero region for a bus O 305 (see (1.3.9)) with the data given in Table 1.3. The range of the virtual mass  $\tilde{m}$  results from a vehicle mass range [9950 ; 16000] [kg] and a friction coefficient range  $\mu \in [0.5 ; 1]$ .

$$\begin{aligned}\ell_f &= 3.67 \text{ [m]}, \\ \ell_r &= 1.93 \text{ [m]}, \\ \ell_s &= 6.12 \text{ [m]}, \\ c_f &= 198000 \text{ [N/rad]}, \\ c_r &= 470000 \text{ [N/rad]}, \\ v &\in [1 ; 20] \text{ [m} \cdot \text{s}^{-1}], \\ \tilde{m} &\in [9950 ; 32000] \text{ [kg]}, \\ \tilde{J} &= i^2 \tilde{m}, \quad i^2 = 10.85 \text{ [m}^2]\end{aligned}$$

Table 1.3: Data for the city bus O 305

## 2 Control System Structures

The examples of Chapter 1 have in common that the plant is insufficiently damped or even unstable in its operating domain  $Q$ . Thus, a primary task of a control system is stabilization with sufficient damping for all  $q \in Q$ . Since eigenvalues cannot be shifted by a feedforward control system, we need a feedback structure. Fig. 2.1 shows an example of a control system with a plant family  $G(s, Q) = \{g(s, q) \mid q \in Q\}$ , a feedback compensator (or controller)  $c(s)$ , and a feedforward path (or prefilter)  $f(s)$ .

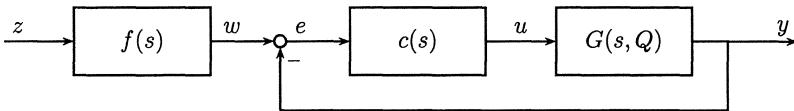


Fig. 2.1: Control system with a plant family  $G(s, Q)$

In this book we are primarily concerned with the analysis of the closed loop with input  $w$  and output  $y$  and with the design of a compensator  $c(s)$ . It is this loop, where the effects of the uncertainty of  $q$  must be reduced by a robust design.

*Remark 2.1.* There are more reasons to use a feedback structure. The compensator  $c(s)$  can also reduce the effects of unmodelled uncertainties and of disturbances acting on the plant.  $\square$

A difference between the classical stabilization problem for a fixed plant and the robust stabilization of a plant family  $G(s, Q)$  is illustrated in the following example.

*Example 2.1.* Consider the finite plant family

$$g^{(1)}(s) = \frac{1}{s-1}, \quad g^{(2)}(s) = \frac{-1}{s-1} \quad (2.0.1)$$

and assume proportional feedback  $c(s) = k$ . Obviously,  $g^{(1)}(s)$  can be stabilized by  $k > 1$  and  $g^{(2)}(s)$  can be stabilized by  $k < -1$ , but there is no  $k$  that stabilizes both plants simultaneously. If we assume a compensator  $c(s) = (c_0 + c_1 s + \dots + c_m s^m)/(d_0 + d_1 s + \dots + d_m s^m)$  and write the plant family as  $g(s, q) = q/(s-1)$ ,  $q \in \{1, -1\}$  then the coefficients of the closed-loop characteristic polynomial are

$$\begin{aligned}
a_0 &= c_0 q - d_0 \\
a_1 &= c_1 q + d_0 - d_1 \\
&\vdots \\
a_m &= c_m q + d_{m-1} - d_m \\
a_{m+1} &= d_m
\end{aligned}$$

Now the necessary stability conditions  $a_i > 0$ ,  $i = 0, 1, \dots, m$ , imply  $d_i < 0$ ,  $i = 0, 1, \dots, m$ , but  $d_m < 0$  contradicts the necessary stability condition  $a_{m+1} > 0$ . Thus, the two plants (2.0.1) cannot be simultaneously stabilized by any linear compensator.  $\square$

The example gives rise to a fundamental question of robust control: Given a plant family  $G(s, Q)$ , does there exist a fixed compensator  $c(s)$  that stabilizes the plant family?

Unfortunately, a complete answer to this question is not yet known. Thus, in a scientific sense, it is premature to synthesize a robust compensator  $c(s)$ . On the other hand there are many control systems in operation that show remarkable robustness with respect to uncertainty in the parameters  $q$  and there is a practical need to analyze and design such control systems. Also, there is good reason to believe that practical examples such as those described in Chapter 1 are not as nasty as Example 2.1. As a consequence, the assumption of a controller structure is presently more an art than a science.

In this chapter, we deal with controller structures primarily in conjunction with the examples of Chapter 1. Several alternative controller structures will be assumed and the closed-loop characteristic polynomial will be calculated as a function of both the plant and the controller parameters. Such uncertain polynomials will be analyzed in later chapters.

## 2.1 Robust Controllability and Observability

It is assumed that the reader is familiar with the notions of controllability and observability of linear state-space systems, see for example [99]. In the parameter-dependent case we are particularly interested in operating conditions  $q$  for which such properties are lost. We review the controllability and observability conditions and apply them to some examples.

Consider a plant family in state-space notation

$$\begin{aligned}
\dot{x} &= A(q)x + B(q)u \\
y &= C(q)x
\end{aligned} \tag{2.1.1}$$

with  $q \in Q$ . The pair  $(A(q), B(q))$  is robustly controllable if

$$\text{rank } [B(q) \ A(q)B(q) \ \dots \ A^{n-1}(q)B(q)] = n \text{ for all } q \in Q \tag{2.1.2}$$

*Example 2.2.* The controllability matrix of the crane is by (1.1.6)

$$\begin{bmatrix} \mathbf{b} & \mathbf{Ab} & \mathbf{A}^2\mathbf{b} & \mathbf{A}^3\mathbf{b} \end{bmatrix} = \begin{bmatrix} 0 & b_2 & 0 & b_4a_{23} \\ b_2 & 0 & b_4a_{23} & 0 \\ 0 & b_4 & 0 & b_4a_{43} \\ b_4 & 0 & b_4a_{43} & 0 \end{bmatrix} \quad (2.1.3)$$

(For notational convenience the dependency of all terms on  $\mathbf{q}$  is not explicitly indicated.) The determinant of the controllability matrix is

$$\det [\mathbf{b} \quad \mathbf{Ab} \quad \mathbf{A}^2\mathbf{b} \quad \mathbf{A}^3\mathbf{b}] = b_4^2(b_2a_{43} - b_4a_{23})^2 \quad (2.1.4)$$

It vanishes for

$$b_4 = -1/m_C\ell = 0$$

and for

$$b_2a_{43} - b_4a_{23} = -g/m_C\ell = 0$$

Controllability is lost only under zero gravity.  $\square$

*Example 2.3.* For front wheel steering of the car (1.2.1), the controllability matrix is

$$[\mathbf{b} \quad \mathbf{Ab}] = \begin{bmatrix} b_{11} & a_{11}b_{11} + a_{12}b_{21} \\ b_{21} & a_{21}b_{11} + a_{22}b_{21} \end{bmatrix}$$

and its determinant is

$$\begin{aligned} \det [\mathbf{b} \quad \mathbf{Ab}] &= b_{11}(a_{21}b_{11} + a_{22}b_{21}) - b_{21}(a_{11}b_{11} + a_{12}b_{21}) \\ &= \frac{c_f^2}{\tilde{m}^2 \tilde{J}^2 v^2} \{ \ell_f^2 \tilde{m}^2 v^2 + \ell_c \tilde{J} - \ell_f \ell_r \ell_c \tilde{m} \} \end{aligned} \quad (2.1.5)$$

Controllability is lost for  $c_f = 0$  and at a speed  $v_{nc}$ , where

$$v_{nc}^2 = \frac{\ell_c(\ell_f \ell_r \tilde{m} - \tilde{J})}{\ell_f^2 \tilde{m}^2} \quad (2.1.6)$$

If  $\tilde{J} > \tilde{m}\ell_f \ell_r$ , then  $v_{nc}$  is imaginary and of no practical interest. If, however,

$$\tilde{J} < \tilde{m}\ell_f \ell_r \quad (2.1.7)$$

then uncontrollability occurs at a real velocity. Substituting  $v^2 = v_{nc}^2$  into  $a_0$  of (1.2.2) and assuming (2.1.7),  $a_0 > 0$  is obtained. Since also  $a_1 > 0$ , the uncontrollable modes are stable. Thus, the steering dynamics is robustly stabilizable, but not robustly controllable.  $\square$

*Example 2.4.* Consider the convex combination of the two plants of Example 2.1

$$g(s, q) = \frac{q}{s-1}, \quad q \in [-1; 1] \quad (2.1.8)$$

The unstable mode of the plant is not controllable or not observable for  $q = 0$ . Thus, the plant family is not stabilizable.  $\square$

The pair  $(C(q), A(q))$  is robustly observable if

$$\text{rank} \begin{bmatrix} C(q) \\ C(q)A(q) \\ \vdots \\ C(q)A^{n-1}(q) \end{bmatrix} = n \text{ for all } q \in Q \quad (2.1.9)$$

*Example 2.5.* The observability matrix for the crane with output  $y_1 = c^T x = [1 \ 0 \ 0 \ 0]x$  is

$$\begin{bmatrix} c^T \\ c^T A \\ c^T A^2 \\ c^T A^3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & a_{23} \end{bmatrix} \quad (2.1.10)$$

As  $m_L \rightarrow 0$ , then  $a_{23} = m_L g / m_C \rightarrow 0$ , see (1.1.6), and the rank of the matrix drops from four to two at  $m_L = 0$ . For small  $m_L$  the system is “almost unobservable”. Physically this means: For the empty hook case the horizontal force transmitted from the load to the crab is so small that its effect is hardly recognizable in the measurements of  $x_1$  and  $x_2$ .  $\square$

*Example 2.6.* For the crane let

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

i.e. all state variables except  $x_1$  are measured.

$$CA = \begin{bmatrix} 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix}$$

In  $C$ ,  $CA$ , and also for all further  $CA^i, i = 2, 3, \dots$ , the first column is zero, i.e. the plant is not observable. More specifically, the crab position  $x_1$  is not observable and one of the eigenvalues at  $s = 0$  cannot be moved by output feedback  $u = -KCx$ . For a stabilizing feedback structure the crab position  $x_1$  must be measured and fed back.

## 2.2 State and Output Feedback

Consider a state-space model of a single-input plant family

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}(\mathbf{q})\mathbf{x} + \mathbf{b}(\mathbf{q})u, \quad \mathbf{q} \in Q \\ \mathbf{y} &= \mathbf{C}(\mathbf{q})\mathbf{x}\end{aligned}\quad (2.2.1)$$

and the control law

$$u = -\mathbf{k}^T \mathbf{y} + w \quad (2.2.2)$$

see Fig. 2.2. State feedback corresponds to the case  $\mathbf{C}(\mathbf{q}) = \mathbf{I}$ .

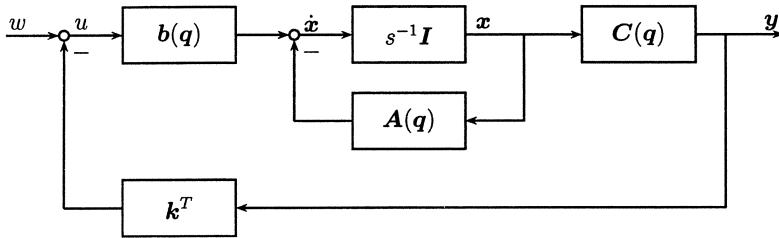


Fig. 2.2: Output feedback system

The closed-loop characteristic polynomial is

$$p(s, \mathbf{q}, \mathbf{k}) = \det [s\mathbf{I} - \mathbf{A}(\mathbf{q}) + \mathbf{b}(\mathbf{q})\mathbf{k}^T \mathbf{C}(\mathbf{q})] \quad (2.2.3)$$

For a given  $\mathbf{q}$  and  $\mathbf{k}$  it is easy to compute the eigenvalues of  $\mathbf{A}(\mathbf{q}) - \mathbf{b}(\mathbf{q})\mathbf{k}^T \mathbf{C}(\mathbf{q})$  numerically. For calculations in general  $\mathbf{q}$  and  $\mathbf{k}$  a symbolic manipulation program may be used to evaluate  $p(s, \mathbf{q}, \mathbf{k})$ .

The simple examples of Chapter 1 will be studied by hand calculations. For this purpose and for later mapping of stability boundaries into  $\mathbf{k}$ -space, it is helpful to write (2.2.3) in a form not involving a determinant.

Consider the loop of Fig. 2.2 broken at  $u$ . Then the open-loop transfer function is

$$g_0(s, \mathbf{q}, \mathbf{k}) = \mathbf{k}^T \mathbf{C}(\mathbf{q}) [s\mathbf{I} - \mathbf{A}(\mathbf{q})]^{-1} \mathbf{b}(\mathbf{q}) \quad (2.2.4)$$

and the closed-loop characteristic equation may be written as

$$p(s, \mathbf{q}, \mathbf{k}) = \text{numerator } \{1 + g_0(s, \mathbf{q}, \mathbf{k})\} \quad (2.2.5)$$

Now,

$$[s\mathbf{I} - \mathbf{A}(\mathbf{q})]^{-1} = \frac{\text{adj}\{s\mathbf{I} - \mathbf{A}(\mathbf{q})\}}{p_0(s, \mathbf{q})}$$

where

$$p_0(s, \mathbf{q}) = \det[s\mathbf{I} - \mathbf{A}(\mathbf{q})] = a_{00} + a_{01}s + \dots + a_{0n-1}s^{n-1} + s^n \quad (2.2.6)$$

is the open-loop characteristic polynomial and (2.2.5) becomes

$$p(s, \mathbf{q}, \mathbf{k}) = p_0(s, \mathbf{q}) + \mathbf{k}^T \mathbf{M}(s, \mathbf{q}) \quad (2.2.7)$$

with

$$\mathbf{M}(s, \mathbf{q}) := \mathbf{C}(\mathbf{q}) \operatorname{adj}\{s\mathbf{I} - \mathbf{A}(\mathbf{q})\} \mathbf{b}(\mathbf{q})$$

For symbolic hand calculations the Leverrier algorithm [73] is recommended, because the last step provides a check. By this algorithm (omitting  $\mathbf{q}$ )

$$\operatorname{adj}\{s\mathbf{I} - \mathbf{A}\} = \mathbf{D}_0 + \mathbf{D}_1 s + \dots + \mathbf{D}_{n-1} s^{n-1} \quad (2.2.8)$$

where

$$\begin{aligned} \mathbf{D}_{n-1} &= \mathbf{I} \\ a_{0n-1} &= -\frac{1}{1} \operatorname{trace} \mathbf{AD}_{n-1}, \quad \mathbf{D}_{n-2} = \mathbf{AD}_{n-1} + a_{0n-1}\mathbf{I} \\ a_{0n-2} &= -\frac{1}{2} \operatorname{trace} \mathbf{AD}_{n-2}, \quad \mathbf{D}_{n-3} = \mathbf{AD}_{n-2} + a_{0n-2}\mathbf{I} \\ &\vdots \\ a_{01} &= -\frac{1}{n-1} \operatorname{trace} \mathbf{AD}_1, \quad \mathbf{D}_0 = \mathbf{AD}_1 + a_{01}\mathbf{I} \\ a_{00} &= -\frac{1}{n} \operatorname{trace} \mathbf{AD}_0, \quad \mathbf{D}_{-1} = \mathbf{AD}_0 + a_{00}\mathbf{I} = \mathbf{0} \end{aligned}$$

The last equation  $\mathbf{D}_{-1} = \mathbf{0}$  serves as a check. The algorithm also yields the coefficients  $a_{0i}$ ,  $i = 0, 1, \dots, n-1$  of the open-loop characteristic polynomial (2.2.6).

The polynomial equation (2.2.7) may also be written in terms of its coefficients vectors. The coefficient vector of the monic (i.e.  $a_{0n} = 1$ ) open-loop polynomial  $p_0$  is

$$\hat{\mathbf{a}}_0^T = [a_{00} \ a_{01} \ \dots \ a_{0n-1}] \quad (2.2.9)$$

while for the monic (i.e.  $a_n = 1$ ) closed-loop polynomial  $p$  the coefficient vector is

$$\hat{\mathbf{a}}^T = [a_0 \ a_1 \ \dots \ a_{n-1}] \quad (2.2.10)$$

Then (2.2.7) reads

$$\begin{bmatrix} \hat{\mathbf{a}}^T & 1 \end{bmatrix} \begin{bmatrix} 1 \\ s \\ \vdots \\ s^n \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{a}}_0^T & 1 \end{bmatrix} \begin{bmatrix} 1 \\ s \\ \vdots \\ s^n \end{bmatrix} + \mathbf{k}^T \mathbf{C} \begin{bmatrix} \mathbf{D}_0 \mathbf{b} & \mathbf{D}_1 \mathbf{b} & \dots & \mathbf{D}_{n-1} \mathbf{b} \end{bmatrix} \begin{bmatrix} 1 \\ s \\ \vdots \\ s^{n-1} \end{bmatrix} \quad (2.2.11)$$

By matching the coefficients of the powers of  $s$  we get

$$\hat{\mathbf{a}}^T = \hat{\mathbf{a}}_0^T + \mathbf{k}^T \mathbf{W} \quad (2.2.12)$$

where

$$\mathbf{W} = \mathbf{C}[\mathbf{D}_0 \mathbf{b} \ \mathbf{D}_1 \mathbf{b} \ \dots \ \mathbf{D}_{n-1} \mathbf{b}] \quad (2.2.13)$$

Thus, the feedback gain  $\mathbf{k}^T$  enters affinely into the coefficients of the closed-loop characteristic polynomial.  $\square$

*Remark 2.2.* The notation  $\hat{\mathbf{a}}$  for the coefficient vector refers to the coefficients of the monic polynomial  $p(s) = [1 \ s \ \dots \ s^n] \begin{bmatrix} \hat{\mathbf{a}} \\ 1 \end{bmatrix}$ , i.e.  $\hat{\mathbf{a}}$  lives in an  $n$ -dimensional space. For parameter-dependent polynomials it is often more convenient to write the polynomial in a non-monic form  $p(s) = [1 \ s \ \dots \ s^n] \mathbf{a}$  with  $\mathbf{a} = [a_0 \ a_1 \ \dots \ a_n]$  a vector in an  $(n+1)$ -dimensional space.  $\square$

### Example 2.7. Crane

Consider the motion of the crane (1.1.6) in the neighborhood of an equilibrium point  $\mathbf{x} = [w \ 0 \ 0 \ 0]^T$ , where  $w$  is a constant reference input for the crab position. The crane is controlled by the state feedback

$$u = k_1(w - x_1) - k_2x_2 - k_3x_3 - k_4x_4 \quad (2.2.14)$$

By (1.1.6) the matrices of the state-space model are

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 0 \\ b_2 \\ 0 \\ b_4 \end{bmatrix}$$

$$a_{23} = \frac{m_L}{m_C} g, \quad b_2 = \frac{1}{m_C}$$

$$a_{43} = -\frac{(m_L + m_C)g}{m_C \ell}, \quad b_4 = -\frac{1}{m_C \ell}$$

and  $\mathbf{C} = \mathbf{I}$  for state feedback. The Leverrier algorithm yields

$$\mathbf{D}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$a_{03} = -\text{trace} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix} = 0 \quad \mathbf{D}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix}$$

$$a_{02} = -\frac{1}{2} \text{trace} \begin{bmatrix} 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & a_{23} \\ 0 & 0 & a_{43} & 0 \\ 0 & 0 & 0 & a_{43} \end{bmatrix} = -a_{43}, \quad \mathbf{D}_1 = \begin{bmatrix} -a_{43} & 0 & a_{23} & 0 \\ 0 & -a_{43} & 0 & a_{23} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$a_{01} = -\frac{1}{3} \text{trace} \begin{bmatrix} 0 & -a_{43} & 0 & a_{23} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0, \quad \mathbf{D}_0 = \begin{bmatrix} 0 & -a_{43} & 0 & a_{23} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$a_{00} = -\frac{1}{4} \text{trace} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = 0, \quad \mathbf{D}_{-1} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The check  $\mathbf{D}_{-1} = \mathbf{0}$  is reassuring after such hand calculations. Now,

$$\begin{aligned} \mathbf{W} &= [\mathbf{D}_0 \mathbf{b} \ \mathbf{D}_1 \mathbf{b} \ \mathbf{D}_2 \mathbf{b} \ \mathbf{D}_3 \mathbf{b}] \\ &= \begin{bmatrix} -a_{43}b_2 + a_{23}b_4 & 0 & b_2 & 0 \\ 0 & -a_{43}b_2 + a_{23}b_4 & 0 & b_2 \\ 0 & 0 & b_4 & 0 \\ 0 & 0 & 0 & b_4 \end{bmatrix} \end{aligned}$$

$$\begin{aligned}
&= \begin{bmatrix} g/m_C\ell & 0 & 1/m_C & 0 \\ 0 & g/m_C\ell & 0 & 1/m_C \\ 0 & 0 & -1/m_C\ell & 0 \\ 0 & 0 & 0 & -1/m_C\ell \end{bmatrix} \\
&= \frac{1}{m_C\ell} \begin{bmatrix} g & 0 & \ell & 0 \\ 0 & g & 0 & \ell \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \tag{2.2.15}
\end{aligned}$$

The closed-loop characteristic coefficient vector is by (2.2.12)

$$\begin{aligned}
\hat{\mathbf{a}}^T &= \left[ 0 \ 0 \ \frac{(m_L + m_C)g}{m_C\ell} \ 0 \right] + \\
&\quad + [k_1 \ k_2 \ k_3 \ k_4] \begin{bmatrix} g & 0 & \ell & 0 \\ 0 & g & 0 & \ell \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \frac{1}{\ell m_C} \\
&= \frac{1}{\ell m_C} [k_1 g \ k_2 g \ (m_L + m_C)g + k_1 \ell - k_3 \ k_2 \ell - k_4] \tag{2.2.16}
\end{aligned}$$

For  $\ell > 0$ ,  $m_C > 0$ , we can replace the monic polynomial  $p(s)$  with rational coefficient functions by the non-monic polynomial

$$\begin{aligned}
\bar{p}(s) := \ell m_C p(s) &= a_0 + a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 \tag{2.2.17} \\
a_0 &= k_1 g \\
a_1 &= k_2 g \\
a_2 &= (m_L + m_C)g + k_1 \ell - k_3 \\
a_3 &= k_2 \ell - k_4 \\
a_4 &= \ell m_C
\end{aligned}$$

Note that  $\bar{p}(s)$  and  $p(s)$  have the same roots, however, the coefficient functions of  $\bar{p}(s)$  are multilinear in the uncertain parameters. This fourth order example is simple enough such that stability can be analyzed by the Hurwitz stability conditions. Necessary and sufficient conditions are that

a) all coefficients  $a_i$  are positive, i.e.

$$\begin{aligned}
a_0 > 0 &\implies k_1 > 0 \\
a_1 > 0 &\implies k_2 > 0 \\
a_2 > 0 &\implies k_3 < (m_L + m_C)g + k_1 \ell \\
a_3 > 0 &\implies k_4 < k_2 \ell
\end{aligned} \tag{2.2.18}$$

$$\text{b) } \Delta_3 = \det \mathbf{H}_3 = \begin{vmatrix} a_3 & a_1 & 0 \\ a_4 & a_2 & a_0 \\ 0 & a_3 & a_1 \end{vmatrix} > 0$$

The evaluation yields

$$\Delta_3 = [k_2(m_L g - k_3) + k_1 k_4](k_2 \ell - k_4) - k_2 k_4 m_C g > 0 \quad (2.2.19)$$

An interesting special case results for  $k_3 = 0$ ,  $k_4 = 0$ . Then

$$\Delta_3 = k_2^2 \ell m_L g > 0 \quad (2.2.20)$$

i.e. the output feedback controller

$$k_1 > 0, \quad k_2 > 0, \quad k_3 = 0, \quad k_4 = 0 \quad (2.2.21)$$

stabilizes all cranes as modelled by (1.1.6).

If we are only interested in stability, then the task is completely solved for all cranes. But a well designed control system should be more than just stable. In Chapter 9 we will discuss Gamma-stability, that is the property that all eigenvalues are located in a specified region  $\Gamma$  in the complex  $s$ -plane.  $\square$

*Example 2.8.*  $\Gamma$ -stability of the crane

Here we introduce a simple example in order to discuss some nontrivial robust  $\Gamma$ -stability problems for the crane in the next chapters. For the polynomial (2.2.18) with  $k_3 = 0$ ,  $k_4 = 0$ , we require a stability margin in which the real part of all eigenvalues are smaller than  $-a$ . The half plane  $\text{Re } s < -a$  is mapped via

$$v = s + a \quad (2.2.22)$$

onto the left half plane of the new complex variable  $v$ . The necessary and sufficient robust  $\Gamma$ -stability condition is that the new polynomial in  $v$  is robustly Hurwitz stable. This new polynomial is

$$\begin{aligned} \bar{p}(v) &= p(v - a) \\ &= k_1 g + k_2 g(v - a) + [(m_L + m_C)g + k_1 \ell](v - a)^2 + \\ &\quad + k_2 \ell(v - a)^3 + \ell m_C(v - a)^4 \end{aligned} \quad (2.2.23)$$

For a Hurwitz test this expression must be ordered by powers of  $v$ . In later references to this example we will replace  $v$  by  $s$  and  $\bar{p}$  by  $p$ , i.e. we will discuss Hurwitz stability of the uncertain polynomial

$$\begin{aligned} p(s) &:= a_0 + a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 \\ a_0 &= k_1 g - k_2 g a + [(m_L + m_C)g + k_1 \ell]a^2 - k_2 \ell a^3 + \ell m_C a^4 \\ a_1 &= k_2 g - 2[(m_L + m_C)g + k_1 \ell]a + 3k_2 \ell a^2 - 4\ell m_C a^3 \\ a_2 &= (m_L + m_C)g + k_1 \ell - 3k_2 \ell a + 6\ell m_C a^2 \\ a_3 &= k_2 \ell - 4\ell m_C a \\ a_4 &= \ell m_C \end{aligned} \quad (2.2.24)$$

$\square$

## 2.3 Choice of Sensors

In the examples of Chapter 1 the actuator inputs  $u$  are determined by the problem formulation. There are, however, several alternatives for the choice of the sensors and therefore of the output equation

$$\mathbf{y} = \mathbf{C}(\mathbf{q})\mathbf{x} \quad (2.3.1)$$

In this section, we compare some sensor choices considering observability, controller structures, and sensor costs. For plants with known parameters, one of the approaches to design a dynamic output feedback system is based on the separation principle. The states are reconstructed for example by an observer and the reconstructed state  $\hat{\mathbf{x}}$  replaces the true state  $\mathbf{x}$  in a state feedback controller.

For uncertain plants, state estimation by an observer or a Kalman filter is less useful because a plant model is required. If the plant parameters deviate from the model parameters, then in general separation does not hold and the advantage of separate design of state feedback and observer is lost. Thus, the robust stability of the closed loop must be analyzed anyway and the feedback structure may as well be assumed in the form of a dynamic compensator. We prefer to assume a low order compensator, because an increase in order results in additional closed-loop eigenvalues that must be robustly stabilized, too. A pragmatic approach is to begin with a very simple controller (e.g. a PI controller) and then to expand it if the results were not yet satisfactory. It is recommended to use the best possible design of the simpler controller structure as starting point for further improvement in a higher-dimensional space of free controller parameters.

*Example 2.9.* Crane

The crab position  $x_1$  must be measured, see Example 2.6. The feedback  $u = -k_1x_1$  does not suffice for stabilization as shown in (2.2.18). Stabilization is achieved with additional feedback of the crab velocity  $x_2$

$$u = k_1(w - x_1) - k_2x_2, \quad k_1 > 0, \quad k_2 > 0 \quad (2.3.2)$$

In view of (2.1.10) the rope angle  $x_3$  should be used for fast placement of the empty hook. Expanding (2.3.2) to take this into account gives

$$u = k_1(w - x_1) - k_2x_2 - k_3x_3 \quad (2.3.3)$$

Starting with  $k_1 > 0$ ,  $k_2 > 0$ , and  $k_3 = 0$ , the possible improvements with  $k_3 \neq 0$  can be analyzed.

Since  $x_2$  is observable from  $x_1$ , the sensor for the crab velocity can be replaced by low pass filtered differentiation of the crab position. In the Laplace domain a feasible controller structure is then

$$u(s) = (k_1 + k_2s/d(s))(w(s) - x_1(s)) - k_3x_3(s) \quad (2.3.4)$$

For compensator properness the denominator  $d(s)$  must be at least of order one, for example

$$d(s) = 1 + Ts \quad (2.3.5)$$

or

$$d(s) = 1 + 2Ds/\omega_0 + s^2/\omega_0^2 \quad (2.3.6)$$

This control system is illustrated in Fig. 2.3.

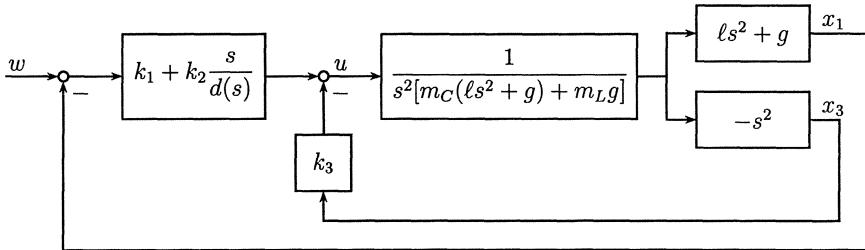


Fig. 2.3: Crane control system structure

The time constant  $T =: k_4$  in (2.3.5) is a fourth free controller parameter (similarly, in (2.3.6)  $D$  may be fixed and  $k_4 = 1/\omega_0$  is chosen as additional controller parameter). The design for the controller structure (2.3.4) is started with  $k_4 = 0$  and  $k_1, k_2, k_3$  from the design for the controller structure (2.3.3). In further iterations with  $k_4 \neq 0$ ,  $k_1, k_2, k_3$  may also be varied. In any case, there is a stable starting value in the four dimensional design space. The expansion of the controller by additional parameters makes the design procedure more transparent than a complete design immediately starting with (2.3.4). In the above controller structure, we have iterated from two to three and then back to two sensors.

The transfer functions of the plant (1.1.10) and (1.1.11) and the controller (2.3.4) yield the closed-loop characteristic polynomial

$$\ell m_C p(s) = (s^2 \ell + g)[d(s)(k_1 + m_C s^2) + k_2 s] + d(s)s^2(m_L g - k_3) \quad (2.3.7)$$

□

*Remark 2.3.* Note that (2.3.7) is written in a form such that each of the parameters  $\ell, m_C, m_L, k_1, k_2, k_3$  appears only once. This form has advantages for one of the approaches to robustness analysis as will be shown in Chapter 6. For a Hurwitz stability analysis, the polynomial must be first multiplied out and ordered by powers of  $s$  as in (2.2.18). This process, however, “smears” the parameters over several polynomial coefficients. □

#### Example 2.10. Automatic car steering

If you steer a car, then you play the role of a controller. Which sensors do you use for this task? You will use the visual measurement of the lateral displacement from your planned path and of the yaw rate (change of line of sight). By the seat you feel the

lateral acceleration. Additional information may be obtained from the reaction forces at the steering wheel. For an automatic steering system, at least a measurement of the lateral displacement  $y$  from the guideline is needed because  $y$  is not observable from any other state variable, see Fig. 1.9. Note that reference input  $w$  (guideline inertial position) and the controlled variable  $y_{abs}$  (inertial sensor position) are not available separately. Only their difference  $e = y_{abs} - w$  is measured and fed back. If only front wheel steering is used, then the control system structure of Fig. 2.4 is obtained.

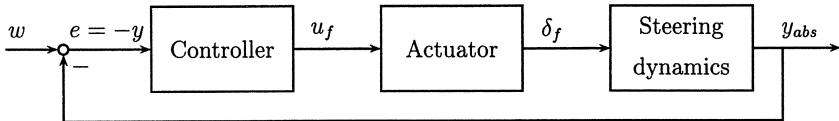


Fig. 2.4: Automatic steering of a car. The error  $e = w - y_{abs}$  between guideline  $w$  and sensor position  $y_{abs}$  is measured

The controller of Fig. 2.4 can be expanded if further state variables of steering dynamics are measured, e.g. by a gyro for the yaw rate or by an accelerometer for the lateral acceleration. The displacement from the guideline and the lateral acceleration may also be measured at several locations, e.g. at the front and rear ends of the vehicle. Also the actual steering angle  $\delta_f$  may be measured (e.g. by a potentiometer) and fed back. Further the car velocity  $v$  can be measured and used for scheduling the controller.

For the automatic steering example, several sensor concepts will be reviewed. In an early version of the automatic steering system for a bus the displacements  $y_f$  and  $y_r$  at the front and rear ends were measured. In this case it is convenient to transform the state space model (1.3.4) such that  $y_f$  and  $y_r$  and their derivatives become state variables and to augment the model by the low pass servo steering system

$$\dot{\delta}_f = -\frac{1}{T}\delta_f + \frac{1}{T}u \quad (2.3.8)$$

with transfer function  $1/(1 + Ts)$ . A feasible state vector is

$$\bar{x} = \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \\ \delta_f \end{bmatrix} \quad (2.3.9)$$

with measurement

$$\begin{aligned} y &= C\bar{x} \\ C &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.3.10)$$

The controller was designed in two steps [20]. First, a robust state feedback

$$u = -[k_1 \ k_2 \ k_3 \ k_4 \ k_5] \bar{x} \quad (2.3.11)$$

was designed. Then the unmeasurable term  $k_2\dot{y}_f + k_4\dot{y}_r$  was formed by approximate differentiation of  $k_2y_f + k_4y_r$ . The resulting controller structure is

$$u(s) = -k_1y_f(s) - k_3y_r(s) - k_5\delta_f(s) - \frac{s}{d(s)}[k_2y_f(s) + k_4y_r(s)] \quad (2.3.12)$$

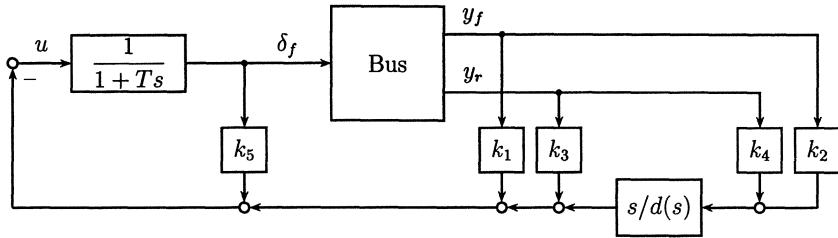


Fig. 2.5: Feedback structure for automatic steering with three sensors

Fig. 2.5 illustrates the control system structure. The denominator  $d(s)$  was chosen as in (2.3.6)

$$d(s) = 1 + 2Ds/\omega_0 + s^2/\omega_0^2 = 1 + 2k_6Ds + k_6^2s^2 \quad (2.3.13)$$

For low frequencies  $\omega < \omega_0$  the term  $s/d(s)$  is an approximation for the ideal differentiator  $s$ . In the second design step  $D = 1/\sqrt{2}$  was fixed and the reciprocal bandwidth  $k_6 = 1/\omega_0$  was enlarged starting from  $k_6 = 0$ . The approximate differentiation of a position for velocity reconstruction may be used also for other mechanical systems. Compared to the reconstruction of the velocity by an observer, the structure of (2.3.12) has the great advantage that no model of the plant is needed in the controller.

In a later version of the automatic bus steering controller the sensor effort was reduced and only  $y_f$  was fed back. What remains is the control system of Fig. 2.4. Also the actuator was changed to an integrating actuator without position feedback, i.e. the actuator transfer function is  $1/s$  and the transfer function from actuator input to the displacement from the guideline is

$$g(s) = \frac{n_f(s)}{(a_0 + a_1s + s^2)s^3} \quad (2.3.14)$$

with the coefficients given in (1.3.8).

For illustration of the design problem, first proportional feedback  $u_f = k_e$  is assumed. Fig. 2.6 shows the corresponding root locus for the case of maximal velocity  $v$  and maximal virtual mass  $\tilde{m}$ . The transfer function (2.3.14) has a triple pole at  $s = 0$ . Thus, there are three root locus branches at zero with breakaway angles  $\pm 60^\circ$  and  $180^\circ$  and the closed loop is unstable for small loop gains, see Fig. 2.6. The loop is also unstable for large loop gains, because the asymptotes of the root locus have angles

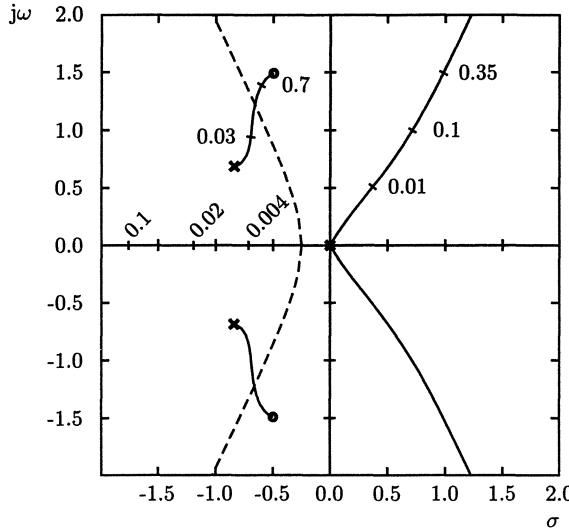


Fig. 2.6: Root locus of an automatic car steering system with proportional feedback of the front displacement from the guideline

$\pm 60^\circ$  and  $180^\circ$ . The uncertain velocity  $v$  in the numerator has the effect of an uncertain gain. Thus, the difficulty of a robust design for a large velocity range may be roughly described as pulling the root locus into the left half  $s$ -plane for a large enough interval of loop gains.

For accomplishing this goal it is crucial to place a pair of zeros in the vicinity of  $s = 0$  very accurately. On the other hand, the location of the controller poles further left in the  $s$ -plane has much less influence. A controller with relative degree one and a Butterworth pole configuration was assumed, i.e.

$$u_f(s) = \frac{k_1 + k_2 s + k_3 s^2}{(1 + s/\omega_0 + s^2/\omega_0^2)(1 + s/\omega_0)} e(s) \quad (2.3.15)$$

For frequencies  $\omega < \omega_0$  the gain  $k_1$  may be viewed as proportional gain,  $k_2$  as derivative gain, and  $k_3$  as second derivative gain. During the design primarily  $k_1$ ,  $k_2$ , and  $k_3$  were varied, i.e. a pair of zeros and the loop gain. The reciprocal bandwidth  $k_4 = 1/\omega_0$  was increased as long as no significant loss in the stability margin was noted. The resulting low pass behavior keeps high frequency noise from the sensor and guideline away from the actuator. The closed-loop characteristic polynomial follows from (1.3.9) and (2.3.15)

$$\begin{aligned} p(s, q, k) = & (k_1 + k_2 s + k_3 s^2) c_f v [e_0(q) + e_1(q)s + e_2(q)s^2] + \\ & +(1 + k_4 s + k_4^2 s^2)(1 + k_4 s) [d_0(q) + d_1(q)s + d_2(q)s^2] \end{aligned} \quad (2.3.16)$$

□

*Example 2.11.* Aircraft pitch stabilization

In the flight control problem of Section 1.4 with state vector  $\mathbf{x} = [n_z \ q \ \delta_e]^T$ , the normal acceleration  $n_z$  can be measured by an accelerometer and the pitch rate  $q$  can be measured by a gyro. The third state  $\delta_e$  is the deviation of the elevator from its trim position. The latter is not known with sufficient accuracy, therefore,  $\delta_e$  is not used for feedback. Thus, the measurement equation is

$$\mathbf{y} = \mathbf{C}\mathbf{x}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.3.17)$$

The simplest controller structure is output feedback

$$u = -[k_{nz} \ k_q]\mathbf{y} \quad (2.3.18)$$

From the viewpoint of observability each of the two sensors would suffice. But it is not clear at this point which sensor is better suited for robust control. During the design in Chapter 11 both alternatives will be considered and it will be shown why the gyro is the better choice. This example illustrates that the controller structure may not always be fixed before the controller design, it may also be a result of the design process.  $\square$

## 2.4 Further Aspects of Controller Structures

For robustness with respect to large plant parameter uncertainty, it is most important that the controller shifts the eigenvalues for all operating conditions  $q \in Q$  into a desirable pole region in the  $s$ -plane. In this section, we consider further requirements for the controlled systems as far as they have influence on the controller structure. Some well-known qualitative rules are discussed in their simplest form in order to point out some robustness aspects of controller structures.

### Degrees of freedom

Horowitz [85] classifies control system structures by the number of controller inputs. He calls this number the degrees of freedom (not to be confused with the degrees of freedom of a mechanical system!). In this sense, feedback of the tracking error in automatic steering is a control system structure with one degree of freedom, see Fig. 2.4.

A second degree of freedom is available, if reference input and controlled variable are separate input signals of the controller as in Fig. 2.1. An example is the flight control problem, where the pilot command  $z(t)$  is low pass filtered in order to reduce saturation of the rudder actuator motors. The prefilter  $f(s)$  modifies the reference response from  $z$  to  $y$ . Closed-loop specifications for control systems are frequently formulated in terms of step responses. The prefilter  $f(s)$  gives more design freedom for the satisfaction of the specifications. In eigenvalue-oriented robustness analysis we are only interested in the design of the compensator  $c(s)$  inside the loop. The sharing of tasks between  $c(s)$  and  $f(s)$  may be viewed such that  $c(s)$  takes care that the controlled output  $y(t)$  is a

sum of well-damped and quickly decaying solution terms, while amplitude and phase of these terms depend on  $f(s)$  and on the zeros of  $g(s, q)$ .

#### *Integral term*

In the control system of Fig. 2.1, consider first  $w$  as a step input signal with Laplace transform  $w(s) = 1/s$ . Also assume that the loop is stable for all  $q \in Q$ . The step responses of the signals in the loop have the stationary values  $e_{stat}$ ,  $u_{stat}$ , and  $y_{stat}$ . By the final value theorem of Laplace transformation

$$\begin{aligned} e_{stat} &= \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} se(s) \\ &= \lim_{s \rightarrow 0} s \frac{1}{1 + c(s)g(s, q)} w(s) \\ &= \lim_{s \rightarrow 0} \frac{1}{1 + c(s)g(s, q)} \end{aligned} \quad (2.4.1)$$

Assume that  $g(0, q) \neq 0$  for all  $q \in Q$ . The stationary error  $e_{stat}$  becomes zero, i.e.  $y_{stat} = w_{stat} = 1$ , if the transfer function  $c(s)g(s, q)$  contains an integration, i.e. a pole at  $s = 0$ . If the plant does not have a pole at  $s = 0$ , then the compensator  $c(s)$  should have this pole, i.e. an integral term. This structural assumption guarantees stationary accuracy for all  $q \in Q$ . An integral term in the controller may be useful even if the plant already has a pole at  $s = 0$ , and constant disturbances enter into the loop before the integration in the plant. An example is a constant side wind force acting on a car with automatic steering.

In the control system of Fig. 2.1 with a step input  $z(s) = 1/s$  the same considerations hold regarding the integrating term. The prefilter transfer function must satisfy  $f(0) = 1$  such that  $w_{stat} = z_{stat}$ .

#### *Relative degree and controller bandwidth*

A further consideration for the assumed controller structure is its relative degree (denominator degree minus numerator degree). A high or even infinite controller bandwidth, as it is obtained with difference degree zero, has the effect that high frequency noise entering into the loop together with  $y$  causes corresponding high frequency actuator activity. This effect is undesirable, especially in elastic mechanical systems (like the aircraft), because higher frequency structural vibrations may be excited. Therefore, we prefer controllers with difference degree one for systems with elasticity or sensor noise. Then the asymptotes of the root locus are located further to the right in  $s$ -plane than for difference degree zero. Thus, robust stabilization is more difficult, and this control system structure forces to use only moderately large loop gains. In view of actuator saturation the above control system structure compares favorably with a controller with difference degree zero and higher loop gain. Controller poles may be assumed with uniform distance  $\omega_0$  from the origin  $s = 0$ , e.g. see (2.3.6), (2.3.13), (2.3.15), such that the open loop frequency response  $|c(j\omega)g(j\omega, q)|$  rapidly decays for frequencies  $\omega > \omega_0$ . Thus, the influence of high frequency modelling uncertainty is kept small.

#### *Cascade structures*

Some plants have a cascade structure, i.e. their transfer function may be written as

$$g(s, \mathbf{q}) = g_B(s, \mathbf{q}_B)g_A(s, \mathbf{q}_A) \quad (2.4.2)$$

If the intermediate signal  $u(s)g(s, \mathbf{q}_A)$  is measured and used for feedback, then the cascade control system structure of Fig. 2.7 is feasible.

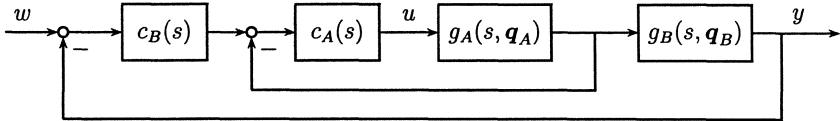


Fig. 2.7: Cascade control system structure

Cascaded systems can be analyzed and synthesized sequentially. First, the compensator  $c_A(s)$  is designed such that the inner loop becomes robust with respect to the uncertainty  $\mathbf{q}_A$ . Then the compensator  $c_B(s)$  is designed such that the outer loop becomes robust with respect to the uncertainty  $\mathbf{q}_B$ . If there are no uncertain parameters entering both into  $\mathbf{q}_A$  and  $\mathbf{q}_B$ , then also the robustness analysis can be performed in two sequential steps as shown in Chapter 9.

#### *Controller structures for MIMO systems*

Consider a state-space model of a multi-input, multi-output (MIMO) system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(q)\mathbf{x} + \mathbf{B}(q)\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(q)\mathbf{x} \end{aligned} \quad (2.4.3)$$

with output feedback

$$\mathbf{u} = -\mathbf{K}\mathbf{y} + \mathbf{w} \quad (2.4.4)$$

The characteristic polynomial is

$$p(s, q, \mathbf{K}) = \det[sI - \mathbf{A}(q) + \mathbf{B}(q)\mathbf{K}\mathbf{C}(q)] \quad (2.4.5)$$

The elements of the matrix  $\mathbf{K}$  no longer enter linearly as in the single-input case, but enter multilinearly into  $p(s, q, \mathbf{K})$ . The problem of robustness analysis and robust controller synthesis for this class of characteristic polynomials is more difficult.

It is desirable to break the MIMO problem into several smaller problems, for example single-input problems. Suppose that we find a  $\mathbf{K}$  such that  $\mathbf{A}(q) - \mathbf{B}(q)\mathbf{K}\mathbf{C}(q)$ ,  $\mathbf{B}(q)$ , and  $\mathbf{C}(q)$ , can be transformed by some similarity transformation  $\mathbf{T}(q)$  to the following form

$$\begin{aligned} \mathbf{T}(q)[\mathbf{A}(q) - \mathbf{B}(q)\mathbf{K}\mathbf{C}(q)]\mathbf{T}^{-1}(q) &= \begin{bmatrix} \mathbf{A}_{11}(q) & \mathbf{0} \\ \mathbf{A}_{21}(q) & \mathbf{A}_{22}(q) \end{bmatrix} \\ \mathbf{T}(q)\mathbf{B}(q) &= \begin{bmatrix} \mathbf{B}_{11}(q) & \mathbf{0} \\ \mathbf{B}_{21}(q) & \mathbf{B}_{22}(q) \end{bmatrix} \\ \mathbf{C}(q)\mathbf{T}^{-1}(q) &= \begin{bmatrix} \mathbf{C}_{11}(q) & \mathbf{0} \\ \mathbf{C}_{21}(q) & \mathbf{C}_{22}(q) \end{bmatrix} \end{aligned} \quad (2.4.6)$$

with a corresponding partition of the input, transformed state, and output vectors

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}, \quad \mathbf{T}(\mathbf{q})\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}$$

It was shown in [101, 78] that the above form allows the following conclusions.

1. The first subsystem

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{A}_{11}(\mathbf{q})\mathbf{x}_1 + \mathbf{B}_{11}(\mathbf{q})\mathbf{u}_1 \\ \mathbf{y}_1 &= \mathbf{C}_{11}(\mathbf{q})\mathbf{x}_1 \end{aligned} \quad (2.4.7)$$

is not controllable from  $\mathbf{u}_2$ .

2. The state  $\mathbf{x}_2$  of the second subsystem

$$\begin{aligned} \dot{\mathbf{x}}_2 &= \mathbf{A}_{22}(\mathbf{q})\mathbf{x}_2 + \mathbf{B}_{21}(\mathbf{q})\mathbf{u}_1 + \mathbf{B}_{22}(\mathbf{q})\mathbf{u}_2 + \mathbf{A}_{21}(\mathbf{q})\mathbf{x}_1 \\ \mathbf{y}_2 &= \mathbf{C}_{22}(\mathbf{q})\mathbf{x}_2 + \mathbf{C}_{21}(\mathbf{q})\mathbf{x}_1 \end{aligned} \quad (2.4.8)$$

is not observable from  $\mathbf{y}_1$ .

Fig. 2.8 illustrates the above system inside the dashed box and a decentralized controller structure.

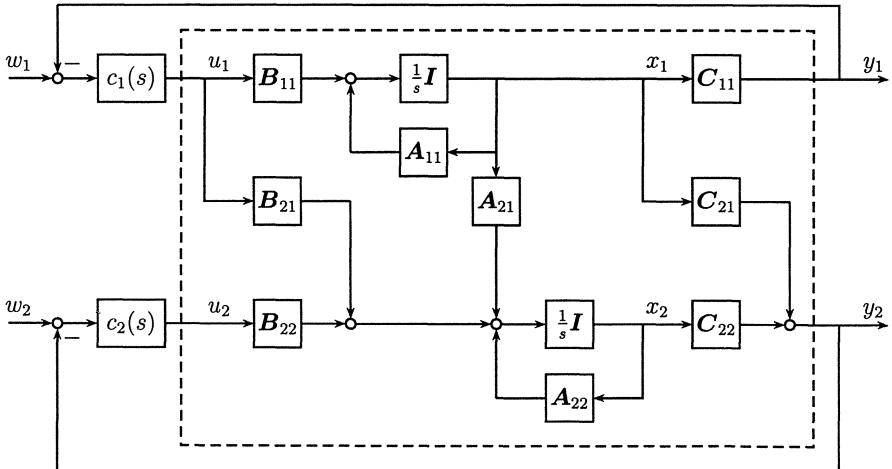


Fig. 2.8: Triangularized system and decentralized controllers  $c_1(s), c_2(s)$

The system has a triangular representation in terms of its transfer matrix

$$\begin{bmatrix} \mathbf{y}_1(s) \\ \mathbf{y}_2(s) \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{11}(s, \mathbf{q}) & \mathbf{0} \\ \mathbf{G}_{21}(s, \mathbf{q}) & \mathbf{G}_{22}(s, \mathbf{q}) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1(s) \\ \mathbf{u}_2(s) \end{bmatrix} \quad (2.4.9)$$

The two controllers  $c_1(s)$  and  $c_2(s)$  can now be designed independently. The eigenvalues of  $\mathbf{A}_{11}$  (the poles of  $\mathbf{G}_{11}(s, \mathbf{q})$ ) are shifted only by  $c_1(s)$  and the eigenvalues of  $\mathbf{A}_{22}$  (the poles of  $\mathbf{G}_{22}(s, \mathbf{q})$ ) are shifted only by  $c_2(s)$ .

In the context of robust control it is important that the zero elements in (2.4.6) and (2.4.9) are not only zero numerical values for some nominal operating point, they must be identical zero for all admissible  $\mathbf{q}$ .

There is no general procedure how to obtain a robustly triangularizing  $\mathbf{K}$ , and it is not yet clear when it exists. In the next section it will be shown that a triangularizing  $\mathbf{K}$  exists for cars with four-wheel steering and feedback of the yaw rate. In that context we will talk about *decoupling* although (2.4.9) is not diagonal. In common language decoupling is better understood than triangularization. In this sense the driver controlling the path tracking is decoupled from the automatic control system for yaw stabilization.

## 2.5 Robust Decoupling of Car Steering Dynamics

In car driving, four steering subtasks may be distinguished:

1. To plan the path within viewing distance.
2. To control lateral deviations of the vehicle from the planned path.
3. To damp the yaw motion that is excited by the steering commands.
4. To compensate the influence of external disturbances (crosswind, lateral slope of the lane,  $\mu$ -split braking, i.e. with different tire-road contact at the left and right wheels).

In this section the steering dynamics are decoupled (i.e. triangularized) by yaw rate feedback. The result will be a system structure as shown in Fig. 2.8 (inside the dashed box). Inputs are the front and rear steering angles  $u_1 = \delta_f$ ,  $u_2 = \delta_r$ . Outputs are  $y_1 = a_f$ , i.e. the lateral acceleration at the front axle and  $y_2 = r$ , i.e. the yaw rate.

Thereby two control problems are decoupled and solved independently. An automatic control system with compensator  $c_2(s)$  takes care of the yaw motion. The lateral control may be entirely left to the driver or it may be made more robust by the compensator  $c_1(s)$ .

The first subsystem in Fig. 2.8 is characterized by  $\mathbf{A}_{11}$ , we refer to it as the “lateral mode”. The second subsystem is characterized by  $\mathbf{A}_{22}$ , we refer to it as the “yaw mode”.

In Section 2.6 we will then independently design compensators  $c_1(s)$  and  $c_2(s)$  for velocity-invariance of the entire system.

The first problem is now to find a robustly triangularizing (= decoupling)  $\mathbf{K}$  that makes the yaw mode unobservable from the output  $a_f$ . A very simple solution results if two

additional assumptions A5 and A6 are made (Assumptions A1 to A4 are explained in Appendix A).

A5)

Let the longitudinal mass distribution of the vehicle be equivalent to two concentrated masses  $m_f$  and  $m_r$  at the front and rear axles, see Fig. 2.9. By the definition of the

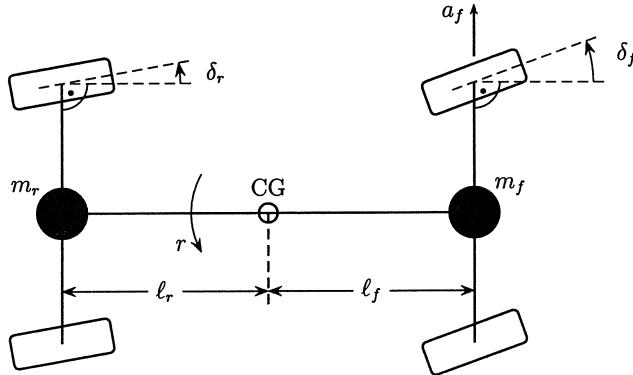


Fig. 2.9: 4WS car with moment of inertia equivalent to concentrated masses  $m_f$  and  $m_r$  at front and rear axles

center of gravity (CG) it follows that

$$m_f \ell_f = m_r \ell_r \quad (2.5.1)$$

A6)

Assume that the front steering angle  $\delta_f$  is adjusted by an integrating motor with transfer function  $1/s$ . This may be a hydraulic or electric motor without the position feedback that is built into power steering systems. An integrating actuator is for example used in the automatic steering system of the bus O 305, see (1.3.6). Thus, we have

$$\dot{\delta}_f = e_f \quad (2.5.2)$$

and  $e_f$  is considered as the plant input before decoupling.

Under assumptions A5 and A6 the plant model of (1.2.1) with the performance variable  $a_f$  of (A.2.10) as output becomes

$$\begin{aligned} \begin{bmatrix} \dot{\beta} \\ \dot{r} \\ \dot{\delta}_f \end{bmatrix} &= \begin{bmatrix} a_{11} & a_{12} & b_{11} \\ a_{21} & a_{22} & b_{21} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ r \\ \delta_f \end{bmatrix} + \begin{bmatrix} 0 & b_{12} \\ 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} e_f \\ \delta_r \end{bmatrix} \\ a_f &= [c_1 \quad c_2 \quad d_1] \begin{bmatrix} \beta \\ r \\ \delta_f \end{bmatrix} \end{aligned} \quad (2.5.3)$$

where

$$\begin{aligned}
 a_{11} &= -(c_r + c_f)/\tilde{m}v \\
 a_{12} &= -1 + (c_r \ell_r - c_f \ell_f)/\tilde{m}v^2 \\
 a_{21} &= (c_r \ell_r - c_f \ell_f)/\tilde{m} \ell_r \ell_f \\
 a_{22} &= -(c_r \ell_r^2 + c_f \ell_f^2)/\tilde{m} v \ell_r \ell_f \\
 b_{11} &= c_f/\tilde{m}v \\
 b_{12} &= c_r/\tilde{m}v \\
 b_{21} &= c_f/\tilde{m} \ell_r \\
 b_{22} &= -c_r/\tilde{m} \ell_f \\
 c_1 &= -\ell c_f/\tilde{m} \ell_r \\
 c_2 &= -\ell \ell_f c_f/\tilde{m} v \ell_r \\
 d_1 &= \ell c_f/\tilde{m} \ell_r
 \end{aligned}$$

Note, that  $c_1, c_2$ , and  $d$  are related by  $c_1 = -d, c_2 = -d \ell_f/v, d_1 = d$ , where  $d = \ell c_f/\tilde{m} \ell_r$ . The main result of this section is now formulated as a theorem, see [6, 7].

*Theorem 2.1. (Ackermann)*

The control law

$$\begin{bmatrix} e_f \\ \delta_r \end{bmatrix} = -\mathbf{K} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} u_f \\ \delta_r \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.5.4)$$

robustly decouples the system (2.5.3).

□

The robustly decoupling control law is illustrated by Fig. 2.10. The yaw rate  $r$  is

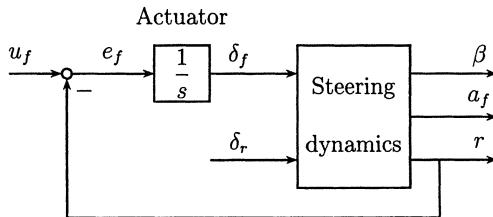


Fig. 2.10: Robustly decoupling control law

measured by a gyro and subtracted from the new input  $u_f$ . The error signal  $e_f = u_f - r$  is directly connected to the input of an integrating steering actuator  $\dot{\delta}_f = e_f$ . The question how  $u_f$  is produced from the steering wheel input will be discussed in Section 2.6.

*Proof.*

The effect of the control law (2.5.4) will be obvious in a different basis of the state space. We introduce the new state vector

$$\begin{bmatrix} a_f \\ r \\ \delta_f \end{bmatrix} = \mathbf{T}(\mathbf{q}) \begin{bmatrix} \beta \\ r \\ \delta_f \end{bmatrix}, \quad \mathbf{T}(\mathbf{q}) = \begin{bmatrix} c_1 & c_2 & d_1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5.5)$$

The transformed state equations are

$$\begin{bmatrix} \dot{a}_f \\ \dot{r} \\ \dot{\delta}_f \end{bmatrix} = \begin{bmatrix} d_{11} & d_1 & 0 \\ d_{21} & d_{22} & d_{23} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_f \\ r \\ \delta_f \end{bmatrix} + \begin{bmatrix} d_1 & 0 \\ 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} e_f \\ \delta_r \end{bmatrix} \quad (2.5.6)$$

with

$$\begin{aligned} d_1 &= \ell c_f / \tilde{m} \ell_r \\ d_{11} &= -\ell c_f / \tilde{m} v \ell_r \\ d_{21} &= (c_f \ell_f - c_r \ell_r) / c_f \ell_f \ell \\ d_{22} &= -c_r \ell / \tilde{m} v \ell_f \\ d_{23} &= c_r / \tilde{m} \ell_f \\ b_{22} &= -c_r / \tilde{m} \ell_f \end{aligned}$$

and the moment of inertia is

$$\begin{aligned} J &= m_f \ell_f^2 + m_r \ell_r^2 \\ &= m_r \ell_r \ell_f + m_f \ell_f \ell_r \\ &= m \ell_f \ell_r \end{aligned} \quad (2.5.7)$$

where

$$m = m_f + m_r \quad (2.5.8)$$

is the total vehicle mass.

A comparison of (2.5.8) with (1.2.10) shows the relation with the radius of inertia  $i$

$$i^2 = \ell_f \ell_r \quad (2.5.9)$$

The virtual moment of inertia  $\tilde{J} = J/\mu$  and the virtual mass  $\tilde{m} = m/\mu$  are related by  $\tilde{J} = \tilde{m} \ell_f \ell_r$ .

Crucial for robust decoupling is that

1. the term  $d_1$  occurs twice identically in the  $\mathbf{A}$  and  $\mathbf{B}$  matrices and
2. the term  $d_{13}$  vanishes.

Substituting the robustly decoupling feedback law, i.e.  $e_f = u_f - r$  in (2.5.6), yields

$$\begin{aligned} \begin{bmatrix} \dot{a}_f \\ - \\ \dot{r} \\ \dot{\delta}_f \end{bmatrix} &= \begin{bmatrix} d_{11} & | & 0^* & 0^* \\ - & - & - & - \\ d_{21} & | & d_{22} & d_{23} \\ 0 & | & -1 & 0 \end{bmatrix} \begin{bmatrix} a_f \\ - \\ r \\ \delta_f \end{bmatrix} + \begin{bmatrix} d_1 & 0^* \\ - & - \\ 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_f \\ \delta_r \end{bmatrix} \quad (2.5.10) \\ a_f &= \begin{bmatrix} 1 & | & 0^* & 0^* \end{bmatrix} \begin{bmatrix} a_f \\ r \\ \delta_f \end{bmatrix} \end{aligned}$$

The zeros marked with asterisks indicate the canonical form introduced by Kalman [101] and Gilbert [78] for the separation of observable and unobservable (or controllable and uncontrollable) subsystems. We read off (2.5.10) that

- the states  $r$  and  $\delta_f$  are unobservable from  $a_f$  and
- $a_f$  is not controllable from  $\delta_r$ .

This completes the proof of Theorem 2.1.

□

In (2.5.10) the steering dynamics have been split into two subsystems

- a) the lateral motion of the front axle described by

$$\dot{a}_f = d_{11}a_f + d_1u_f \quad (2.5.11)$$

- b) the yaw motion described by

$$\begin{bmatrix} \dot{r} \\ \dot{\delta}_f \end{bmatrix} = \begin{bmatrix} d_{22} & d_{23} \\ -1 & 0 \end{bmatrix} \begin{bmatrix} r \\ \delta_f \end{bmatrix} + \begin{bmatrix} d_{21} \\ 0 \end{bmatrix} a_f + \begin{bmatrix} 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_f \\ \delta_r \end{bmatrix} \quad (2.5.12)$$

The yaw mode as modified by the robustly decoupling control law (2.5.4) has the characteristic polynomial

$$\begin{aligned} p_d(s) &= s(s - d_{22}) + d_{23} = \omega_d^2 + 2D_d\omega_d s + s^2 \quad (2.5.13) \\ \omega_d^2 &= \frac{c_r}{m\ell_f} \\ D_d &= \frac{\ell}{2v} \sqrt{\frac{c_r}{m\ell_f}} \end{aligned}$$

It is stable, but weakly damped at high velocities  $v$ . Its natural frequency is velocity invariant. Damping can be increased for example by a long wheelbase  $\ell$  or by feedback control of rear wheel steering as will be shown in Section 2.6.

The driver has to control only subsystem a). He keeps the car, considered as a mass point at the front axle, on top of the planned path by generation of a lateral acceleration via the transfer function

$$\begin{aligned} a_f(s) &= g_f(s, \mathbf{q}) u_f(s) \\ g_f(s, \mathbf{q}) &= \frac{d_1}{s - d_{11}} = \frac{v}{1 + Ts}, \quad T = \frac{mv\ell_r}{\ell c_f} \end{aligned} \quad (2.5.14)$$

This new transfer function for the steering task of the driver has one uncertain pole on the negative real axis. The gain is proportional to the velocity  $v$ . If this effect is not desired, then a constant gain can be achieved by a prefilter with scheduled gain  $1/v$  between the steering wheel command and the reference input  $u_f$ . For comparison, we give the transfer function from the steering wheel command  $\delta_S$  to  $a_f$  for the conventional car without decoupling yaw rate feedback, it is

$$\begin{aligned} \left( \frac{a_f(s)}{\delta_S(s)} \right)_{\text{convent.}} &= \frac{K_f(1 + T_1 s + T_2 s^2)}{1 + \tau_1 s + \tau_2 s^2} \\ K_f &= \frac{\ell c_f c_r v^2}{c_f c_r \ell^2 + mv^2(c_r \ell_r - c_f \ell_f)} \\ T_1 &= \frac{\ell}{v}, \quad T_2 = \frac{\ell_f m}{c_r} \\ \tau_1 &= \frac{\ell m v (c_f \ell_f + c_r \ell_r)}{c_f c_r \ell^2 - mv^2(c_f \ell_f - c_r \ell_r)} \\ \tau_2 &= \frac{\ell_f \ell_r m^2 v^2}{c_f c_r \ell^2 - mv^2(c_f \ell_f - c_r \ell_r)} \end{aligned} \quad (2.5.15)$$

It has two uncertain poles and two uncertain zeros. Both approach the imaginary axis for large velocity. It requires the skill of an experienced driver to control  $a_f$  via such a complicated and parameter dependent transfer function. It should be much easier for the driver to control the system (2.5.15).

The result of this section is summarized by Fig. 2.13. The yaw dynamics are described by (2.5.12). In terms of block diagrams it was shown that the systems of Fig. 2.10 and Fig. 2.11 are the same.

*Remark 2.4.* It was shown in [11,12] that the robust decoupling property of the control law (2.5.4) also holds for the nonlinear system with arbitrary nonlinear tire characteristics.  $\square$

## 2.6 Gain Scheduling Controllers

It is common practice to use gain scheduling compensators in the closed loop in order to achieve good performance in spite of large parameter variations.

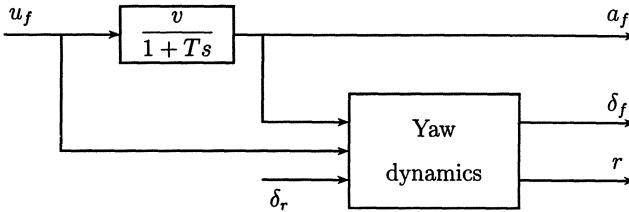


Fig. 2.11: Illustration of the decoupling control law. The control system of Fig. 2.3 is equivalent to the above system.

For example, in flight control systems the dynamic pressure can be measured. In the operating domain of Fig. 1.11 the dynamic pressure is high in the lower right corner (high speed in low altitude, i.e. dense air), it is low in the left upper part (slow flight in thin air). Therefore, dynamic pressure measurements allow some distinction between the operating points and are used for scheduling of controllers [36].

In automatic car steering the velocity may be measured by counting inductively the teeth of the last toothed wheel in the gear box [162]. Also, in four-wheel steering cars a measurement of the velocity and its use for scheduling prefilters is state of the art, e.g. [57, 91].

In the theoretical control literature, gain scheduling is largely ignored. Standard numerical synthesis procedures (e.g. pole placement or optimization of a quadratic performance criterion) assume a nominal plant model and the design procedure leads to a unique solution. The resulting controller allows no flexibility for large parameter variations. Therefore, often designs for different nominal parameter values are made and the resulting family of controllers is scheduled by the actual parameter measurement. The underlying misconception is that controller design should strive to satisfy the same synthesis criteria for all operating conditions. For systems with uncertain parameters such criteria overspecify the synthesis. Gain scheduling is then only an unsatisfactory artifice to obtain a feasible controller structure.

On the other hand there are control problems where the structure of a gain scheduled controller results from analytical studies of the plant model structure. In such cases it may also be possible to introduce a tuning parameter in the controller structure that aims exclusively at one particular system property. There is again no general theory behind this approach, it can only be judged by its success for particular plant model structures. To be more specific, the above idea will be worked out for the car steering example. Consider the robustly decoupled system of Fig. 2.11. First, we will design a velocity scheduled controller for feedback of  $a_f$  to  $u_f$ . It yields a velocity-invariant steering transfer function and the tuning parameter determines the location of the real pole. Second, we will design a velocity-scheduled controller for feedback of  $r$  to  $\delta_r$ . It yields velocity-invariant yaw eigenvalues and the tuning parameter determines the yaw damping. Here we exploit the fact that the two designs are independent of each other.

*The velocity-invariant steering transfer function*

*Theorem 2.2. (Ackermann)*

The control law

$$u_f = k_S(a_{fref} - a_f) + \frac{1}{v}a_f \quad (2.6.1)$$

yields a velocity-invariant steering transfer function relating  $a_{fref}(s)$ , and  $a_f$  by

$$a_f(s) = \frac{1}{1 + (a/k_S)s}a_{fref}(s), \quad a = m\ell_r/c_f\ell \quad (2.6.2)$$

The time constant  $T_s = a/k_S$  can be adjusted by the tuning parameter  $k_S$ , for example so small that the driver does not notice any delay.

□

*Proof.*

Substituting (2.6.1) into (2.5.11) yields

$$\begin{aligned} \dot{a}_f &= -\frac{\ell c_f}{mv\ell_r}a_f + \frac{\ell c_f}{m\ell_r}[k_S(a_{fref} - a_f) + \frac{1}{v}a_f] \\ &= -\frac{\ell c_f k_S}{m\ell_r}a_f + \frac{\ell c_f k_S}{m\ell_r}a_{fref} \end{aligned}$$

The corresponding transfer function is

$$a_f(s) = \frac{1}{1 + T_S s}, \quad T_S = \frac{m\ell_r}{\ell c_f k_S}$$

□

*The velocity-invariant yaw eigenvalues*

In [10] the following result was shown

*Theorem 2.3. (Ackermann)*

The control law

$$\delta_r = u_r - \left( \frac{\ell}{v} - k_D \right) r \quad (2.6.3)$$

yields velocity invariant yaw eigenvalues. It places the yaw eigenvalues at the roots of

$$p_r(s) = \omega_r^2 + 2D_r\omega_r s + s^2$$

with natural frequency

$$\omega_r = \frac{c_r}{m\ell_f} \quad (2.6.4)$$

and damping

$$D_r = \frac{k_D}{2} \sqrt{\frac{c_r}{m\ell_f}} \quad (2.6.5)$$

The damping  $D_r$  can be adjusted by the tuning parameter  $k_D$ .

□

Note that the sign of the feedback changes at a velocity  $v = \ell/k_D$ .

*Proof.*

The effect of the controller (2.6.3) is shown by substitution into (2.5.12).

$$\begin{bmatrix} \dot{r} \\ \dot{\delta}_f \end{bmatrix} = \begin{bmatrix} d_{22} - b_{22}(\ell/v - k_D) & d_{23} \\ -1 & 0 \end{bmatrix} \begin{bmatrix} r \\ \delta_f \end{bmatrix} + \begin{bmatrix} d_{21} \\ 0 \end{bmatrix} a_f + \begin{bmatrix} 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_f \\ u_r \end{bmatrix}$$

and with  $d_{22}$ ,  $b_{22}$ , and  $d_{23}$ , from (2.5.6)

$$\begin{aligned} \begin{bmatrix} \dot{r} \\ \dot{\delta}_f \end{bmatrix} &= \begin{bmatrix} -k_D c_r / m\ell_f & c_r / m\ell_f \\ -1 & 0 \end{bmatrix} \begin{bmatrix} r \\ \delta_f \end{bmatrix} + \begin{bmatrix} d_{21} \\ 0 \end{bmatrix} a_f + \begin{bmatrix} 0 & b_{22} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_f \\ u_r \end{bmatrix} \\ d_{21} &= (c_f \ell_f - c_r \ell_r) / c_f \ell_f \ell \\ b_{22} &= -c_r / m\ell_f \end{aligned} \quad (2.6.6)$$

The characteristic polynomial is

$$p_r(s) = \omega_r^2 + 2D_r \omega_r s + s^2 \quad (2.6.7)$$

$$\omega_r^2 = \frac{c_r}{m\ell_f}$$

$$D_r = \frac{k_D}{2} \sqrt{\frac{c_r}{m\ell_f}}$$

□

Note that the control law (2.6.3) does not change the natural frequency. A comparison of (2.5.14) and (2.6.8) yields  $\omega_r = \omega_d$ .

For desired worst case damping  $D_r$  the free controller parameter  $k_D$  must be set to

$$k_D = 2D_r \sqrt{\frac{m^+ \ell_f}{c_r^-}} \quad (2.6.8)$$

for the maximal mass  $m^+$  and the minimal rear cornering stiffness  $c_r^-$ . The results of Sections 2.5 and 2.6 are summarized in Fig. 2.12.

For the implementation of the control system the following sensors are required:

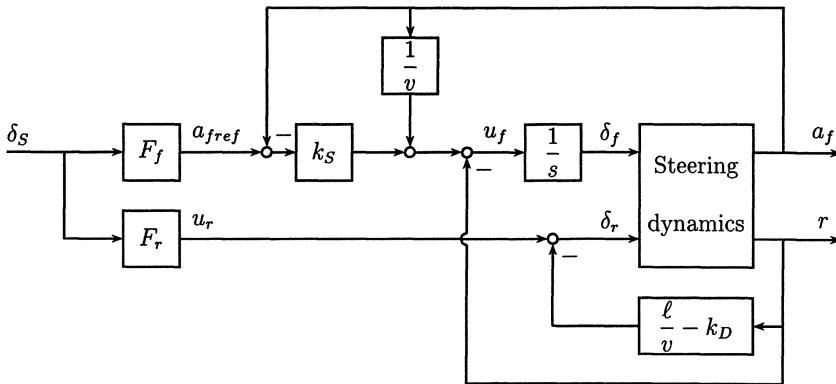


Fig. 2.12: Velocity-invariant car steering system. The tuning parameter  $k_S$  adjusts the steering time constant, the tuning parameter  $k_D$  adjusts the yaw damping

- i) an accelerometer at the front axle for  $a_f$ ,
  - ii) a gyro for  $r$ ,
  - iii) a sensor for the velocity  $v$ .
- i) and ii) are used for feedback, iii) is used for gain scheduling. The only vehicle parameter that must be known is the wheelbase  $\ell$ , otherwise the control system is generic for all cars.

#### *The prefilters*

In Fig. 2.12 two prefilters  $F_f$  and  $F_r$  are assumed that produce the reference inputs  $a_{fref}$  and  $u_r$  from the steering wheel command  $\delta_S$ .

A simple choice would be  $F_f = 1$ ,  $F_r = 0$ , i.e. rear-wheel steering is only used for yaw stabilization and the driver commands the lateral acceleration  $a_f$  directly. This solution seems ideal for a lane change with small yaw rate. Another situation is entering a curve of radius  $R$ . Then the required stationary yaw rate

$$r_{stat} = v/R \quad (2.6.9)$$

should be reached quickly. Another design goal mentioned in the automotive literature [57] is zero sideslip angle, i.e.  $\beta \equiv 0$ . Then the longitudinal center line stays tangential to the driving path. It is easy to compute the required ratio  $F_r/F_f$  of the two prefilters for this purpose, see Exercise 2.8.

## 2.7 Problem Classes of Parametric Polynomials

In the previous sections of this chapter various controller structures have been assumed and the closed-loop characteristic polynomials were obtained in the parametric form as

$$p(s, \mathbf{q}, \mathbf{k}) = \sum_{i=0}^n a_i(\mathbf{q}, \mathbf{k}) s^i \quad (2.7.1)$$

With unspecified  $\mathbf{k}$  and  $\mathbf{q}$  this is called an “uncertain polynomial”. If a controller  $\mathbf{k} = \mathbf{k}^*$  and an operating domain  $Q$  is given, then we say that  $p$  generates a polynomial family. For the stability analysis of a polynomial family the type of coefficient function  $a_i(\mathbf{q}, \mathbf{k})$  plays an important role. We will construct several examples from (2.2.18), i.e. the crane with state feedback and characteristic polynomial coefficients

$$\begin{aligned} a_0 &= k_1 g \\ a_1 &= k_2 g \\ a_2 &= (m_L + m_C)g + k_1 \ell - k_3 \\ a_3 &= k_2 \ell - k_4 \\ a_4 &= \ell m_C \end{aligned} \quad (2.7.2)$$

Typical classes of parametric polynomials are

### 1. Interval coefficients

$$a_i \in [a_i^- ; a_i^+] \quad (2.7.3)$$

*Example 2.12.* Consider the crane with fixed state feedback and uncertain load mass  $m_L$ . Then  $a_0, a_1, a_3$ , and  $a_4$ , are fixed and  $a_2 \in [(m_L^- + m_C)g + k_1 \ell - k_3 ; (m_L^+ + m_C)g + k_1 \ell - k_3]$   $\square$

### 2. Affine coefficients

$$a_i(\mathbf{q}) = b_i + \mathbf{c}_i^T \mathbf{q} \quad (2.7.4)$$

An affine function has a constant term ( $b_i$ ) and a term that is linear in the uncertain parameter vector  $\mathbf{q}$ .

*Example 2.13.* Consider the crane with fixed state feedback and uncertain rope length  $\ell$ . Now,  $a_0$  and  $a_1$  are fixed and  $a_2, a_3$ , and  $a_4$  depend affinely on  $\ell$ .  $\square$

*Example 2.14.* Consider the crane with undetermined state feedback and uncertain load and crab masses  $m_L$  and  $m_C$ . The uncertain parameters  $k_1, k_2, k_3, k_4, m_L$ , and  $m_C$  enter linearly into the coefficients.  $\square$

### 3. Multilinear coefficients

*Example 2.15.* Consider the crane with undetermined state feedback and uncertain load and crab mass and rope length. Now  $a_2, a_3$ , and  $a_4$  contain the bilinear terms  $k_1 \ell, k_2 \ell$ , and  $\ell m_C$ . The parameters  $k_3, k_4$  and  $m_L$  enter linearly.  $\square$

#### 4. Polynomial coefficients

*Example 2.16.* Consider the automatic car steering problem (2.3.16) and (1.3.6) with uncertain mass and velocity and a fixed compensator transfer function. The coefficient functions contain the terms  $v, mv, v^2, mv^2$ , and  $m^2v^2$ .  $\square$

For the later robustness analysis it is important that the uncertain polynomial is classified in the simplest category. Sometimes simple polynomials look complicated but can be factorized into

$$p(s, \mathbf{q}) = f(\mathbf{q})\bar{p}(s, \mathbf{q}) , \quad f(\mathbf{q}) \neq 0 \text{ for all } \mathbf{q} \in Q \quad (2.7.5)$$

Then  $p(s, \mathbf{q})$  and  $\bar{p}(s, \mathbf{q})$  have the same roots but  $\bar{p}(s, \mathbf{q})$  may be simpler.

*Example 2.17.*  $p(s, \mathbf{q}) = q_1 + q_1^2 s + q_1 q_2 s^2 + q_1 s^3$  can be converted into an interval polynomial because

$$\bar{p}(s, \mathbf{q}) = \frac{p(s, \mathbf{q})}{q_1} = (1 + q_1 s + q_2 s^2 + s^3)$$

$\square$

*Example 2.18.*

$$p(s, \mathbf{q}) = \left(1 + \frac{q_1}{q_2}\right) + \frac{1+q_1}{q_2}s + \left(5 + \frac{2+q_1}{q_2}\right)s^2 + s^3$$

can be converted into an affine polynomial

$$\bar{p}(s, \mathbf{q}) = q_2 p(s, \mathbf{q}) = (q_1 + q_2) + (1 + q_1)s + (2 + q_1 + 5q_2)s^2 + q_2 s^3$$

$\square$

See also Exercise 2.4.

## 2.8 Exercises

- 2.1. Check robust controllability and stabilizability of the car with rear wheel steering.
- 2.2. Are car steering dynamics observable from the yaw rate?
- 2.3. Consider the transfer function (1.3.7) of the bus O 305 with data of Exercise 1.4.
  - a) For the most critical case  $\mu = 0.5$ ,  $m = 16000 \text{ [kg]}$ ,  $v = 20 \text{ [m} \cdot \text{s}^{-1}]$ , plot the root locus for proportional feedback of the displacement  $y$  to the steering actuator input  $u$  and compare with Fig. 2.6.
  - b) Repeat a) with a lead-lag compensator  $c(s) = k(s + 0.5)^2/(s + 6.16)^2$ .

- 2.4. Consider a hydraulic actuator for a robot arm. Uncertain parameters are actuator time constant  $q_2 \in [0.05 ; 0.2]$  and gripper mass, resulting in an uncertain moment of inertia  $q_1 \in [1 ; 10]$ . A compensator transfer function is given in Fig. 2.13. How do the uncertain parameters enter into the characteristic polynomial? Is it possible to reduce the problem to the affine case?

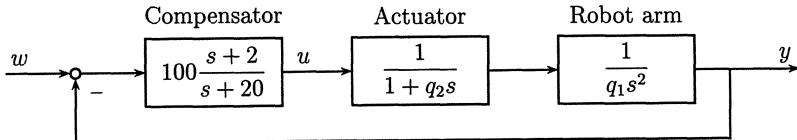


Fig. 2.13: Hydraulic robot arm control system

- 2.5. Show that the decoupling result of Section 2.5 also holds for nonlinear tire characteristics  $f_\alpha$ , see [12].
- 2.6. Check the four flight conditions of the aircraft for
- controllability,
  - observability from the pitch rate  $q$ ,
  - observability from the normal acceleration  $n_z$ .
- 2.7. For the car steering system of Fig. 2.12 calculate the transfer functions from the steering wheel input  $\delta_S$  to  $a_f$  (lateral acceleration at the front axle),  $r$  (yaw rate) and  $\beta$  (sideslip angle), see (2.5.3). Discuss the choices of prefilters  $F_f$  and  $F_r$  in view of the paragraph on prefilters in Section 2.6.

## 3 Analysis and Design

By assuming controller structures in Chapter 2 we have generated several examples of closed-loop characteristic polynomials  $p(s, \mathbf{q}, \mathbf{k})$ , where the vector  $\mathbf{k}$  contains the free design parameters in the fixed controller structure and  $\mathbf{q}$  contains the uncertain plant parameters in a given operating domain  $Q$ , i.e.  $\mathbf{q} \in Q$ .

The basic question of robustness analysis is: Given  $\mathbf{k} = \mathbf{k}^*$  and  $Q$ , is the polynomial family

$$P(s, Q, \mathbf{k}^*) = \{p(s, \mathbf{q}, \mathbf{k}^*) \mid \mathbf{q} \in Q\}$$

stable?

The basic question of robustness synthesis is: Given  $Q$ , find a  $\mathbf{k} = \mathbf{k}^*$ , such that  $P(s, Q, \mathbf{k}^*)$  is stable. In this chapter, the design specifications are extended to include more than just stability. In particular, we discuss performance measures in terms of eigenvalues locations. A system (or its characteristic polynomial) is called “Gamma”-stable, if all eigenvalues are located in a specified region  $\Gamma$  in the complex plane.

In the design of robust control systems, tradeoffs with other requirements must be made. In this situation we are not only interested in a particular solution  $\mathbf{k} = \mathbf{k}^*$  but in the set  $K_H$  such that  $P(s, Q, \mathbf{k}, \mathbf{q})$  is stable if and only if  $\mathbf{k} \in K_H$ .  $K_H$  describes the set of all (Hurwitz) stabilizing controllers of the assumed structure.

Various aspects of analysis and design are also discussed.

### 3.1 Eigenvalue Specifications

The performance of control systems is primarily evaluated by typical responses to reference and disturbance inputs. Examples are:

- Transport of a load by a crane over a distance of 1 meter, where the initial and final values of the rope angle and its rate and also of the crab velocity are zero.
- Transition of a vehicle from a straight track into a curve.

- Rapid increase of the crosswind acting on a car.
- Transition of an aircraft to a higher altitude.

Usually, several state variables and the actuator signals are of interest in such simulations. For systems with uncertain parameters simulations can be performed only for a grid of parameter values and require high computational effort in analysis and design iterations.

It is more practical to improve the time responses indirectly by eigenvalue specifications. Undesirable properties of a response and appropriate countermeasures are:

1. There are oscillations that do not decay quickly enough and cause excessive overshoot. The frequency  $\omega = 2\pi/T$  is determined from the period  $T$  of the undesired oscillation. In a distance  $\omega$  from the origin of the  $s$ -plane there is a complex pair of eigenvalues with insufficient damping. Improvement of this damping has priority in the next design step.
2. The response is “sluggish”, it creeps towards the stationary value. A negative real eigenvalue is too close to the origin and must be moved to the left in the next design step.
3. There is undesired high frequency content in the actuator signal. Countermeasures are reduction of the controller bandwidth, increase of the difference degree of the controller, shifting far left eigenvalues closer to the origin on a circle with radius  $\omega_b$ , such that the absolute value of the frequency response drops rapidly for frequencies  $\omega > \omega_b$ . In sampled-data systems usually the anti-aliasing filter is a cure.

For the single-input single-output plant some further aspects of pole shifting must be observed:

4. If the closed-loop eigenvalues are shifted too close to open loop zeros, then large loop gains may result as can be seen from the root locus. High loop gains are undesirable for our examples of mechanical systems in view of the limited actuator forces.
5. If the open loop has a relative degree of two or more (this is a common case), then the centrum of all eigenvalues of plant and compensator cannot be shifted. (In this case the coefficient  $a_{m-1}$  in a polynomial  $\prod_{i=1}^m (s - s_i) = a_0 + \dots + a_{m-1}s^{m-1} + s^m$  is determined by the plant and compensator poles only and not by any zeros. The coefficient defines the centrum of the  $s_i$  by  $a_{m-1} = -\sum_{i=1}^m s_i$ .) If some eigenvalues must be shifted to the left, then necessarily other eigenvalues migrate to the right. In this case the compensator poles must be chosen sufficiently far left, such that they still satisfy the eigenvalue specifications after a right shift.

6. Pole-zero cancellations outside the desired closed-loop eigenvalue region  $\Gamma$  should be avoided. From the frequency domain point of view they may still look acceptable, but there is always a worst disturbance or initial condition that excites the subsystem with the undesired response. The control system should be “internally  $\Gamma$ -stable”, i.e. the transfer functions from each possible input to each signal inside the feedback loop should have all their poles in the region  $\Gamma$ .

In the following paragraphs we recapitulate some simple relationships between time responses and eigenvalue locations for second and third order systems. Such relationships are useful if a “dominant behavior” similar to such simple systems is desired.

If all eigenvalues are located to the left of a parallel to the imaginary axis of the  $s$ -plane at  $\sigma = -a$ , then all solution terms decay at least like  $e^{-at}$ . Fig. 3.1 shows two examples:

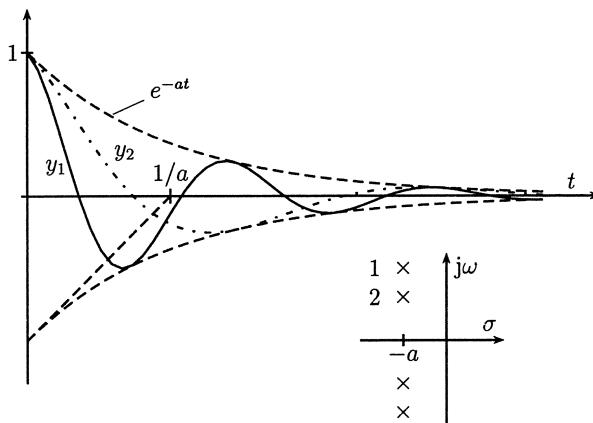


Fig. 3.1: Two solution terms with the same negative real part  $\sigma = -a$  of the eigenvalues

$y_1$  and  $y_2$  have the same negative real part  $\sigma = -a$  of the eigenvalues. Terms of the type  $y_1$  are however undesirable because more overshoot and larger oscillations occur inside the envelope  $\pm e^{-at}$  than for  $y_2$  with a lower frequency  $\omega_2$ . Thus, the high frequency eigenvalues should be located further to the left in the  $s$ -plane. Therefore, a minimum value of the damping  $D$  is required. A complex conjugate pair of eigenvalues  $\sigma_i \pm j\omega_i$  may be written as a second order factor of the closed-loop characteristic polynomial

$$p_i(s) = (s - \sigma_i - j\omega_i)(s - \sigma_i + j\omega_i) = s^2 - 2\sigma_i s + \sigma_i^2 + \omega_i^2 = s^2 + 2D\omega_0 s + \omega_0^2$$

The distance of the eigenvalues from the origin is the natural frequency  $\omega_0 = \sqrt{\sigma_i^2 + \omega_i^2}$ , and  $D = -\sigma_i/\omega_0$  is the damping. The inverse relation for the real part  $\sigma_i$  and the imaginary part  $\omega_i$  of the eigenvalue is  $\sigma_i = -D\omega_0$ ,  $\omega_i = \omega_0\sqrt{1 - D^2}$ . Fig. 3.2 illustrates these relations in the  $s$ -plane.

A damping value  $D$  corresponds to an angle  $\alpha$  with respect to the imaginary axis, where

$$D = \sin \alpha \quad (3.1.1)$$

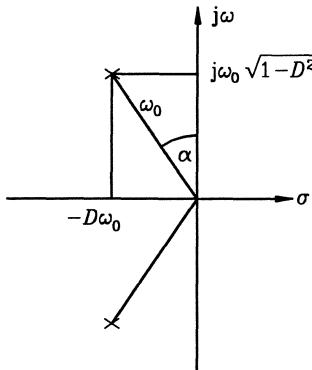


Fig. 3.2: Natural frequency  $\omega_0$  and damping  $D$  of a complex conjugate pair of poles

The corresponding solution term in the time domain is

$$y_i(t) = e^{-D\omega_0 t} \cos(\sqrt{1-D^2}\omega_0 t + \varphi) \text{ for } |D| < 1 \quad (3.1.2)$$

Natural frequency  $\omega_0$  and time  $t$  appear only as product  $\omega_0 t$ , i.e.  $\omega_0$  may be considered as a scaling factor for the time. As an example consider the step responses of the system

$$g(s) = \frac{1}{1 + 2Ds/\omega_0 + s^2/\omega_0^2} \quad (3.1.3)$$

for different damping values. For  $\omega_0 = 1$ ,  $u(s) = 1/s$ , and  $y(s) = g(s)u(s)$ , we have

$$y(s) = \frac{1}{(s^2 + 2Ds + 1)s} = \frac{1}{s} - \frac{(s + D) + D}{(s + D)^2 + (1 - D^2)}$$

The inverse Laplace transform is

$$y(t) = 1 - e^{-Dt} \left[ \cos(\sqrt{1-D^2}t) + \frac{D}{\sqrt{1-D^2}} \sin(\sqrt{1-D^2}t) \right] \quad (3.1.4)$$

The time responses for  $D = 0.5$ ,  $D = 1/\sqrt{2}$ , and  $D = 0.9$ , are shown in Fig. 3.3.

The step response for  $D = 1/\sqrt{2} \approx 0.7$  is considered as particularly favorable. Its maximum overshoot of 4.3% occurs at  $\omega_0 t = 4.4$ . The value  $D = 1/\sqrt{2}$  is characterized by the fact that the magnitude of the frequency response  $|g(j\omega)|$  has a maximum only for smaller damping; the resonance frequency is  $\omega_0 = \sqrt{1-2D^2}$ . For larger damping no resonance occurs. In Fig. 3.2 we have  $\alpha = 45^\circ$  for the damping value  $D = 1/\sqrt{2}$ .

Next we assume a zero at  $s = -b\omega_0$ .

$$g(s) = \frac{(1 + s/(b\omega_0))}{1 + 2Ds/\omega_0 + s^2/\omega_0^2} \quad (3.1.5)$$

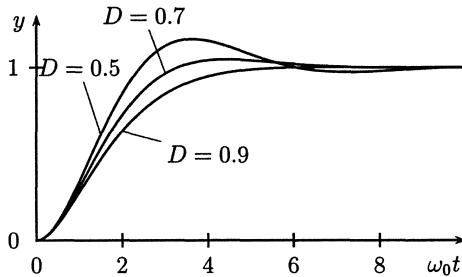


Fig. 3.3: Step responses of the system (3.1.3)

The step response is

$$y(t) = 1 - e^{-D\omega_0 t} \left[ \cos(\sqrt{1-D^2}\omega_0 t) + \frac{D-1/b}{\sqrt{1-D^2}} \sin(\sqrt{1-D^2}\omega_0 t) \right] \quad (3.1.6)$$

Again  $\omega_0 t$  can be introduced as scaled time. For  $D = 1/\sqrt{2}$  and some values of  $b$  the pole zero locations and the step responses are shown in Fig. 3.4.

All curves are focussed at  $\omega_0 t = \pi\sqrt{2} = 4.44$ , because here the sine term in (3.1.6) vanishes and it is the only term that contains the varying  $b$ . The curve for  $b \rightarrow \infty$  is identical to the center curve of Fig. 3.3. It is the only one with relative degree two such that (by the initial value theorem of Laplace transforms) the step response begins with slope zero. A zero for  $b = 2$  does not change the response significantly. Even for  $b = 1$  the response is acceptable, but if the zero is closer to the origin than the poles then a large overshoot occurs (40.7% for  $b = 0.5$ ). It could be removed in the step response by cancellation of the zero at  $s = -0.5$ . If we want to avoid this cancellation, then the choices are to reduce the distance  $\omega_0$  of the eigenvalues from the origin (i.e. to reduce the bandwidth) or to increase the damping.

For negative  $b$  the system has nonminimum phase behavior, the step response starts in the negative direction. Cancellation is impossible here because it would give the system an unstable eigenvalue. The unfavorable undershoot can be reduced by canceling the mirror image of the zero by a prefilter  $1/(1-s/(b\omega_0))$ . In the example it is  $0.5/(s+0.5)$ . The step response with this prefilter is shown in dashed lines in Fig. 3.4; it is significantly slower.

Next the mutual influence of poles will be studied. Extend the system (3.1.5) by a low pass filter with a pole at  $-a\omega_0$ , i.e.

$$g(s) = \frac{1}{(1+2Ds/\omega_0 + s^2/\omega_0^2)(1+s/(a\omega_0))} \quad (3.1.7)$$

The step response is

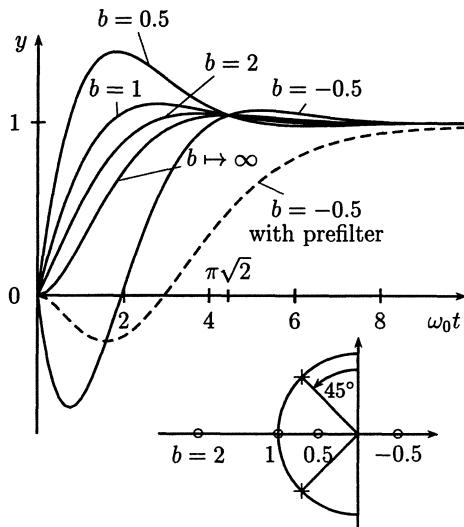


Fig. 3.4: Step response of the system (3.1.5) for different zero locations  $b$

$$y(t) = 1 - \frac{1}{a^2 - 2aD + 1} \left\{ e^{-a\omega_0 t} + e^{-Dt} [a(a - 2D) \cos(\sqrt{1 - D^2}\omega_0 t) + \frac{a(1 + aD - 2D^2)}{\sqrt{1 - D^2}} \sin(\sqrt{1 - D^2}\omega_0 t)] \right\}. \quad (3.1.8)$$

Now a smaller damping may be chosen because the additional pole counteracts a resonance in the frequency response. The chosen value is  $D = 0.5$ , such that the third order Butterworth filter is contained as a special case with  $a = 1$ . For  $a \rightarrow \infty$ , i.e. without the additional pole the response is identical to the curve for  $D = 0.5$  from Fig. 3.3. Fig. 3.5 shows some typical step responses.

The pole for  $a = 2$  has little influence. For  $a = 1$  the overshoot is reduced from 15.5% to 8.1% and for  $a = 0.5$  there is no overshoot. The solution gets slower as the real pole migrates to the right. Finally the real pole dominates and the complex poles show their influence only in the beginning of the step response.

#### Summary

The question, when the step response of a robustly stable system approaches its stationary value depends only on the dominant poles with the smallest distance from the origin  $s = 0$ . Further left and more remote poles and zeros (more than twice the distance) have an influence only on the initial part of the step response, the corresponding time response terms have died out before the entire solution approaches its stationary value. They may have strong influence on the initial actuator signal magnitude. Zeros

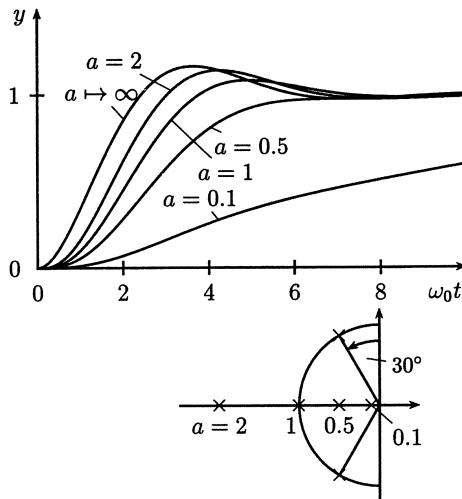


Fig. 3.5: Influence of a pole at  $s = -a$  on the step response

in the left half plane have a similar effect as reduced damping. Their influence can be reduced by cancellation in the prefilter or in the closed loop.

The unfavorable influence of right half plane zeros can be reduced by placing poles in positions that are obtained by reflecting the zeros at the imaginary axis. This cancellation however slows down the solution.

It is recommended to place several closed-loop poles at about the same distance from the origin. If more poles on this radius contribute to the dominant behavior, then the lowest damping may be small.

#### $\Gamma$ -stability

The above discussion of eigenvalue specifications illustrates how the described performance of a control system may be iteratively improved by shifting all eigenvalues into a desired region  $\Gamma$  in the complex plane and motivates the following definition.

*Definition 3.1.* A polynomial  $p(s) = (s - s_1)(s - s_2) \dots (s - s_n)$  is called  $\Gamma$ -stable if all  $s_i \in \Gamma$ .  $\square$

The region  $\Gamma$  has a boundary  $\partial\Gamma$  that consists of one or more contours in the  $s$ -plane such that the membership  $s_i \in \Gamma$  is well defined.

So far our considerations on eigenvalue placement are not directly concerned with robustness. They are for example useful for root locus design for a known plant. They may be useful as well for pole placement. Note that pole placement for single-input systems is primarily attractive as a synthesis method because it yields a unique solution for

state feedback or for the compensator. This supposed advantage in the synthesis procedure, however, turns into a disadvantage for uncertain parameters: A unique solution does not allow any flexibility: if plant parameters change and the same pole locations are prescribed then also the controller must be changed. The preceding discussion on eigenvalue specifications has shown that a precise assignment of all eigenvalues is more than what actually follows from the design specifications. For uncertain parameters it is a reasonable requirement that all eigenvalues are located in a specified region  $\Gamma$  in the  $s$ -plane and must remain in this region under all admissible parameter variations. We call this approach “pole region assignment”.

Fig. 3.6 shows a pole region  $\Gamma$ . Its bandwidth is bounded by a circular arc and its damping by a hyperbola that guarantees damping according to the asymptotes and negative real part.

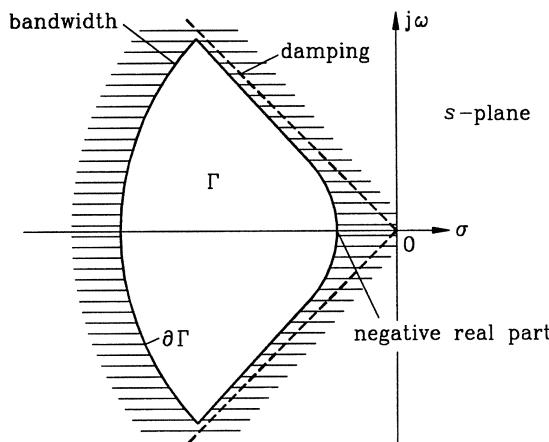


Fig. 3.6: Specification of a pole region  $\Gamma$  that guarantees damping, negative real part and bandwidth limitation

The pole region does not have to be connected, we may also allow variations around a nominal pole pattern. In Fig. 3.7, for example, the dominating behavior is specified by a Butterworth pole configuration for three eigenvalues, and all other eigenvalues must have a real part smaller than  $-a$ . If  $\Gamma$  is not connected, then we also have to specify the number of poles in each of the disjoint parts of  $\Gamma$ .

In our examples of mechanical systems the plant input  $u$  is usually a force for which actuator constraints like  $|u| \leq 1$  have to be considered. Therefore, we are interested in low gain solutions (“soft control”) that provide sufficient damping but do not use control energy to change the natural frequencies.

If parameter variations change the distance of plant eigenvalues from the origin, then also the closed-loop eigenvalues should vary in about the same distance. Fig. 3.8 shows the result of a robust controller design for the crane with moderately large feedback gains. It has the parameters  $m_C = 1000$  [kg],  $\ell = 10$  [m],  $m_L \in [50 ; 2395]$  [kg]. The

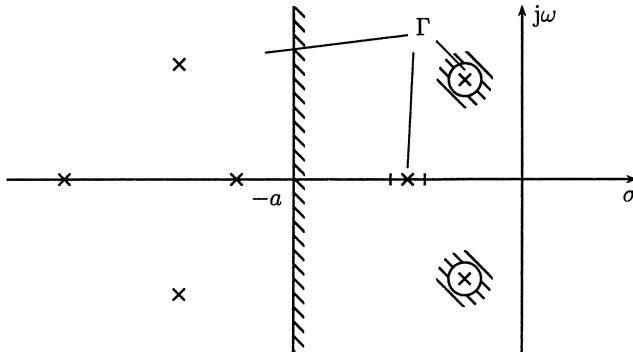


Fig. 3.7: A dominating Butterworth pole configuration should be approximately preserved under parameter variations

robust output feedback controller  $u = -[500 \ 2769 \ -21557 \ 0] \mathbf{x}$  was designed in [5] such that all eigenvalues are on the left side of the left branch of the hyperbola  $(\sigma/0.25)^2 - (\omega/0.5)^2 = 1$ . The plant has a pair of imaginary eigenvalues. A small load  $m_L = 50$  [kg] is swinging slowly, the eigenvalues ( $\Delta$ ) are close to the origin. By  $\Gamma$ -stabilization they are shifted left just across the hyperbolic boundary, the two eigenvalues at  $s = 0$  are shifted to the negative real axis. A large load  $m_L = 2395$  [kg] is vibrating with a higher frequency. The open-loop eigenvalues ( $\square$ ) are further remote from  $s = 0$ . In the closed loop the imaginary eigenvalue is shifted to the left side of the hyperbola and there occurs a second complex conjugate pair of eigenvalues on the hyperbola. It dominates the slow terms of the transient motion. We formulate this observation as the **second basic rule of robust control**:

*When you close a loop with actuator constraints, leave a slow system slow and leave a fast system fast.*

For the flight control problem it is not necessary to translate time domain specifications into eigenvalue specifications. Based on pilot ratings admissible intervals for damping and frequency of the short period mode have been determined and must be verified in the qualification tests for the certification of a new aircraft [1]. For the characteristic polynomial  $p(s) = a_0 + a_1 s + s^2 = \omega_0^2 + 2D\omega_0 s + s^2$  the following bounds are prescribed

$$0.35 \leq D \leq 1.3; \quad \omega_a \leq \omega_0 \leq \omega_b \quad (3.1.9)$$

The requirements for the eigenvalues can be formulated better in the  $s$ -plane. The specification  $\omega_a < \omega < \omega_b$  describes an annulus. The requirement that  $D$  is greater than 0.35 cuts out a segment which resembles a pineapple segment  $\Gamma$ , see the solid line in Fig. 3.9. The upper bound for  $D$  will be discussed in Chapter 11.

A difficulty arises from the fact that the specification (3.1.9) for the short-period mode was initially made for the uncontrolled aircraft. If we design a control system to have such eigenvalues in the closed loop, then additional eigenvalues arise from actuator and

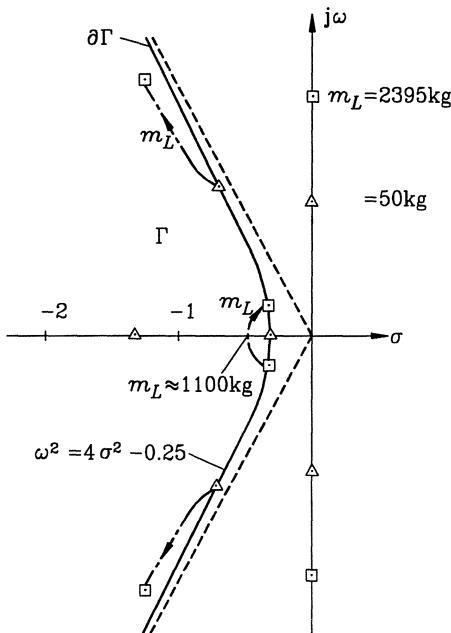
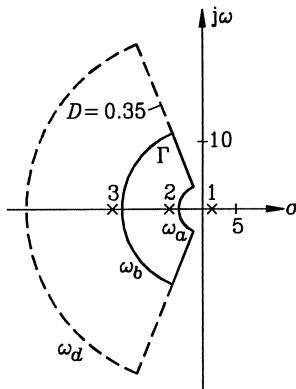


Fig. 3.8: A fast system remains fast, a slow system remains slow

controller dynamics. The simplest approach to guarantee satisfaction of (3.1.9) for the short period mode is to require that *all* eigenvalues are located in  $\Gamma$ . Then, however, the actuator pole must be shifted unnecessarily far to the right. In the open loop the actuator pole is at  $s = -14$  to the left of  $\Gamma$ . The uncontrolled short period mode has two real eigenvalues, one in  $\Gamma$  and one in the right half plane as indicated in Fig. 3.1.

Here the question arises, when we can still distinguish short period and actuator eigenvalues after closing the loop. Suppose there is no additional feedback dynamics and the output feedback vector  $k_y$  of (2.3.18) is changed continuously starting from  $k_y = [k_{nz} \ k_q]^T = [0 \ 0]^T$ . Clearly the distinction gets lost if the left short period eigenvalue “marries” the actuator pole, i.e. moves across a branching point to form a complex conjugate pair. Such a path in the  $k_y$ -space must be avoided. If, however, the two short period eigenvalues form a complex pair and are moved into  $\Gamma$ , then we can still identify them as the poles resulting in the closed loop from the short period mode. We extend the original  $\Gamma$  now by a third “bandwidth” circle as indicated in Fig. 3.1. The boundary  $\omega_b$  separates the short period eigenvalues from all other eigenvalues originating from the actuator and possibly from a dynamic compensator.

Fig. 3.9: Nice stability in the  $s$ -plane

## 3.2 Introduction to Robustness Analysis

The basic question of robustness analysis is:

*Is a polynomial family  $P(s, Q)$  stable (or  $\Gamma$ -stable)?*

The classical stability tests primarily apply to a specific polynomial  $p(s, \mathbf{q}^{(1)})$  with numerically given  $\mathbf{q} = \mathbf{q}^{(1)}$ . One might think of repeating such test many times for a large number of grid points  $\mathbf{q}^{(i)} \in Q$ . But in this way we can never be sure that there does not exist an unstable point in  $Q$ . A pragmatic approach is to use a graphical visualization that does not only show the yes-no answer to the stability question for grid points but also gives an idea about “closeness to instability”. Examples are:

1. Show the root locations in the  $s$ -plane and see if they come close to the stability boundary.
2. Plot a large number of Nyquist plots for a grid of parameters and see how close they come to the critical point  $-1$ .
3. For only two parameters  $q_1$  and  $q_2$  we may plot the stability region in the  $(q_1, q_2)$ -plane and see if  $Q$  is contained in it.

The graphical representation may give more or less confidence in the sufficient representation of the continuum of parameters by the chosen grid points. Several graphical stability tests for uncertain polynomials are introduced in Chapter 4. Another approach is the use of symbolic calculations of Hurwitz determinants. This leads to nonlinear inequalities in the uncertain parameters and in general it is difficult to check if they are satisfied for all  $\mathbf{q} \in Q$ . In simple cases, however, Hurwitz determinants may even show the worst case of the parameter values.

*Example 3.1.* Consider the crane with state feedback. By (2.2.19) the third Hurwitz stability condition is

$$\Delta_3 = [k_2(m_L g - k_3) + k_1 k_4](k_2 \ell - k_4) - k_2 k_4 m_C g > 0 \quad (3.2.1)$$

Notice that the parameters  $m_L$ ,  $m_C$ ,  $\ell$ ,  $k_1$ , and  $k_3$ , enter linearly into  $\Delta_3$ , i.e. the critical value is either their minimum or their maximum value (depending on the sign of the factor of these uncertain parameters). In contrast,  $k_2$  and  $k_4$  enter quadratically into  $\Delta_3$ , these parameters may yield a worst case value in the intervals  $k_2 \in [k_2^- ; k_2^+]$  and  $k_4 \in [k_4^- ; k_4^+]$ .

A further analysis shows that among the stable systems the empty hook  $m_L = m_L^-$  is the worst case because  $m_L$  appears with a positive factor  $k_2 g a_3 = k_2 g (k_2 \ell - k_4)$  in  $\Delta_3 > 0$  and with a positive factor  $g$  in  $a_2 > 0$  of (2.2.18).  $\square$

The above example suggests that there are “nice” parameters  $q_i$ , for which it suffices to check stability for one or both of their extremal values  $q_i^-$  and  $q_i^+$  and there are “nasty” parameters for which the worst case may be in the interval  $q_i \in [q_i^- ; q_i^+]$ . Only the nasty parameters must be analyzed as a continuum. The question of “extremal value results” is of current research interest and some practically useful results will be presented in Chapters 5 and 8.

Another approach to robustness analysis uses the idea of a stability radius in  $q$ -space. Starting from the stable center  $q^0$  of the operating domain, a box (or ball) is blown up until it touches the stability boundary. This approach is treated in Chapter 7.

The robustness analysis is relatively easy for interval coefficients (2.7.3) and for the affine case Req2.7.4. It is much more complicated for the multilinear and polynomial cases. In these nonlinear cases some simplifications occur, if specific uncertainty structures of the polynomial can be exploited, see Chapter 6. Also, the case of only two uncertain parameters can be handled in the polynomial case, see Chapter 9. An even more difficult case arises when a continuous plant is controlled digitally. In the discretization process the uncertain plant parameters enter exponentially into the characteristic polynomial. An approximation for this case is given in Chapter 10.

As a rule we avoid approximations and overbounding in robustness analysis. We may have to grid some parameters or the frequency  $\omega$  in frequency domain approaches. With these limitations we strive for a “yes” or “no” answer to the basic question of robustness analysis. For this reason we do not cover conservative estimates, which tell us only if we are fortunate that a particular polynomial family is robustly stable, but in other cases give no conclusive answer.

Examples of overbounding for real control engineering robustness problems are:

- i) State space models with norm bounded perturbation matrices,  
e.g.  $\mathbf{A} + \Delta \mathbf{A}$  ,  $\|\Delta \mathbf{A}\| < R$ ,
- ii) State space models in form of interval matrices with elements  $a_{ij} \in [a_{ij}^- ; a_{ij}^+]$ ,
- iii) Overbounding of real uncertainties by complex ones,

- iv) Embedding an uncertain polynomial with dependent coefficients into an interval polynomial family.

*Example 3.2.* Consider the crane with yet undetermined output feedback  $u = -[k_1 \ k_2 \ 0 \ 0]\mathbf{x}$ . We know from (2.2.21) that it is stable if and only if  $k_1 > 0$ ,  $k_2 > 0$ . Now let  $m_C = 1000$  [kg],  $m_L = 1$  [kg],  $g = 10$  [m · s<sup>-2</sup>],  $\ell \in [9.99 ; 10.01]$  [m], i.e. only the rope length is uncertain, it varies by ±1 centimeter around its nominal length of 10 [m]. The coefficient values or intervals are

$$\begin{aligned} a_0 &= k_1 g \in [10k_1^- ; 10k_1^+] \\ a_1 &= k_2 g \in [10k_2^- ; 10k_2^+] \\ a_2 &= (m_L + m_C)g + k_1 \ell \in 10010 + [9.99k_1^- ; 10.01k_1^+] \\ a_3 &= k_2 \ell \in [9.99k_2^- ; 10.01k_2^+] \\ a_4 &= \ell m_C \in [9990 ; 10010] \end{aligned} \quad (3.2.2)$$

We overbound this affine polynomial by pretending that the coefficients  $a_0$  to  $a_4$  vary independently between their respective lower and upper bounds. Substitute these intervals into the Hurwitz condition

$$\Delta_3 = a_1 a_2 a_3 - a_0 a_3^2 - a_1^2 a_4$$

and obtain  $\Delta_3 \in [\Delta_3^- ; \Delta_3^+]$  with

$$\Delta_3^- = 10k_2^2[(9.99^2 - 10.01^2)k_1 - 100.1]$$

$$\Delta_3^+ = 10k_2^2[(10.01^2 - 9.99^2)k_1 + 300.1]$$

There is no positive  $k_1$  such that  $\Delta_3^- > 0$ . The surprising result is that stability cannot be shown for any  $(k_1, k_2)$  by this conservative estimate. An explanation can be given by the transfer function (1.1.10) with the pole zero pattern of Fig. 1.2. If we choose  $\ell = \ell^-$  in the numerator and  $\ell = \ell^+$  in the denominator, then  $g_C(s)$  has zeros at  $s_0 = \pm j\sqrt{g/\ell^-}$  with

$$\frac{g}{\ell^-} = \frac{10}{9.99} > 1$$

and poles at  $s_p = \pm j\sqrt{g/\ell^+}\sqrt{1 + m_L/m_C}$  with

$$\frac{g}{\ell^+} \left(1 + \frac{m_L}{m_C}\right) = \frac{10.01}{10.01} = 1$$

Thus, the conservative estimate contains cases, where the zero has a larger distance from  $s = 0$  than the pole. By this exchange of pole and zero the initial direction of the root locus is turned around by 180°. The above case does not describe a crane and there is no need to stabilize it by  $(k_1, k_2)$ .  $\square$

Admittedly, the above example was specifically constructed to show an extreme effect of overbounding. For a different interval of rope lengths we may obtain a simple sufficient

condition for some  $(k_1, k_2)$ -region from overbounding by an interval polynomial. Practically overbounding is meaningful if it results in a very simple test. Among the cases i) to iv) listed above this is true only for iv), overbounding by an interval polynomial.

In the literature on robust stability analysis there are some confusing distinctions between “unstructured” versus “structured” or even “highly structured” uncertainty, usually relating to different norms on the size of the uncertainty of standard model structures in frequency domain or coefficient space. We avoid this classification and speak only about the “physically motivated” uncertainty (that should be obvious from the examples in Chapter 1) versus “mathematically motivated” uncertainty. In order to obtain a physically motivated uncertainty structure we have to go through analytical plant modelling. For a general control theory it is of course more convenient to begin with a standard nominal model (e.g. state space model, transfer function, frequency response, impulse response) and to superimpose a norm bounded uncertainty. There are situations, where we have no other choice, because analytical modelling is impossible or leaves some effects unmodelled.

In the examples crane and car steering we have dealt with the effects of large parameter variations in a known model structure. In addition there may exist model uncertainties resulting from unmodelled dynamics. The model of the crane for example ignores

- the electric motor that generates the force  $u$  for acceleration of the crab,
- the elasticity and mass of the rope.

In the car steering model we have ignored

- the coupling with the vertical motion (heave, pitch, roll),
- the mass of the wheels,
- acceleration of the car and the related power train dynamics.

If we want to model the above effects, then we obtain a complicated higher-order model with higher dimensional state space and higher degree of the transfer function. Frequency domain methods are advantageous for such dynamic model uncertainties including order uncertainty. A classical approach to consider model uncertainties is to design the closed-loop control system with sufficient gain and phase margin. Then it is guaranteed that phase delays caused by neglected actuator dynamics or by a sample and hold in a digital controller realization does not lead to instability. For high frequencies, the magnitude of the open-loop frequency response should decay rapidly with frequency in order to avoid frequency response resonances of higher harmonic structural vibrations. In some instances we consider such countermeasures to the effects of modelling uncertainties. The main focus of this book is however on uncertain physical parameters in known model structures.

### 3.3 Introduction to Robust Controller Design

The basic question of robust synthesis is:

*Given a polynomial family  $P(s, Q, k)$ , find a  $k = k^*$  such that  $P(s, Q, k^*)$  is robustly  $\Gamma$ -stable.*

If there exists such a  $k^*$ , then there is also a neighborhood of  $k^*$  that  $\Gamma$ -stabilizes the plant family. Thus,  $k$  may be varied, starting from  $k^*$  in order to improve additional design criteria. Conceptually, we are interested in a description of the set of all  $\Gamma$ -stabilizers

$$K_\Gamma = \{ k \mid P(s, Q, k) \text{ is robustly } \Gamma\text{-stable} \} \quad (3.3.1)$$

Then, other design requirements can be employed to select a specific  $k \in K_\Gamma$ . Fig. 3.10 illustrates the robust design problem for a situation with only one uncertain parameter  $q \in Q$  and one free gain  $k$ .

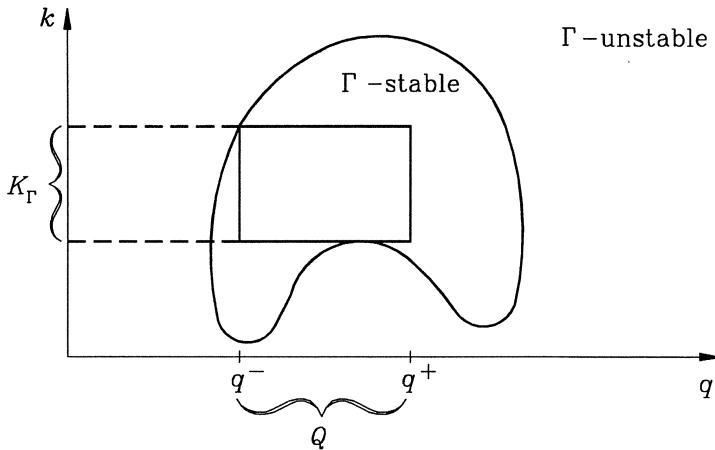


Fig. 3.10:  $\Gamma$ -stabilizing controllers in  $K_\Gamma$  yield  $\Gamma$ -stability for all parameter values  $q \in Q$

The  $\Gamma$ -stability region in the  $(q, k)$ -plane is nonconvex. Therefore, it does not suffice to find a simultaneously  $\Gamma$ -stabilizing  $k$  for  $q^-$  and  $q^+$ . In Fig. 3.10  $K_\Gamma$  can be constructed graphically. Fig. 3.11 shows a situation, where the set  $K_\Gamma$  is empty.

From a mathematical point of view the synthesis problem is well-posed but unsolved because there are no necessary and sufficient conditions for the existence of a robustly  $\Gamma$ -stabilizing controller. Thus, the engineer needs some optimism to develop design tools and to apply them to examples.

If we have exact methods for robustness analysis, then we can afford to be optimistic during the design phase. Every approach is admissible if the result is analyzed afterwards. We formulate the **third basic rule of robust control**:

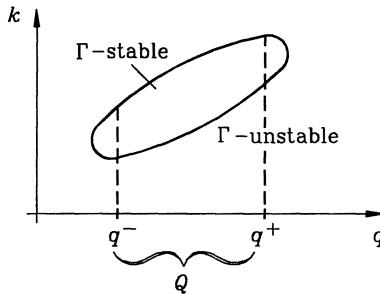


Fig. 3.11: There does not exist a simultaneous stabilizer  $k$  for all  $q \in Q$

*Be a pessimist in analysis, then you can afford to be an optimist in design.*

It is a matter of personal preference, education, experience of the design engineer and available software tools which of the many design approaches in the literature is preferred. There are several toolboxes for Matlab for the design of robust controllers and new proposals and algorithms keep appearing. Most of these approaches are based on conservative estimates of the worst case perturbation. An exception is the “Quantitative Feedback Theory” by Horowitz [86]. It is based on a frequency domain representation of the uncertainty. It is particularly suited for engineers who have experience in the design of control systems by Nichols charts.

Robust design itself may be viewed as a feedback process [120] involving robustness analysis as illustrated by Fig. 3.12.

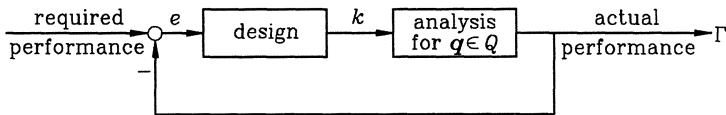


Fig. 3.12: Design process viewed as a feedback process involving analysis

The actual and required performances can be measured in different ways and different strategies result from the performance measures.

i) Robustness in  $s$ -plane.

$Q$  is given. The actual root set for all  $q \in Q$  is represented in the  $s$ -plane. In the next design step this set is contracted such that it better fits into the desired pole region  $\Gamma$ . If this cannot be achieved, then the required  $\Gamma$  may be relaxed.

ii) Robustness in  $q$ -space

$\Gamma$  is given. The actual  $\Gamma$ -stable region

$$Q_\Gamma = \{q \mid p(s, q, k) \text{ is } \Gamma\text{-stable}\} \quad (3.3.2)$$

is compared with the desired robust parameter region  $Q$ . In the next step  $Q_\Gamma$  is enlarged such that it covers  $Q$  as well as possible. If this cannot be achieved we may have to reduce the operating domain that can be controlled by a fixed-gain controller.

### iii) Robustness in the Nyquist-plane

$Q$  is given. The Nyquist value set (union of Nyquist loci for all  $q \in Q$ ) of the open loop is plotted. In the next design step,  $\mathbf{k}$  is chosen such that the smallest distance of the Nyquist value set from the critical point  $-1$  is enlarged. If this cannot be achieved, then requirements on gain and phase margin, minimal singular value, etc., may be relaxed.

Most design procedures use a one-dimensional measure  $\|\mathbf{e}\|$  for the deviation between desired and actual performance. Then a standard optimization  $\min_k \|\mathbf{e}\|$  can be performed, where many combinations of norms and optimization routines are possible. Two-dimensional measures of the deviation between desired and actual performance can be judged by the designer using computer graphics.

From an engineering point of view, design is appropriately described as a multi-dimensional tradeoff between several design requirements. A systematic search in higher dimensional spaces may be supported by visualization of vector valued performance indices and by Pareto optimization techniques.

In Part IV of this book we will deal with two design tools for robust control systems that were developed and successfully applied by the authors. The optimistic approach to design that we take is triggered by the observation that all the nasty examples that we can construct (e.g. in Chapter 5 we show an example with an isolated unstable point  $q$  inside an otherwise stable  $Q$ -box) are not typical for realistic applications. It is much more likely that  $\Gamma$ -instability first occurs at one of the vertices of the  $Q$ -box if we change controller parameters. Therefore, a controller that simultaneously stabilizes the vertices of the  $Q$ -box is a good candidate for further robustness analysis (and improvement if necessary). In fact under some restrictive assumptions it can even be shown that it suffices to simultaneously  $\Gamma$ -stabilize the vertices of the  $Q$ -box, see Chapter 8. If such assumptions are not satisfied, then we may be in a situation as illustrated by Fig. 3.13.

Initially, the vertices A, B, C, and D, were chosen as representatives of the operating domain  $Q$ . Assume that we find a fixed gain controller that simultaneously  $\Gamma$ -stabilizes the four vertices. In the following analysis step the exact stability region  $Q_\Gamma$  (dashed boundary) is determined. By construction, the  $\Gamma$ -stable region must include the points A, B, C, and D, but not necessarily the entire  $Q$ . In the situation of Fig. 3.12 it is an obvious empirical approach to repeat the design with the additional representative E. There is no guarantee of success with this procedure of alternating synthesis and analysis steps. At least the problem with a continuum of parameter values has been reduced to a multi-model formulation with good chances of success. In the synthesis step only a finite family of plants must be stabilized simultaneously. This “multi-model problem” can be treated for example by simultaneous  $\Gamma$ -stabilization or “pole region assignment”.

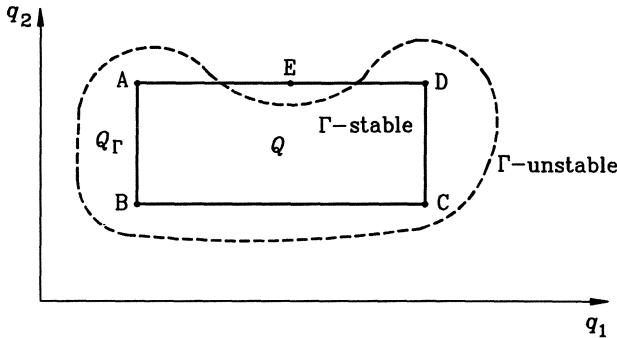


Fig. 3.13: Stability of the vertices ABCD does not guarantee stability of the rectangle ABCD

Pole assignment is a standard design technique for control systems. For simplicity it will be discussed here only for the single-input case.

Given

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{bu}, \quad (\mathbf{A}, \mathbf{b}) \text{ controllable} \quad (3.3.3)$$

Find a state-feedback law

$$u = -\mathbf{k}^T \mathbf{x} + w \quad (3.3.4)$$

such that the closed-loop system

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{b}\mathbf{k}^T)\mathbf{x} + \mathbf{bw} \quad (3.3.5)$$

has a specified set of real or complex conjugate eigenvalues  $s_1, s_2, \dots, s_n$ , i.e. the closed-loop characteristic polynomial is

$$p(s) = \det(s\mathbf{I} - \mathbf{A} + \mathbf{b}\mathbf{k}^T) = (s - s_1)(s - s_2) \dots (s - s_n) \quad (3.3.6)$$

There are several techniques for solving this equation for  $\mathbf{k}^T$  [99]. One of them is solving (2.2.12) with the coefficient vector  $\mathbf{a}^T$  resulting from (3.3.6). Shifting the individual eigenvalues is more transparent by Ackermann's formula [2]

$$\mathbf{k}^T = \mathbf{e}^T p(\mathbf{A}) \quad (3.3.7)$$

where

$$\mathbf{e}^T = [0 \dots 0 \ 1][\mathbf{b} \ \mathbf{Ab} \ \dots \ \mathbf{A}^{n-1}\mathbf{b}]^{-1} \quad (3.3.8)$$

Using the factorized form of  $p(s)$  in (3.3.6)

$$\mathbf{k}^T = \mathbf{e}^T (\mathbf{A} - s_1 \mathbf{I})(\mathbf{A} - s_2 \mathbf{I}) \dots (\mathbf{A} - s_n \mathbf{I}) \quad (3.3.9)$$

The standard use of pole assignment is to specify the desired closed-loop eigenvalues  $s_1, s_2, \dots, s_n$ , and to calculate the required feedback gain vector  $\mathbf{k}$ . In pole region assignment only an admissible region  $\Gamma$  in  $s$ -plane is specified, i.e. it is only required that

$$s_1, s_2, \dots, s_n \in \Gamma \quad (3.3.10)$$

This form of specification is more natural for the designer, because generally he does not know how to choose the  $s_i$  exactly, but he has a good idea about  $\Gamma$ . The solution of (3.3.9) is then an admissible set  $K_{\Gamma}^{(1)}$  such that

$$s_1, s_2, \dots, s_n \in \Gamma \iff \mathbf{k} \in K_{\Gamma}^{(1)} \quad (3.3.11)$$

In comparison with pole assignment, pole region assignment offers more flexibility for simultaneous  $\Gamma$ -stabilization of a family of plant models  $(A_j, b_j)$ ,  $j = 1, 2, \dots, N$ , by one fixed gain vector  $\mathbf{k}$ . Each plant model  $(A_j, b_j)$  gives rise to a corresponding admissible region  $K_{\Gamma}^{(j)}$  in  $\mathbf{k}$ -space. The set of simultaneous  $\Gamma$ -stabilizers is the intersection

$$K_{\Gamma} = \bigcap_{j=1}^N K_{\Gamma}^{(j)} \quad (3.3.12)$$

Fig. 3.14 illustrates (3.3.12) for the case of two plant models.  $\mathbf{k} \in K_{\Gamma}^{(1)}$  places all

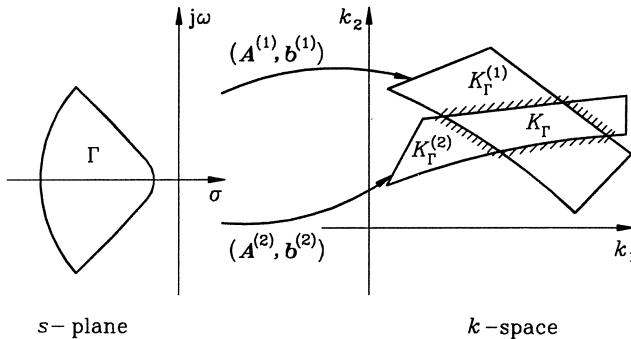


Fig. 3.14: A necessary and sufficient condition for the controller gains  $k_1$  and  $k_2$  to stabilize the plants  $(A_1, b_1)$  and  $(A_2, b_2)$  simultaneously is  $\mathbf{k} \in K_{\Gamma}$

eigenvalues of  $(A_1 - b_1 \mathbf{k}^T)$  into  $\Gamma$ ,  $\mathbf{k} \in K_{\Gamma}^{(1)}$  places all eigenvalues of  $(A_2 - b_2 \mathbf{k}^T)$  into  $\Gamma$ . The intersection  $K_{\Gamma} = K_{\Gamma}^{(1)} \cap K_{\Gamma}^{(2)}$  describes the set of simultaneous  $\Gamma$ -stabilizers for the two plant models.

A graphical representation of  $K_{\Gamma}$  can be made in two-dimensional [3, 5] and three-dimensional [142] cross sections of  $\mathbf{k}$ -space. The choice of the cross-section may be determined by the controller structure [68] or by the choice of invariance planes [20], such that some eigenvalues are not shifted in one design step. It may also be chosen such that the admissible area or volume in the subspace is maximized [142]. The design point can be selected from the admissible set in consideration of additional design requirements [3, 5]

- small  $\|\mathbf{k}\|$  in order to reduce  $|u|$ ,
- a safety margin for  $\mathbf{k}$  away from the boundaries of  $K_{\Gamma}$  for the case of implementation inaccuracies (e.g. quantization),

- robustness with respect to sensor failures,
- gain reduction margins.

We will come back to these issues in Chapter 11.

### 3.4 Three Basic Rules of Robust Control

At the end of Part I we summarize the three basic rules of robust control that were postulated in the first three chapters:

**Rule 1**

Require robustness of a control system only for physically motivated parameter values and not with respect to arbitrarily assumed uncertainties of the mathematical model.

**Rule 2**

When you close a loop with actuator constraints, leave a slow system slow and leave a fast system fast.

**Rule 3**

Be a pessimist in analysis, then you can afford to be an optimist in design.

### 3.5 Exercises

- 3.1. For the example of Fig. 3.8 find the eigenvalue locations of the extremal operating conditions if the output feedback law is changed to

$$u = - \begin{bmatrix} 500 & 2000 & -20000 & 0 \end{bmatrix} x$$

- 3.2. Consider Example 3.2. Change the rope length interval to  $\ell \in [4; 6]$  [m]. Use overbounding by the assumption of independent coefficients. Sketch the stability boundary of overbounded polynomial in the  $(k_1, k_2)$ -plane.

## **Part II**

# **Stability Analysis of Polynomial Families**

## 4 Classical Stability Tests Applied to Uncertain Polynomials

It is well known that a linear time-invariant system is stable if the roots of its characteristic polynomial have a negative real part. In short we speak of “stability of a polynomial”.

With Chapter 4 we enter into Part II on stability analysis of polynomial families. Part II is organized by methods and the emphasis is shifting from the engineering art (Part I) to engineering science. For clarity of presentation all methods are introduced for the case of stability. Generalizations to Gamma-stability will be given in Chapter 9.

This chapter looks at methods for testing the stability of polynomials. One of the purposes is to review some classical ways of testing the stability of a given polynomial. A more important goal is to present the extensions of these test methods to families of polynomials

$$P(s, Q) = \{ p(s, \mathbf{q}) \mid \mathbf{q} \in Q \} \quad (4.0.1)$$

generated by an uncertain polynomial

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + a_2(\mathbf{q})s^2 + \dots + a_n(\mathbf{q})s^n \quad (4.0.2)$$

It is assumed that the coefficients

$$a_0(\mathbf{q}), a_1(\mathbf{q}), a_2(\mathbf{q}), \dots, a_n(\mathbf{q}) \quad (4.0.3)$$

are real and depend continuously on a constant but unknown real parameter vector

$$\mathbf{q} = [q_1 \ q_2 \ \dots \ q_\ell]^T \quad (4.0.4)$$

It will also be assumed that independent bounds are given for each parameter

$$q_i \in [q_i^- ; q_i^+] \quad (4.0.5)$$

As a result, the set of possible parameter vectors is given by

$$Q = \{ \mathbf{q} = [q_1 \ q_2 \ \dots \ q_\ell]^T \mid q_i \in [q_i^- ; q_i^+], i = 1, 2, \dots, \ell \} \quad (4.0.6)$$

Vertices and edges of the hyperrectangle  $Q$  are of particular interest.

A *vertex* is a parameter vector

$$\mathbf{q}_v = [q_1 \ q_2 \ \dots \ q_\ell]^T \quad \text{with} \quad q_i \in \{q_i^-, q_i^+\}, \ i = 1, 2, \dots, \ell$$

An *edge* is a parameter set

$$Q_E = \{ \mathbf{q} \mid q_i \in [q_i^-, q_i^+], q_j \in \{q_j^-, q_j^+\} \text{ for all } j \neq i \}$$

Unless stated otherwise it will also be assumed in this chapter that

$$a_n(\mathbf{q}) > 0 \quad \text{for all } \mathbf{q} \in Q \tag{4.0.7}$$

and hence the degree of the polynomial does not vary in the operating domain  $Q$ .

*Remark 4.1.* Consider the polynomial family  $P(s, Q) = \{p(s, q) = 1 + qs \mid q \in [-1; 1]\}$ . Vary  $q$  from  $+1$  to  $-1$ . A stable root, initially at  $s = -1$ , moves along the real axis to  $-\infty$ , comes back on the positive real axis and ends at the unstable root at  $s = 1$ . Thus, a stability boundary has been crossed for  $q = 0$ . The assumption (4.0.7) restricts the uncertainty interval to  $[-1; \varepsilon]$  or  $[\varepsilon; 1]$ . For the physical systems of Chapter 1 the assumption (4.0.7) rules out singular perturbations like  $a_4 = \ell m_C = 0$  for the crane (Example 2.8) or  $a_5 = J^2 v^2 / \omega_0^4 = 0$  for the automatic steering system (Example 2.10). Whenever we talk about an uncertain polynomial (4.0.2) (without a given  $Q$ ), then

$$a_n(\mathbf{q}) = 0 \tag{4.0.8}$$

is a validity boundary in the coefficient space and induces a validity boundary in  $\mathbf{q}$ -space. The significance of a validity boundary is illustrated by Example 2.7 of a crane with  $a_n(\mathbf{q}) = \ell m_C$ . The condition  $\ell = 0$  or  $m_C = 0$  does not necessarily imply that there is an eigenvalue at infinity. In fact, for the singular perturbation  $\ell = 0$  the system is only of second order. However, for  $\ell < 0$  (the inverted pendulum) the system is unstable and the validity boundary has the same effect as a stability boundary. For robust stability of a  $Q$ -box it must not be intersected by a validity boundary. In our further treatment we will not distinguish between validity boundaries and stability boundaries.  $\square$

The assumptions used in this chapter are not very restrictive. A large number of physical systems, such as the crane and car steering examples in Part I, meet these mild assumptions. Therefore, the stability tests presented in this chapter are useful for analyzing the robustness properties of uncertain systems.

If a linear time-invariant system has the characteristic polynomial  $p(s, \mathbf{q})$  and this polynomial has all its roots in the open left half plane for a particular parameter vector  $\mathbf{q} = \mathbf{q}^*$ , then, in short, we call  $\mathbf{q}^*$  a “stable operating point”. Similarly, we also talk about “stable edges” or “stable vertices” of the  $Q$ -box.

## 4.1 Root Set Problem Formulation

This section presents a stability definition and analysis method for families of polynomials. The development begins by recalling the traditional root location stability definition.

For a known polynomial, the definitions of roots and of stability are stated as follows. If  $p(s)$  is an  $n$ -th order polynomial

$$p(s) = a_0 + a_1 s + a_2 s^2 + \cdots + a_n s^n, \quad a_n > 0, \quad (4.1.1)$$

then it can be factored into  $n$  terms.

$$p(s) = a_n \prod_{i=1}^m (s - \sigma_i - j\omega_i)(s - \sigma_i + j\omega_i) \prod_{k=2m+1}^n (s - \sigma_k) \quad (4.1.2)$$

The  $2m$  complex numbers  $\sigma_1 + j\omega_1, \sigma_2 + j\omega_2, \dots, \sigma_m + j\omega_m, \sigma_1 - j\omega_1, \sigma_2 - j\omega_2, \dots, \sigma_m - j\omega_m$  and the  $n - 2m$  real numbers  $\sigma_{2m+1}, \sigma_{2m+2}, \dots, \sigma_n$  are the roots of  $p(s)$ . A polynomial that characterizes a continuous-time system is called stable if all its roots are contained in the open left half plane

$$\mathbb{C}^- = \{ s \in \mathbb{C} \mid \operatorname{Re}[s] < 0 \} \quad (4.1.3)$$

Conversely, if one or more roots are not in the open left half plane, then the polynomial is said to be unstable. (Note, that this definition is equivalent to "asymptotic stability" in the sense of Lyapunov.)

Based on this definition, the obvious way to determine the stability of a polynomial is to find all its roots. To denote the set of all roots of the polynomial  $p(s)$ , the following notation will be used:

$$\operatorname{Roots}[p(s)] = \{ v \in \mathbb{C} \mid p(v) = 0 \} \quad (4.1.4)$$

Once the set of roots is found, the stability condition

$$\operatorname{Roots}[p(s)] \subset \mathbb{C}^- \quad (4.1.5)$$

can be determined by inspection. Graphical inspection is typically the most aesthetically appealing approach. Each of the elements in  $\operatorname{Roots}[p(s)]$  is plotted in the complex plane. Given this plot, visual inspection is often adequate to determine whether or not all the roots are in the open left half plane. Availability of convenient software packages that can do both the computing and the plotting has bolstered the use of this stability analysis method.

*Example 4.1.* This analysis method is demonstrated on some examples. Consider the following three polynomials

$$\begin{aligned} p_1(s) &= 81 + 9s + 16s^2 + s^3 + s^4 \\ p_2(s) &= 192 + 196s + 77s^2 + 14s^3 + s^4 \\ p_3(s) &= 36 + 43s + 25s^2 + 7s^3 + s^4 \end{aligned} \quad (4.1.6)$$

The computed values of the roots of these polynomials are shown below.

$$\begin{aligned} \operatorname{Roots}[p_1(s)] &= \{-1 + j2.828, -1 - j2.828, 0.5 + j2.958, 0.5 - j2.958\} \\ \operatorname{Roots}[p_2(s)] &= \{-3, -4, -3.5 + j1.937, -3.5 - j1.937\} \\ \operatorname{Roots}[p_3(s)] &= \{-1.5 + j1.323, -1.5 - j1.323, -2 + j2.236, -2 - j2.236\} \end{aligned}$$

Fig. 4.1 shows the roots of  $p_1(s)$ ,  $p_2(s)$ , and  $p_3(s)$  plotted in the complex plane using pentagons, squares, and triangles, respectively. From the figure, it is clear that  $p_2(s)$  and  $p_3(s)$  have all their roots in the open left half plane and are therefore stable. It is also apparent that  $p_1(s)$  has a pair of roots in the right half of the complex plane and is therefore unstable.  $\square$

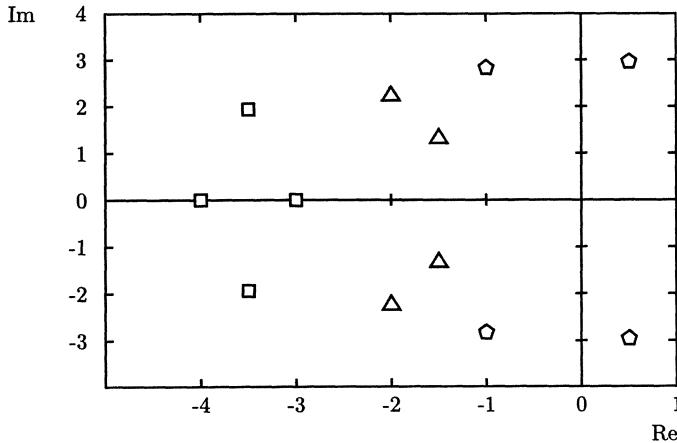


Fig. 4.1: Roots of  $p_1(s)$  - pentagons,  $p_2(s)$  - squares, and  $p_3(s)$  - triangles

The root location stability analysis method can be extended from individual polynomials to families of polynomials. To do this, it is first necessary to formally define what is meant by stability of a family of polynomials. The definition used in this book is referred to as robust stability. It is based on a worst-case consideration. Given a family of polynomials  $P(s, Q)$ , if every polynomial in  $P(s, Q)$  is stable, then  $P(s, Q)$  is said to be robustly stable. Conversely, if one or more polynomials in  $P(s, Q)$  are unstable, then  $P(s, Q)$  is not robustly stable.

Stability of an uncertain polynomial can be tested in basically the same way that was given above for known polynomials. The set of all possible roots of a polynomial family  $P(s, Q)$  will be denoted as

$$\text{Roots}[P(s, Q)] = \{ v \in \mathbb{C} \mid p(v, q) = 0, p(s, q) \in P(s, Q) \} \quad (4.1.7)$$

The polynomial family is robustly stable if and only if the root set is contained in the left half plane:

$$\text{Roots}[P(s, Q)] \subset \mathbb{C}^- \quad (4.1.8)$$

The similarity of the stability conditions for the known case (4.1.5) and the uncertain case (4.1.8) is easily seen. As in the known case, once the set of possible roots is ascertained, robust stability can usually be determined by inspection. Again, the graphical approach of plotting the root set in the complex plane is generally the most aesthetically appealing.

*Example 4.2.* To make sure that the definition of robust stability is clear, two simple examples shall be investigated. Using the polynomials given in (4.1.7), define the following two polynomial families

$$P_a(s) = \{ p_2(s), p_3(s) \} \quad (4.1.9)$$

$$P_b(s) = \{ p_1(s), p_2(s), p_3(s) \} \quad (4.1.10)$$

For  $P_a(s)$ , the root set Roots[ $P_a(s)$ ] is given by the union of the triangles and squares in Fig. 4.1. All roots of  $P_a(s)$  are in the open left half plane, so this polynomial family is robustly stable. For the larger set  $P_b(s)$ , the root set Roots[ $P_b(s)$ ] is given by the union of the pentagons, squares, and triangles in Fig. 4.1. Some of the possible roots are in the right half plane, so this polynomial family is not robustly stable. It can be seen from this latter example why robust stability is a worst-case condition. The majority, 2 out of 3, of the polynomials in  $P_b(s)$  are stable, but as a whole, the family is not robustly stable. The underlying consideration is that any chance of instability is unacceptable.  $\square$

The two examples above were generated for instructional purposes. This chapter shall now work towards a stability analysis of a physically motivated polynomial family example.

*Example 4.3.* Recall the linearized model of the crane given in Chapter 1. If a state feedback control

$$u(t) = [ k_1 \ k_2 \ k_3 \ k_4 ] \mathbf{x}(t) \quad (4.1.11)$$

is applied, then the closed-loop characteristic polynomial is given by (2.2.18)

$$p(s, g, \ell, m_L, m_C, k_1, k_2, k_3, k_4) = p(s, \mathbf{q}, \mathbf{k})$$

$$= \frac{gk_1}{\ell m_C} + \frac{gk_2}{\ell m_C} s + \frac{-k_3 + k_1\ell + gm_C + gm_L}{\ell m_C} s^2 + \frac{-k_4 + k_2\ell}{\ell m_C} s^3 + s^4 \quad (4.1.12)$$

This is just a symbolic representation of the polynomial. To get the actual polynomial, numerical values must be substituted for each of the parameters and feedback gains. However, not all of these parameters are known. As a result, the closed-loop characteristic polynomial is uncertain. To construct the set of possible polynomials, our knowledge about the parameters must be investigated. First, the parameter  $g$  equals gravitational acceleration which is a known constant.

$$g = 10 \text{ [ms}^{-2}\text{]}$$

Next, it is assumed that the feedback gains take the following values.

$$[ k_1 \ k_2 \ k_3 \ k_4 ] = [ 600 \ 2000 \ -10000 \ 0 ]$$

This leaves the physical parameters of the system. For now, it will be assumed that the crab mass was accurately measured

$$m_C = 1000 \text{ [kg]}$$

and that the loading bridge is operated using a constant, known rope length

$$\ell = 10 \text{ [m]}$$

Using this information, a simplified symbolic representation of the characteristic polynomial can be obtained.

$$p(s, m_L) = 0.6 + 2s + (2.6 + 0.001m_L)s^2 + 2s^3 + s^4 \quad (4.1.13)$$

The remaining parameter, load mass, is not known exactly because the crane picks up a variety of items having a variety of weights. All that is known is that the weights are in the range

$$m_L \in [50; 2395] \text{ [kg]} \quad (4.1.14)$$

From this knowledge of the parameters, it is seen that the closed-loop characteristic polynomial could be any element in the set

$$P(s, Q) = \left\{ 0.6 + 2s + 2.6s^2 + 2s^3 + s^4 + \frac{m_L}{1000}s^2 \mid m_L \in [50; 2395] \text{ [kg]} \right\} \quad (4.1.15)$$

Unlike the instructional examples which involved a finite number of possible polynomials, this physically motivated example contains an infinite number of possible polynomials. This is typical of examples motivated by parameter uncertainties.  $\square$

The fact that  $P(s, Q)$  usually contains an infinite number of elements complicates stability analysis by root set construction. It is clear how to compute Roots[ $P(s, Q)$ ] when  $P(s, Q)$  contains a finite number of polynomials. Taking one polynomial at a time, all the roots of each polynomial can be computed. Taking each polynomial one at a time is simply impossible if there are an infinite number of polynomials. In fact, except in very special cases, there is no feasible way of completely determining Roots[ $P(s, Q)$ ] when  $P(s, Q)$  is an infinite set. However, this theoretical limitation has not prevented engineers from obtaining the desired information via approximations of the root set.

In 1948, Evans [62] presented a method for root set construction that is known as the root locus method. This method essentially considers root set construction for a special case of (4.0.1–4.0.7) that is motivated as follows. Consider, a plant with scalar input  $u$  and scalar output  $y$  described by

$$y(s) = \frac{n(s)}{d(s)}u(s)$$

and subjected to constant gain output feedback

$$u(s) = -ky(s)$$

The root locus method shows how to find the roots of the closed-loop characteristic polynomial

$$p(s, k) = d(s) + kn(s) \quad (4.1.16)$$

as  $k$  ranges from a low value  $k^-$  (usually 0) to high value  $k^+$  (usually  $+\infty$ ), i.e.  $k \in K = [k^-; k^+]$ . This is equivalent to finding the root set of the infinite set of polynomials

$$P(s, K) = \{ d(s) + kn(s) \mid k^- \leq k \leq k^+ \} \quad (4.1.17)$$

generated by the *uncertain parameter*  $k$ . The set Roots[ $P(s, K)$ ], called a root locus, shows the variations of the closed-loop poles with respect to gain variations. Despite the infinite nature of the problem, the Evans root locus method shows how to obtain a useful approximation of Roots[ $P(s, K)$ ].

The traditional techniques of the Evans root locus method will not be discussed here. Instead, the approach typically used in a computer-based root locus construction shall be described. As mentioned previously, given a known polynomial  $p(s)$ , software is available to find the roots of  $p(s)$  with satisfactory accuracy. Using this basic root finding algorithm, the roots of the characteristic polynomial (4.1.16) can be found for any fixed value of  $k$ . The root locus is typically approximated by computing and plotting root values one polynomial at a time for several fixed values of  $k$ . The values chosen for  $k$  are referred to as a *grid*. One common method is to use an evenly spaced grid that ranges from  $k^-$  to  $k^+$ . For example, if  $k^- = 0$  and  $k^+ = 8$ , then a five point evenly space grid is the following set of values for  $k$ :

$$\{ 0, 2, 4, 6, 8 \}$$

To improve the accuracy of the root locus approximation, it is generally necessary to increase the number of grid points. For example, the nine point evenly spaced grid

$$\{ 0, 1, 2, 3, 4, 5, 6, 7, 8 \}$$

would give better accuracy than the five point grid above. The problem with using a larger grid is that more computations and more time are required. To save computational effort, the grid should be just large enough to get satisfactory accuracy. One of the best indicators of accuracy is continuity of the root locus. The roots of a polynomial

$$p(s) = a_0 + a_1 s + \dots + a_n s^n, \quad a_n > 0$$

depend continuously on the coefficients  $a_i$ . By assumption also the coefficient functions  $a_i(q)$  are continuous. Thus, we have the following theorem:

*Theorem 4.1.*

Let the set of polynomials  $P(s, Q)$  be generated as in (4.0.1–4.0.7) and for any  $\mathbf{q}^1, \mathbf{q}^2 \in Q$

$$a_0(\mathbf{q}^1) + a_1(\mathbf{q}^1)s + a_2(\mathbf{q}^1)s^2 + \dots + a_n(\mathbf{q}^1)s^n = a_n(\mathbf{q}^1)(s - \alpha_1)(s - \alpha_2) \cdots (s - \alpha_n)$$

$$a_0(\mathbf{q}^2) + a_1(\mathbf{q}^2)s + a_2(\mathbf{q}^2)s^2 + \dots + a_n(\mathbf{q}^2)s^n = a_n(\mathbf{q}^2)(s - \beta_1)(s - \beta_2) \cdots (s - \beta_n)$$

Then the roots  $\alpha_i$  may be so ordered relative to the roots  $\beta_i$  such that Roots[ $P(s, Q)$ ] contains at least one continuous path beginning at  $\alpha_i$  and ending at  $\beta_i$  for each  $i = 1, 2, \dots, n$ .

□

A proof of this well known continuity property is not given here, instead, the reader is referred to Marden [124]. Theorem 4.1 indicates that if an approximation of the root locus of (4.1.17) does not vary continuously from the roots of  $d(s) + k^-n(s)$  to the roots of  $d(s) + k^+n(s)$ , then the level of accuracy is not very high.

*Remark 4.2.* A reader with some previous knowledge of the root locus method or of root continuity properties may wonder how Theorem 4.1 treats the case of roots going to infinity. As an example of this case, suppose that  $d(s) + k^-n(s)$  has degree  $n$  for a finite  $k^-$  and  $d(s) + k^+n(s)$  has degree  $n - m$  for a finite  $k^+$ . In this case,  $m$  of the roots of  $d(s) + kn(s)$  would go off to infinity as  $k$  approaches  $k^+$ . Theorem 4.1 “treats” this case by not permitting it. Assumption (4.0.7), i.e.  $a_n(\mathbf{q}) > 0$  for all  $\mathbf{q} \in Q$ , prohibits any change of the polynomial’s degree. This assumption combined with the assumption of continuous parameter dependence (4.0.3) ensures that no roots go off to infinity.

A change of degree is qualitatively different than a change of parameter values. The reader who is interested in the case of degree changes is referred to the extension of Theorem 4.1 given by Zedek [182].  $\square$

*Example 4.4.* Using the root locus method, a stability analysis of the crane example can finally be carried out. By comparing equations (4.1.15) and (4.1.17), it can be seen that the characteristic polynomial of the crane has the same form as the root locus problem if the uncertain load mass  $m_L$  is treated as the gain  $k$ . A root locus for the crane with uncertain load mass was generated as follows. For each value of load mass on a 101 point evenly spaced grid, the roots of the polynomial (4.1.13) were computed and marked in the complex plane using a “+”. In addition, the roots for  $m_L$  equal to its minimum and maximum were marked using squares and triangles, respectively. This approximation of the root locus is shown in Fig. 4.2. The accuracy is quite good as can be seen by the continuous paths from squares to triangles. In fact, the accuracy is sufficient to conclude that the closed-loop characteristic polynomial of the crane is stable for all possible load masses (4.1.14).  $\square$

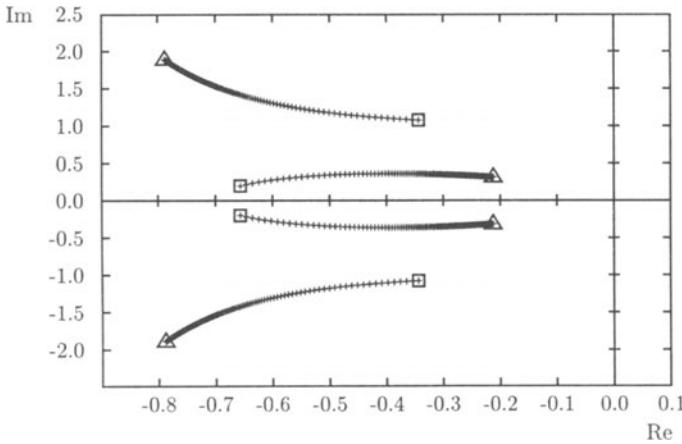


Fig. 4.2: Root locus of the crane with uncertain load mass

The root locus method can be generalized. In the traditional Evans root locus method, the set representing the uncertain polynomial is assumed to match the form of (4.1.17). The computer based approach of determining the polynomial’s roots for each parameter value on a chosen grid does not need to make this assumption. In fact, the method

can be generalized to allow a polynomial set  $P(s, Q)$  generated by continuous dependencies on several uncertain parameters as in (4.0.1–4.0.7). As in the one parameter case, the multiparameter root set Roots [ $P(s, Q)$ ] can be approximated using gridding. Normally, to form a grid of the set  $Q$ , the Cartesian products of independent grids of each interval  $[q_i^- ; q_i^+]$  are used. For example, suppose the number of parameters is two and that

$$q_1 \in [1 ; 4]$$

$$q_2 \in [6 ; 7]$$

Using 4 and 3 point evenly spaced grids of  $q_1$  and  $q_2$ , respectively, gives a 12 point grid of  $Q$ :

$$\{1, 2, 3, 4\} \times \{6, 6.5, 7\}$$

Once the grid is chosen, the multiparameter root set is approximated by computing the roots of each polynomial corresponding to a point on the grid.

*Example 4.5.* To illustrate a multiparameter root set, the controlled crane example (4.1.11–4.1.14) is used again. This time, however, the rope length is uncertain rather than known:

$$\ell \in [7 ; 12] \text{ [m]} \quad (4.1.18)$$

The corresponding set of characteristic polynomials is given by

$$m_L \in [50 ; 2395], \quad \ell \in [7 ; 12]$$

$$P(s, Q) = \left\{ \frac{6}{\ell} + \frac{20}{\ell}s + \frac{0.6\ell + 20 + 0.01m_L}{\ell}s^2 + 2s^3 + s^4 \mid [m_L \ \ell]^T \in Q \right\} \quad (4.1.19)$$

Using 101 and 11 point evenly spaced grids of  $m_L$  and  $\ell$ , respectively, the root set was computed and plotted in Fig. 4.3. The figure shows that the crane is stable for all admissible values of load mass (4.1.14) and rope length (4.1.18).  $\square$

The main difficulty in carrying out a stability analysis by root set construction is large computational time. An algorithm to find the roots of a single polynomial of reasonably low order is not particularly time consuming. Generally, repetition of the algorithm up to a few thousand times for a one or two parameter root set is acceptably brief. Now, consider the general case of  $\ell$  parameters. If an  $N$  point grid were used for each parameter, the root finding algorithm would need to be repeated  $N^\ell$  times. No matter how fast the computer is, it is easy to select fairly small values of  $\ell$  and  $N$  that would make the root computations last hours, weeks, even years. Motivated by these limits, the following sections look at alternate, potentially more efficient stability analysis methods.

## 4.2 Boundary Crossing

In the previous section, a stability analysis method for uncertain polynomials was presented. The method in that section required that all possible roots of the polynomial

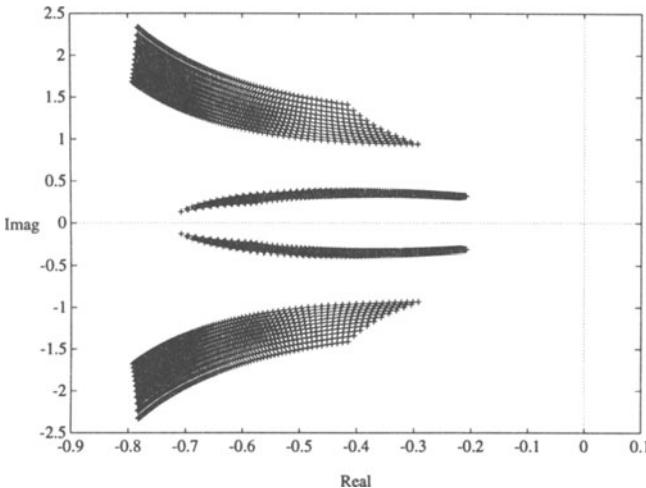


Fig. 4.3: Root set of the crane with uncertain load mass and rope length

be determined. This section shall show that determination of all possible root locations is not necessary. Based on the root continuity property, it is shown that the crucial root locations are those on the boundary of the stability region in  $s$ -plane. Later sections will exploit this fact in an attempt to provide computationally efficient alternatives to stability analysis by root set construction.

In 1929, Frazer and Duncan [69] investigated the effect of parameter variations on stability. Their work used the following important property of polynomials:

*Theorem 4.2.*

Let the set of polynomials  $P(s, Q)$  be generated as in (4.0.1–4.0.7). If there exist  $\mathbf{q}^1, \mathbf{q}^2 \in Q$  such that

$$p(s, \underline{\mathbf{q}}^1) = a_0(\mathbf{q}^1) + a_1(\mathbf{q}^1)s + a_2(\mathbf{q}^1)s^2 + \dots + a_n(\mathbf{q}^1)s^n$$

is stable and

$$p(s, \mathbf{q}^2) = a_0(\mathbf{q}^2) + a_1(\mathbf{q}^2)s + a_2(\mathbf{q}^2)s^2 + \dots + a_n(\mathbf{q}^2)s^n$$

is unstable then the root set of  $P(s, Q)$  contains at least one point on the nonnegative  $j\omega$ -axis, i.e. there exist  $\omega \geq 0$  such that

$$j\omega \in \text{Roots}[P(s, Q)]$$

□

*Proof.*

Theorem 4.1 implies that there exist factorizations

$$a_0(\mathbf{q}^1) + a_1(\mathbf{q}^1)s + a_2(\mathbf{q}^1)s^2 + \dots + a_n(\mathbf{q}^1)s^n = a_n(\mathbf{q}^1)(s - \alpha_1)(s - \alpha_2) \cdots (s - \alpha_n)$$

$$a_0(\mathbf{q}^2) + a_1(\mathbf{q}^2)s + a_2(\mathbf{q}^2)s^2 + \dots + a_n(\mathbf{q}^2)s^n = a_n(\mathbf{q}^2)(s - \beta_1)(s - \beta_2) \cdots (s - \beta_n)$$

such that the root set of Roots[ $P(s, Q)$ ] contains at least one continuous path beginning at  $\alpha_i$  and ending at  $\beta_i$  for all  $i = 1, 2, \dots, n$ . All  $\alpha_i$  are in the open left half plane and at least one  $\beta_i$  is in the closed right half plane. Assume without loss of generality that  $\beta_1$  is in the closed right half plane. The continuous root set path from  $\alpha_1$  to  $\beta_1$  must intersect the  $j\omega$ -axis. Hence, there exists  $\omega \in \mathbb{R}$  such that  $j\omega \in \text{Roots}[P(s, Q)]$ . Since  $P(s, Q)$  contains only real polynomials, it is also true that  $-j\omega \in \text{Roots}[P(s, Q)]$ . Since  $\omega \geq 0$  or  $-\omega \geq 0$ , this completes the proof.

□

Using this polynomial property, Frazer and Duncan provided the Boundary Crossing Theorem which is given next. This theorem follows easily from Theorem 4.2, so the proof is omitted.

*Theorem 4.3. (Boundary Crossing Theorem of Frazer and Duncan)*

Let  $P(s, Q)$  be a set of polynomials as in (4.0.1–4.0.7). The set  $P(s, Q)$  is robustly stable if and only if

- 1) there exists a stable polynomial  $p(s, \mathbf{q}) \in P(s, Q)$ ,
- 2)  $j\omega \notin \text{Roots}[P(s, Q)]$  for all  $\omega \geq 0$ .

□

*Remark 4.3.* A reader with some familiarity of the robust control literature may have seen a version of the Boundary Crossing Theorem with a third condition. This third condition guards against a stable root going into the right half plane by “sneaking” through the point at infinity. In this chapter this third condition is not needed because roots passing through infinity are not possible as discussed in Remark 4.2. □

The first condition of the Boundary Crossing Theorem is easy to check. One simply chooses any polynomial in  $P(s, Q)$  and tests stability of this single polynomial. If the chosen polynomial is not stable, then  $P(s, Q)$  is not robustly stable and the analysis is complete. If the chosen polynomial is stable, then to determine stability of the family, one must proceed to the second condition. Testing the second condition, i.e. searching for roots along the nonnegative  $j\omega$ -axis, is generally more efficient than finding all possible roots.

The Boundary Crossing Theorem is useful for explaining the validity of three commonly used robust stability analysis methods. The following sections of this chapter introduce these three methods. These approaches provide stability analysis tests that are generally more efficient than direct root set computation.

## 4.3 Algebraic Problem Formulation

This section looks at algebraic stability analysis methods. In Section 4.1, it was stated that the stability of a polynomial could be checked by using a computer to determine the polynomial's roots. Before the days of computers, the roots of a polynomial could not be easily found, but stability was of interest. Stability tests that could easily be carried out by hand computations were of great importance. These tests took the form of inequalities involving simple algebraic manipulations of a polynomial's coefficients. This section reviews the use of algebraic methods for testing the stability of a known polynomial. Generalizations of these methods for uncertain polynomials are also presented.

There are three main originators of algebraic stability tests. The first algebraic test was given by Hermite [82] in 1856. Partly due to its abstractness, Hermite's paper went unnoticed by most people interested in stability of engineering systems. In 1875, Routh [146] independently presented another algebraic stability test. Unaware of Routh's work, Hurwitz [90] removed the abstraction from Hermite's method and produced a more attractive stability test in 1895. Of the three methods, this section shall present only the Hurwitz method. This method admits the simplest generalization to uncertain polynomials.

Hurwitz has related the stability of an  $n$ -th order polynomial

$$p(s) = a_0 + a_1 s + a_2 s^2 + \cdots + a_n s^n, \quad a_n > 0, \quad (4.3.1)$$

to a set of determinants  $\Delta_i = \det \mathbf{H}_i$ . The determinants come from the following set of  $n$  Hurwitz matrices

$$\begin{aligned} \mathbf{H}_1 &= \left[ \begin{array}{c} a_{n-1} \end{array} \right] \\ \mathbf{H}_2 &= \left[ \begin{array}{cc} a_{n-1} & a_{n-3} \\ a_n & a_{n-2} \end{array} \right] \\ \mathbf{H}_3 &= \left[ \begin{array}{ccc} a_{n-1} & a_{n-3} & a_{n-5} \\ a_n & a_{n-2} & a_{n-4} \\ 0 & a_{n-1} & a_{n-3} \end{array} \right] \end{aligned} \quad (4.3.2)$$

This pattern continues until an  $n \times n$  matrix is obtained. For  $n$  even, this last matrix

$\mathbf{H}_n$  has the form

$$\mathbf{H}_n = \begin{bmatrix} a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_1 & 0 & 0 & 0 & \cdots & 0 \\ a_n & a_{n-2} & a_{n-4} & \cdots & a_2 & a_0 & 0 & 0 & \cdots & 0 \\ 0 & a_{n-1} & a_{n-3} & \cdots & a_3 & a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_n & a_{n-2} & \cdots & a_4 & a_2 & a_0 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_3 & a_1 & 0 \\ 0 & \cdots & 0 & a_n & a_{n-2} & a_{n-4} & \cdots & a_4 & a_2 & a_0 \end{bmatrix}$$

while for  $n$  odd, it has the form

$$\mathbf{H}_n = \begin{bmatrix} a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_0 & 0 & 0 & \cdots & 0 \\ a_n & a_{n-2} & a_{n-4} & \cdots & a_1 & 0 & 0 & \cdots & 0 \\ 0 & a_{n-1} & a_{n-3} & \cdots & a_2 & a_0 & 0 & \cdots & 0 \\ 0 & a_n & a_{n-2} & \cdots & a_3 & a_1 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & \vdots \\ 0 & \cdots & 0 & a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_1 & 0 \\ 0 & \cdots & 0 & 0 & a_n & a_{n-2} & \cdots & a_2 & a_0 \end{bmatrix}$$

As an example, if the polynomial is of order  $n = 4$ , the largest Hurwitz matrix is

$$\mathbf{H}_4 = \begin{bmatrix} a_3 & a_1 & 0 & 0 \\ a_4 & a_2 & a_0 & 0 \\ 0 & a_3 & a_1 & 0 \\ 0 & a_4 & a_2 & a_0 \end{bmatrix}$$

while for  $n = 5$ , the largest Hurwitz matrix is

$$\mathbf{H}_5 = \begin{bmatrix} a_4 & a_2 & a_0 & 0 & 0 \\ a_5 & a_3 & a_1 & 0 & 0 \\ 0 & a_4 & a_2 & a_0 & 0 \\ 0 & a_5 & a_3 & a_1 & 0 \\ 0 & 0 & a_4 & a_2 & a_0 \end{bmatrix}$$

The relation of these matrices to stability is given in the following theorem. No proof shall be provided for this well-known result of Hurwitz.

*Theorem 4.4. (Hurwitz)*

An  $n$ -th order polynomial (4.3.1) is stable if and only if

$$\det \mathbf{H}_i > 0 \quad (4.3.3)$$

for all  $i = 1, 2, \dots, n$ .

□

Note, that for both  $n$  even and  $n$  odd it follows from expanding  $\det \mathbf{H}_n$  by its last column that

$$\det \mathbf{H}_n = a_0 \det \mathbf{H}_{n-1} \quad (4.3.4)$$

Thus, the stability condition (4.3.3) is equivalent to  $\det \mathbf{H}_i > 0$  for  $i = 1, 2, \dots, n-1$ , and  $a_0 > 0$ .

Given a polynomial with known coefficients  $a_0, a_1, a_2, \dots, a_n$ , it is straightforward to apply the Hurwitz stability test. The resulting determinants are real numbers that can be easily compared with zero. If the coefficients depend on uncertain parameters,  $a_0(\mathbf{q}), a_1(\mathbf{q}), a_2(\mathbf{q}), \dots, a_n(\mathbf{q})$ , it is still possible to form the Hurwitz determinants symbolically.

*Example 4.6.* Consider the controlled crane with uncertain load mass and rope length (4.1.19). The four symbolically computed Hurwitz determinants are given below.

$$\det \mathbf{H}_1(\mathbf{q}) = \det \begin{bmatrix} 2 \end{bmatrix} = 2$$

$$\det \mathbf{H}_2(\mathbf{q}) = \det \begin{bmatrix} 2 & \frac{20}{\ell} \\ 1 & \frac{0.6\ell + 20 + 0.01m_L}{\ell} \end{bmatrix} = \frac{1000 + 60\ell + m_L}{50\ell}$$

$$\det \mathbf{H}_3(\mathbf{q}) = \det \begin{bmatrix} 2 & \frac{20}{\ell} & 0 \\ 1 & \frac{0.6\ell + 20 + 0.01m_L}{\ell} & \frac{6}{\ell} \\ 0 & 2 & \frac{20}{\ell} \end{bmatrix} = \frac{2000 + 2m_L}{5\ell^2}$$

$$\det \mathbf{H}_4(\mathbf{q}) = a_0 \det \mathbf{H}_3(\mathbf{q}) = \frac{6}{\ell} \det \mathbf{H}_3(\mathbf{q})$$

The symbolic Hurwitz determinants can be useful in testing the stability of an uncertain system. For example, it is easily seen that the preceding Hurwitz determinants of the crane are positive for all positive values of  $m_L$  and  $\ell$ . Since only positive parameter

values are physically possible, this implies that the crane is robustly stable. The crane example is deceptively simple.  $\square$

In general, it is a difficult problem to determine if a given Hurwitz determinant is positive for a specified range of parameter values. Given this difficulty, it is helpful to know that all but one of the determinants can basically be ignored. In other words, it is sufficient to use only one particular determinant. This single determinant result is due to Frazer and Duncan and is presented in the following alternate version of the Boundary Crossing Theorem.

*Theorem 4.5. (Frazer and Duncan)*

Let  $P(s, Q)$  be a set of polynomials  $P(s, Q)$  as in (4.0.1–4.0.7). The set  $P(s, Q)$  is robustly stable if and only if

- 1) there exists a stable polynomial  $p(s, \mathbf{q}) \in P(s, Q)$ ,
- 2)  $\det \mathbf{H}_n(\mathbf{q}) \neq 0$  for all  $\mathbf{q} \in Q$ .

$\square$

*Proof.*

Necessity of condition 1 is obvious, and necessity of condition 2 follows from the Hurwitz conditions for stability. To prove sufficiency, it shall be shown that satisfaction of condition 2 of this theorem implies satisfaction of condition 2 of the Boundary Crossing Theorem. Let

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + a_2(\mathbf{q})s^2 + a_3(\mathbf{q})s^3 + \dots + a_n(\mathbf{q})s^n \quad (4.3.5)$$

be an arbitrary polynomial in  $P(s, Q)$ . The polynomial may be written

$$p(s, \mathbf{q}) = p_{even}(s, \mathbf{q}) + p_{odd}(s, \mathbf{q}) \quad (4.3.6)$$

where

$$p_{even}(s, \mathbf{q}) = h(s^2, \mathbf{q}) = a_0(\mathbf{q}) + a_2(\mathbf{q})s^2 + a_4(\mathbf{q})s^4 + \dots \quad (4.3.7)$$

is the even part and

$$p_{odd}(s, \mathbf{q}) = sg(s^2, \mathbf{q}) = a_1(\mathbf{q})s + a_3(\mathbf{q})s^3 + a_5(\mathbf{q})s^5 + \dots \quad (4.3.8)$$

is the odd part of the polynomial. It will be shown first that condition 2 of Theorem 4.5 is identical to the condition, that  $p_{even}(s, \mathbf{q})$  and  $p_{odd}(s, \mathbf{q})$  do not have common roots. Note, that  $s = 0$  is a root of  $p_{odd}(s, \mathbf{q})$ , so  $p_{even}(s, \mathbf{q})$  and  $p_{odd}(s, \mathbf{q})$  have a common root (i.e. they both equal zero for the same value of  $s$ ) if and only if  $h(s^2, \mathbf{q})$  and  $s^2g(s^2, \mathbf{q})$  have a common root. A new complex variable  $v := s^2$  is introduced and  $h(s^2, \mathbf{q})$  and  $s^2g(s^2, \mathbf{q})$  have a common root if and only if the two polynomials of  $v$

$$\begin{aligned} h(v, \mathbf{q}) &= a_0(\mathbf{q}) + a_2(\mathbf{q})v + a_4(\mathbf{q})v^2 + \dots \\ vg(v, \mathbf{q}) &= a_1(\mathbf{q})v + a_3(\mathbf{q})v^2 + a_5(\mathbf{q})v^3 + \dots \end{aligned}$$

have a common root. These latter two polynomials share a root only if their resultant matrix has a zero determinant (see Appendix B). If  $n$  is even, then the  $n \times n$  resultant matrix is

$$\mathbf{R}(\mathbf{q}) = \begin{bmatrix} a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & a_{n-1} & a_{n-3} & \cdots & a_3 & a_1 & 0 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & a_{n-1} & a_{n-3} & a_{n-5} & \cdots & a_3 & a_1 & 0 \\ a_n & a_{n-2} & a_{n-4} & \cdots & a_2 & a_0 & 0 & 0 & \cdots & 0 \\ 0 & a_n & a_{n-2} & \cdots & a_4 & a_2 & a_0 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & a_n & a_{n-2} & a_{n-4} & \cdots & a_4 & a_2 & a_0 \end{bmatrix}$$

where the dependence of the coefficients on  $\mathbf{q}$  has been omitted for brevity. By interchanging rows, it is possible to transform  $\mathbf{R}(\mathbf{q})$  into the Hurwitz matrix  $\mathbf{H}_n(\mathbf{q})$ . (The relation of  $\mathbf{R}(\mathbf{q})$  and  $\mathbf{H}_n(\mathbf{q})$  holds both for  $n$  even and for  $n$  odd.) This relationship implies that

$$\det \mathbf{R}(\mathbf{q}) = \pm \det \mathbf{H}_n(\mathbf{q}). \quad (4.3.9)$$

By condition 2,  $\det \mathbf{H}_n(\mathbf{q}) \neq 0$ , so the determinant of the resultant matrix is also nonzero. From the arguments above, this implies that the even and the odd part of the polynomial do not equal zero for the same value of  $s$ .

Now let  $s = j\omega$ , then

$$p(j\omega, \mathbf{q}) = h(-\omega^2, \mathbf{q}) + j\omega g(-\omega^2, \mathbf{q}) \quad (4.3.10)$$

The real part  $h(-\omega^2, \mathbf{q})$  and the imaginary part  $j\omega g(-\omega^2, \mathbf{q})$  do not equal zero for the same value of  $\omega$ , therefore

$$p(j\omega, \mathbf{q}) \neq 0 \quad \text{for all real } \omega \quad (4.3.11)$$

This implies that no  $p(s) \in P(s, Q)$  has a root on the  $j\omega$ -axis. Condition 2 of the Boundary Crossing Theorem is satisfied and the proof is complete.

□

*Remark 4.4.* In the proof of Theorem 4.5, rather than use the resultant method as Frazer and Duncan did, some prefer to base the proof on Orlando's formula [136]. Given an  $n$ -th order polynomial in factored form

$$p(s) = a_n(s - \sigma_1 - j\omega_1)(s - \sigma_2 - j\omega_2) \cdots (s - \sigma_n - j\omega_n),$$

Orlando's formula relates  $\det \mathbf{H}_n$  to the polynomial's roots as follows:

$$\det \mathbf{H}_n = (-1)^{n(n+1)/2} a_n^n \prod_{i=1}^n (\sigma_i + j\omega_i) \prod_{j=1}^{n-1} \prod_{k=j+1}^n (\sigma_j + j\omega_j + \sigma_k + j\omega_k) \quad (4.3.12)$$

If the real polynomial (4.1.1) has a root on the positive  $j\omega$ -axis, then without loss of generality  $\sigma_1 = 0$ ,  $\sigma_2 = 0$ , and  $\omega_1 = -\omega_2$ . In this case, one of the product terms in Orlando's formula

$$(\sigma_1 + j\omega_1 + \sigma_2 + j\omega_2)$$

is zero and hence  $\det \mathbf{H}_n$  also equals zero. If the polynomial has a root at  $s = 0$ , then without loss of generality  $\sigma_1 = 0$  and  $\omega_1 = 0$ . In this case, the product term

$$(\sigma_1 + j\omega_1)$$

is zero and hence  $\det \mathbf{H}_n$  is equal to zero. These facts would provide an alternate proof of Theorem 4.5 again using the Boundary Crossing Theorem.  $\square$

Theorem 4.5 shows that it is sufficient to work with only one inequality rather than  $n$  such inequalities. It may seem counterproductive, but this one critical inequality can be split into two separate inequalities. Individually, the two separate inequalities are simpler than the single inequality, so for calculation this is a useful thing to do. The separation is based on (4.3.4)

$$\det \mathbf{H}_n = a_0 \det \mathbf{H}_{n-1} \quad (4.3.13)$$

This implies that

$$\det \mathbf{H}_n \neq 0 \quad (4.3.14)$$

if and only if both

$$a_0 \neq 0 \quad (4.3.15)$$

$$\det \mathbf{H}_{n-1} \neq 0 \quad (4.3.16)$$

For  $a_0 = 0$  the polynomial has a real root at  $s = 0$ . Therefore, we refer to  $a_0 = 0$  as the real root boundary.

An interesting and useful application of Theorem 4.5 has been given by Bialas [37]. This result gives a simple stability analysis method for families of polynomials generated by a linear dependence on a single uncertain parameter. The Bialas result is given in the following theorem.

*Theorem 4.6. (Bialas)*

Let  $\mathbf{H}_n^b$  and  $\mathbf{H}_n^c$  be the Hurwitz matrices of

$$p_b(s) = b_0 + b_1 s + b_2 s^2 + \cdots + b_n s^n, \quad b_n > 0,$$

$$p_c(s) = c_0 + c_1 s + c_2 s^2 + \cdots + c_n s^n, \quad c_n > 0,$$

respectively. The polynomial family

$$P(s, Q) = \{ (1 - q)p_b(s) + qp_c(s) \mid q \in [0; 1] \}$$

is stable if and only if

- 1)  $p_b(s)$  is stable
- 2) the matrix  $(\mathbf{H}_n^b)^{-1} \mathbf{H}_n^c$  has no nonpositive *real* eigenvalues.

$\square$

*Proof.*

The family of polynomials considered in this theorem is a special case of the family considered in Theorem 4.5. For this special family, this proof will show the relationship of the conditions of the two theorems. Polynomial  $p_b(s)$  is in  $P(s, Q)$ , so obviously, condition 1 of the present theorem is necessary. Furthermore, its satisfaction implies condition 1 of Theorem 4.5. Now assuming that condition 1 is indeed satisfied, the relationship between the two second conditions is investigated. For an arbitrary polynomial

$$p(s, q) = (1 - q)p_b(s) + qp_c(s),$$

in  $P(s, Q)$  it can be shown by straightforward algebraic manipulations that the polynomial's Hurwitz matrix satisfies the following relationship

$$\mathbf{H}_n(q) = (1 - q)\mathbf{H}_n^b + q\mathbf{H}_n^c.$$

Condition 1 and Theorem 4.4 imply that for  $q = 0$

$$\det \mathbf{H}_n(0) = \det \mathbf{H}_n^b \neq 0.$$

This implies that  $\mathbf{H}_n^b$  is invertible, so for arbitrary  $q \in (0, 1]$ , it can be written that

$$\begin{aligned} \det \mathbf{H}_n(q) &= \det \left( -q\mathbf{H}_n^b \left[ \left( \frac{1-q}{-q} \right) \mathbf{I} - (\mathbf{H}_n^b)^{-1}\mathbf{H}_n^c \right] \right) \\ &= (-q)^n \det \mathbf{H}_n^b \det \left( \left( \frac{1-q}{-q} \right) \mathbf{I} - (\mathbf{H}_n^b)^{-1}\mathbf{H}_n^c \right) \end{aligned}$$

This implies that, for any  $q \in (0, 1]$ ,  $\det \mathbf{H}_n(q) = 0$  if and only if

$$\det \left( \left( \frac{1-q}{-q} \right) \mathbf{I} - (\mathbf{H}_n^b)^{-1}\mathbf{H}_n^c \right) = 0 \quad (4.3.17)$$

Recall that by definition,  $s \in \mathbb{C}$  is an eigenvalue of a matrix  $\mathbf{A}$  if and only if  $\det(s\mathbf{I} - \mathbf{A}) = 0$ . This makes it clear that (4.3.17) is satisfied if and only if there is a  $q \in (0, 1]$  such that  $-(1-q)/q$  is an eigenvalue of  $(\mathbf{H}_n^b)^{-1}\mathbf{H}_n^c$ . Since  $-(1-q)/q$  is real and ranges from  $-\infty$  up to 0 as  $q$  ranges from 0 to 1, it follows that (4.3.17) is satisfied if and only if  $(\mathbf{H}_n^b)^{-1}\mathbf{H}_n^c$  has a nonpositive real eigenvalue. This argument has shown the equivalence of condition 2 of this theorem with condition 2 of Theorem 4.5 (when condition 1 is assumed to be satisfied). This equivalence completes the proof.  $\square$

The test of the second condition in Theorem 4.6 can be simplified. By (4.3.13) we have

$$\det \mathbf{H}_n(q) = a_0(q) \det \mathbf{H}_{n-1}(q)$$

where

$$\begin{aligned} a_0(q) &= (1-q)b_0 + qc_0 \\ \mathbf{H}_{n-1} &= (1-q)\mathbf{H}_{n-1}^b + q\mathbf{H}_{n-1}^c \end{aligned}$$

By condition 1  $p_b(s)$  is stable, thus  $b_0 > 0$  and  $c_0 = p_c(0) > 0$  must be required for  $a_0(q) > 0$ . For the second factor, the proof of Theorem 4.6 may be written with  $\mathbf{H}_n$  replaced by  $\mathbf{H}_{n-1}$  [13]. Thus, we have

*Theorem 4.7.*

- $P(s, Q) = \{(1-q)p_b(s) + qp_c(s) \mid q \in [0; 1]\}$  is stable if and only if
- 1)  $p_b(s)$  is stable
  - 2)  $p_c(0) > 0$
  - 3) the matrix  $(\mathbf{H}_{n-1}^b)^{-1}\mathbf{H}_{n-1}^c$  has no nonpositive *real* eigenvalues.

□

Before closing this section, one of the easiest to use stability results shall be presented. This result can tell you at a glance that a polynomial (polynomial family) is not stable (not robustly stable). Unfortunately, this simple result does not provide sufficient conditions.

*Theorem 4.8.*

An  $n$ -th order polynomial (4.3.1) is stable only if

$$a_i > 0$$

for all  $i = 0, 1, 2, \dots, n - 1$ .

□

*Proof.*

The proof of this theorem follows from the fact that (4.3.1) is a real polynomial and can be factored as

$$\begin{aligned} p(s) &= a_n \prod_{i=1}^m (s - \sigma_i - j\omega_i)(s - \sigma_i + j\omega_i) \prod_{k=2m+1}^n (s - \sigma_k) \\ &= a_n \prod_{i=1}^m (s^2 - 2\sigma_i s + \sigma_i^2 + \omega_i^2) \prod_{k=2m+1}^n (s - \sigma_k). \end{aligned}$$

where the  $\sigma$ 's and  $\omega$ 's are real numbers. If this polynomial is stable, then the  $\sigma$ 's are all negative. This implies that all the subpolynomials

$$\begin{aligned} s^2 + (-2\sigma_i)s + (\sigma_i^2 + \omega_i^2) \\ s + (-\sigma_k) \end{aligned}$$

have positive coefficients, and hence their product  $p(s)$  must have only positive coefficients.

□

Note, that for  $n = 1$  and  $n = 2$  the positivity of the coefficients is sufficient for stability.

*Example 4.7.* The positive coefficient criterion is obviously applicable to uncertain polynomials. Recall the characteristic polynomial of the crane under state feedback given in (4.1.12).

$$p(s, \mathbf{q}) = \frac{gk_1}{\ell m_C} + \frac{gk_2}{\ell m_C} s + \frac{-k_3 + k_1\ell + gm_C + gm_L}{\ell m_C} s^2 + \frac{-k_4 + k_2\ell}{\ell m_C} s^3 + s^4$$

Noting that masses and lengths and  $g$  are positive, the positive coefficient criteria gives the following necessary conditions for stability.

$$\begin{aligned} k_1 &> 0 \\ k_2 &> 0 \\ k_3 &< k_1\ell + gm_C + gm_L \\ k_4 &< k_2\ell \end{aligned}$$

These conditions for stability are useful and can be obtained quickly and easily, but it must be emphasised that they are not sufficient conditions. For example, if

$$\begin{aligned} k_1 &= m_C \\ k_2 &= m_C \\ k_3 &= gm_C + \ell m_C + gm_L - \ell \\ k_4 &= m_C\ell - m_C, \end{aligned}$$

then

$$p(s, \mathbf{q}) = \frac{g}{\ell} + \frac{g}{\ell}s + \frac{1}{m_C}s^2 + \frac{1}{\ell}s^3 + s^4$$

has positive coefficients for all lengths and masses. However, if

$$m_C > \frac{1}{g}$$

then, the second Hurwitz determinant

$$\det \mathbf{H}_2 = \frac{1 - gm_C}{\ell m_C}$$

is negative, so the polynomial would be unstable despite the positivity of its coefficients.  $\square$

*Remark 4.5.* The positive coefficient criteria of Theorem 4.8 can be used to reduce the number of determinant criteria in Theorem 4.4. This simplification is due to Liénard and Chipart [117], for a simpler proof see [72]. According to this result, necessary and sufficient conditions for a polynomial

$$p(s) = a_0 + a_1s + \dots + a_n s^n, \quad a_n > 0$$

to be stable can be given in one of the four following forms:

1.  $a_0 > 0, a_2 > 0, \dots; \Delta_1 > 0, \Delta_3 > 0, \dots$
2.  $a_0 > 0, a_2 > 0, \dots; \Delta_2 > 0, \Delta_4 > 0, \dots$
3.  $a_0 > 0, a_1 > 0, a_3 > 0, \dots; \Delta_1 > 0, \Delta_3 > 0, \dots$
4.  $a_0 > 0, a_1 > 0, a_3 > 0, \dots; \Delta_2 > 0, \Delta_4 > 0, \dots$

□

This section has presented the classical algebraic approach to stability analysis. The notion of symbolic computation of Hurwitz determinants was the key idea needed to extend these methods to uncertain polynomials. The Boundary Crossing Theorem was used to show that only one of the symbolic Hurwitz determinants was critical in a robustness analysis. The compactness of the robustness conditions, i.e. a single inequality, is a nice feature of this approach. For the special class of families generated by a single linear uncertainty, the Bialas result gave a straightforward method of checking this inequality. For general polynomial families, the effort required to test the single inequality can range from trivial to very difficult. So in some situations, the algebraic approach is of little value, and in other situations, this approach is extremely useful.

## 4.4 Singular Frequencies

The Boundary Crossing Theorem gives two conditions for the stability of a polynomial family. The first condition involves the relatively simple task of testing the stability of a single polynomial. The second condition

$$j\omega \notin \text{Roots}[P(s, Q)] \quad (4.4.1)$$

for all  $\omega \geq 0$  is more difficult. The difficulty is that it is not feasible to check this condition one frequency at a time for each and every  $\omega \geq 0$  because this would involve an infinite number of repetitions. Based on experience with other frequency domain analysis methods, one might conjecture that it is not necessary to check all  $\omega$  values. For example, it is common practice to construct Nyquist, Bode, and Mikhailov plots, using only a grid of  $\omega$  values. There are, however, cases when “singular frequencies” occur which require special attention in all frequency domain methods for robustness analysis. First, an example will be used to introduce the notion of singular frequencies.

*Example 4.8.* Recall the characteristic polynomial of the crane under state feedback (4.1.12). For this example, the effect of gains  $k_2$  and  $k_3$  on root locations will be investigated. The other parameters have fixed values with  $k_1 = 300$ ,  $k_4 = 0$ ,  $g = 10$ ,  $m_C = 100$ , and with  $m_L$  and  $\ell$  left in symbolic form for the present. It is of interest to have the real part of all the roots of the characteristic polynomial smaller

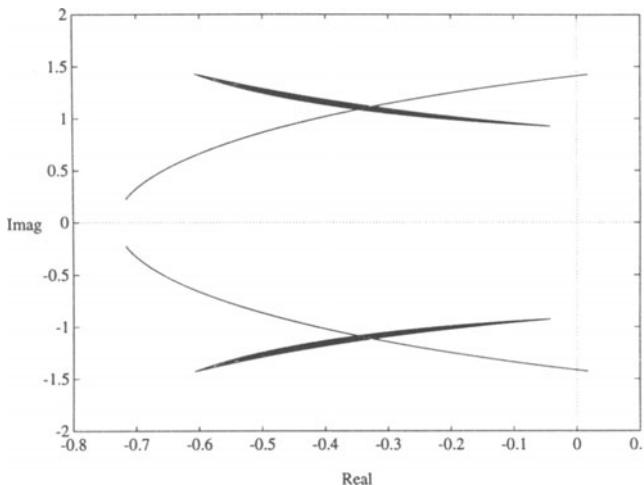


Fig. 4.4: Roots of the shifted characteristic polynomial of the crane as functions of  $k_2$  and  $k_3$

then  $-1$ . This is equivalent to having all the roots of the shifted polynomial (2.2.23) with  $a = 1$ , i.e.

$$\begin{aligned} p(s, k_2, k_3) = & (4000 - 10k_2 - k_3 + 400\ell - k_2\ell + 10m_L) + \\ & + (-2000 + 10k_2 + 2k_3 - 1000\ell + 3k_2\ell - 20m_L)s + \quad (4.4.2) \\ & + (1000 - k_3 + 900\ell - 3k_2\ell + 10m_L)s^2 + (-400\ell + k_2\ell)s^3 + 100\ell s^4 \end{aligned}$$

robustly stable. For this example, the root set of  $p(s, k_2, k_3)$  for

$$m_L = 1000$$

$$\ell = \frac{10}{3}$$

$$530 \leq k_2 \leq 540$$

$$7575 \leq k_3 \leq 7750$$

is shown in Fig. 4.4. For this range of gains, it can be seen that  $p(s, k_2, k_3)$  is not robustly stable. It can also be seen from the figure that the multiparameter root set crosses the imaginary axis of the  $s$ -plane on a very small range of frequencies in the vicinity of  $\omega = 1.4$ . In fact, it will be shown later in this chapter that the root set intersects the imaginary axis at just the one frequency  $\omega = \sqrt{2}$ .

This isolated  $j\omega$ -axis crossing has important implications for the use of frequency domain methods to test the stability of this uncertain polynomial family. Suppose a single test polynomial was chosen from the family and that this polynomial happened to be stable.

Since the first condition of the Boundary Crossing Theorem is not violated, the analysis would proceed to the second condition. Now suppose a large dense grid of frequencies was chosen, but this grid did not contain  $\omega = \sqrt{2}$ . Testing would show that the second condition (4.4.1) was not violated at any point on this dense grid. One might therefore be highly tempted to conclude that this polynomial family was robustly stable when in fact it is not robustly stable. To get the proper analysis conclusion,  $\omega = \sqrt{2}$  would have to be included in the grid. An isolated frequency of this type is an example for the occurrence of singular frequencies.  $\square$

In order to provide a systematic method of identifying singular frequencies, an algebraic definition shall be given. Before presenting this definition, some preliminary notation must be introduced. First, note that when an uncertain polynomial  $p(s, \mathbf{q})$  is evaluated at a frequency  $\omega$ , it equals a complex number  $p(j\omega, \mathbf{q}) = h(-\omega^2, \mathbf{q}) + j\omega g(-\omega^2, \mathbf{q})$ , see (4.3.10). The real and imaginary part of this complex number equal

$$h(-\omega^2, \mathbf{q}) = a_0(\mathbf{q}) - a_2(\mathbf{q})\omega^2 + a_4(\mathbf{q})\omega^4 - \dots \quad (4.4.3)$$

$$\omega g(-\omega^2, \mathbf{q}) = a_1(\mathbf{q})\omega - a_3(\mathbf{q})\omega^3 + a_5(\mathbf{q})\omega^5 - \dots \quad (4.4.4)$$

It was shown in the proof of Theorem 4.5 that

$$j\omega \in \text{Roots } [p(s, \mathbf{q})]$$

if and only if the two simultaneous real equations

$$\begin{aligned} h(-\omega^2, \mathbf{q}) &= 0 \\ \omega g(-\omega^2, \mathbf{q}) &= 0 \end{aligned} \quad (4.4.5)$$

are satisfied. These simultaneous equations can be thought of as a function that implicitly defines  $\omega$  given a vector  $\mathbf{q}$ . Associated with this implicit function is a Jacobian matrix

$$\mathbf{J}(\omega, \mathbf{q}) = \begin{bmatrix} \frac{\partial h(-\omega^2, \mathbf{q})}{\partial q_1} & \frac{\partial h(-\omega^2, \mathbf{q})}{\partial q_2} & \dots & \frac{\partial h(-\omega^2, \mathbf{q})}{\partial q_\ell} \\ \frac{\partial \omega g(-\omega^2, \mathbf{q})}{\partial q_1} & \frac{\partial \omega g(-\omega^2, \mathbf{q})}{\partial q_2} & \dots & \frac{\partial \omega g(-\omega^2, \mathbf{q})}{\partial q_\ell} \end{bmatrix}$$

The Jacobian  $\mathbf{J}(\omega, \mathbf{q})$ , the real part  $h(-\omega^2, \mathbf{q})$ , and the imaginary part  $\omega g(-\omega^2, \mathbf{q})$  are used to give an algebraic definition of singular frequencies.

*Definition 4.1.* The nonnegative frequency  $\omega_s$  is a *singular frequency* of the uncertain polynomial  $p(s, \mathbf{q})$  if there exists a  $\mathbf{q}^0 \in \mathbb{R}^\ell$  such that the three conditions

$$h(-\omega_s^2, \mathbf{q}^0) = 0$$

$$\omega_s g(-\omega_s^2, \mathbf{q}^0) = 0$$

$$\text{rank } [\mathbf{J}(\omega_s, \mathbf{q}^0)] < 2$$

are simultaneously satisfied.  $\square$

It is worth noting that, for any example, setting  $\omega = 0$  always makes the Jacobian matrix less than rank two and makes the imaginary part equal to zero. Hence, in any example,  $\omega = 0$  is a singular frequency (in general of the type isolated frequency) if there exists a  $\mathbf{q}$  that makes the real part equal to zero

$$h(0, \mathbf{q}) = a_0(\mathbf{q}) = 0$$

As might be expected, it is quite common for  $\omega = 0$  to be an isolated frequency. It describes the situation that a real root crosses the imaginary axis at  $s = 0$  under variation of  $\mathbf{q}$ .

The algebraic definition is useful for identifying the singular frequencies. For  $\ell$  uncertain parameters, the rank condition provides  $\ell - 1$  determinant equations, so the definition gives a total of  $\ell + 1$  equations. Finding the real solutions of these simultaneous equations (using, for example, the resultant method of Appendix B) will determine all possible singular frequencies.

The notion of singular frequencies is not standard in the literature, the concept has been used, however, by several authors, e.g. [154, 176, 56].

*Example 4.9.* This example will show how isolated frequencies can be identified in the case of affine coefficient functions. Again, consider the effect of gains  $k_2$  and  $k_3$  on the shifted characteristic polynomial of the crane (4.4.2). The first two conditions in the definition of singular frequencies are on the real part resp. on the imaginary part

$$\begin{aligned} h(-\omega^2, \mathbf{q}) &= 0 \\ \omega g(-\omega^2, \mathbf{q}) &= 0 \end{aligned}$$

These two terms become with  $\mathbf{q} = [k_2 \ k_3]^T$  the following

$$\begin{aligned} h(-\omega^2, \mathbf{q}) &= (4000 - 10k_2 - k_3 + 400\ell - k_2\ell + 10m_L) - \\ &\quad - (1000 - k_3 + 900\ell - 3k_2\ell + 10m_L)\omega^2 + 100\ell\omega^4 \end{aligned} \quad (4.4.6)$$

$$\begin{aligned} \omega g(-\omega^2, \mathbf{q}) &= (-2000 + 10k_2 + 2k_3 - 1000\ell + 3k_2\ell - 20m_L)\omega - \\ &\quad - (-400\ell + k_2\ell)\omega^3 \end{aligned} \quad (4.4.7)$$

Recalling that  $k_2$  and  $k_3$  are the uncertain parameters, the Jacobian of these two equations is given by

$$\mathbf{J}(\omega, \mathbf{q}) = \begin{bmatrix} -10 - \ell + 3\ell\omega^2 & -1 + \omega^2 \\ (10 + 3\ell)\omega - \ell\omega^3 & 2\omega \end{bmatrix} \quad (4.4.8)$$

By taking the determinant, it is seen that this matrix has rank less than two when

$$0 = \omega(1 + \omega^2)(-10 + \ell + \ell\omega^2)$$

is satisfied. Since this determinant equation does not depend on the uncertain parameters, all potential singular frequencies are given by its nonnegative real roots. There is definitely at least one such root

$$\omega_{s1} = 0$$

and if  $0 < \ell < 10$ , then there is a second nonnegative root.

$$\omega_{s2} = \sqrt{\frac{10 - \ell}{\ell}}$$

To see if  $\omega_{s1} = 0$  is actually a singular frequency, it must be determined if this frequency admits a solution to the conditions on the real and imaginary parts. Substituting  $\omega = \omega_{s1}$  into (4.4.6) and (4.4.7) produces the following simultaneous equations

$$(4000 - 10k_2 - k_3 + 400\ell - k_2\ell + 10m_L) = 0$$

$$0 = 0$$

For any  $\ell$ ,  $m_L$ , and  $k_2$ , these equations are solved using the following value of  $k_3$ .

$$k_3 = (4000 - 10k_2 + 400\ell - k_2\ell + 10m_L)$$

It can therefore be concluded that  $\omega_{s1} = 0$  is a singular frequency. Following the same procedure,  $\omega_{s2}$  is substituted into the real and imaginary parts (4.4.6) and (4.4.7) to determine if it is truly a singular frequency

$$(20 - 4\ell)k_2 + \left(-2 + \frac{10}{\ell}\right)k_3 + \left(-6000 + 1400\ell + 20m_L - \frac{100m_L}{\ell}\right) = 0$$

$$(4\ell)k_2 + (2)k_3 + (2000 - 1400\ell - 20m_L) = 0$$

Eliminating  $k_2$  from these simultaneous equations coincidentally eliminates  $k_3$  and leads to a single equation

$$3000 - \frac{10000}{\ell} = 0$$

This equation shows that  $\omega_{s2}$  can only be a singular frequency if

$$\ell = \frac{10}{3}$$

For this length and for any load mass  $m_L$ , the singular frequency is

$$\omega_{s2} = \sqrt{2}$$

This analysis supports the previous claim that the root set in Fig. 4.4 crosses the  $j\omega$ -axis only at  $\omega = \sqrt{2}$ .

In systems without uncertain parameters the Jacobian condition is trivially satisfied. Then the occurrence of a singular frequency as  $\omega_{s2}$  is the standard case, it is the frequency where the root locus crosses the imaginary axis and the Nyquist plot crosses the negative real axis. Under uncertainty the standard case is that this crossover frequency is dependent on  $\mathbf{q}$ . For the singular frequency  $\omega_{s2} = \sqrt{2}$ , however, the special case occurs that there exists a parameter independent crossover frequency.  $\square$

In the following examples we will treat the real root case  $\omega = 0$  separately and form the Jacobian of  $h(-\omega^2, \mathbf{q})$  and  $g(-\omega^2, \mathbf{q})$ .

In the case of affine coefficient functions, as in the above example, we have

$$\begin{aligned} h(-\omega^2, \mathbf{q}) &= (b_0 + \mathbf{c}_0^T \mathbf{q}) - (b_2 + \mathbf{c}_2^T \mathbf{q})\omega^2 + (b_4 + \mathbf{c}_4^T \mathbf{q})\omega^4 - \dots = 0 \\ g(-\omega^2, \mathbf{q}) &= (b_1 + \mathbf{c}_1^T \mathbf{q}) - (b_3 + \mathbf{c}_3^T \mathbf{q})\omega^2 + (b_5 + \mathbf{c}_5^T \mathbf{q})\omega^4 - \dots = 0 \end{aligned}$$

which may be written as

$$\begin{bmatrix} \mathbf{c}_0^T - \mathbf{c}_2^T \omega^2 + \mathbf{c}_4^T \omega^4 - \dots \\ \mathbf{c}_1^T - \mathbf{c}_3^T \omega^2 + \mathbf{c}_5^T \omega^4 - \dots \end{bmatrix} \mathbf{q} = \begin{bmatrix} -b_0 + b_2 \omega^2 - b_4 \omega^4 + \dots \\ -b_1 + b_3 \omega^2 - b_5 \omega^4 + \dots \end{bmatrix} \quad (4.4.9)$$

It is easily verified that the matrix on the left hand side of the above equation is identical to the Jacobian. Note, that it does not depend on  $\mathbf{q}$ .

$$\mathbf{J}(\omega) = \begin{bmatrix} \mathbf{c}_0^T - \mathbf{c}_2^T \omega^2 + \mathbf{c}_4^T \omega^4 - \dots \\ \mathbf{c}_1^T - \mathbf{c}_3^T \omega^2 + \mathbf{c}_5^T \omega^4 - \dots \end{bmatrix}$$

The condition  $\text{rank } \mathbf{J}(\omega_s) < 2$  is satisfied only for a finite number of frequencies  $\omega_i$ ,  $i = 1, 2, \dots, N$ . Geometrically, the simultaneous solutions of  $h(-\omega^2, \mathbf{q}) = 0$  and  $g(-\omega^2, \mathbf{q}) = 0$  at some fixed frequency  $\omega = \omega^*$  are intersections of two hyperplanes. At the frequency  $\omega^* = \omega_i$  the two hyperplanes are parallel. The frequency  $\omega_i$  is a singular frequency in the special case that the two hyperplanes are identical. Then the first and second row of (4.4.9) yield identical equations (except for a constant factor). In the affine case such isolated frequencies are the only manifestations of singular frequencies.

For nonlinear coefficient functions  $a_i(\mathbf{q})$  the Jacobian is a function of the uncertainty  $\mathbf{q}$ .

*Example 4.10.* Consider the bilinear polynomial family

$$P(s, q_1, q_2) = \{ p(s, q_1, q_2) \mid q_1 \in [0; 2], q_2 \in [0.25; 3] \} \quad (4.4.10)$$

generated by the uncertain polynomial

$$p(s, q_1, q_2) = 2 + r^2 + 6(q_1 + q_2) + 2q_1q_2 + (2 + q_1 + q_2)s + (2 + q_1 + q_2)s^2 + s^3$$

The interesting part of the root set for  $r = 0.5$  as obtained by gridding is shown in Fig. 4.5. The dashed lines are the images of the four edges of the  $Q$ -box.

The singular frequencies are determined by the simultaneous solutions of

$$\begin{aligned} h(-\omega^2, q_1, q_2) &= 2 + r^2 + 6(q_1 + q_2) + 2q_1q_2 - (2 + q_1 + q_2)\omega^2 = 0 \\ g(-\omega^2, q_1, q_2) &= 2 + q_1 + q_2 - \omega^2 = 0 \end{aligned}$$

$$\det \mathbf{J}(\omega, \mathbf{q}) = 0$$

where

$$\mathbf{J}(\omega, \mathbf{q}) = \begin{bmatrix} 6 + 2q_2 - \omega^2 & 6 + 2q_1 - \omega^2 \\ 1 & 1 \end{bmatrix} \text{ for } \omega \neq 0$$

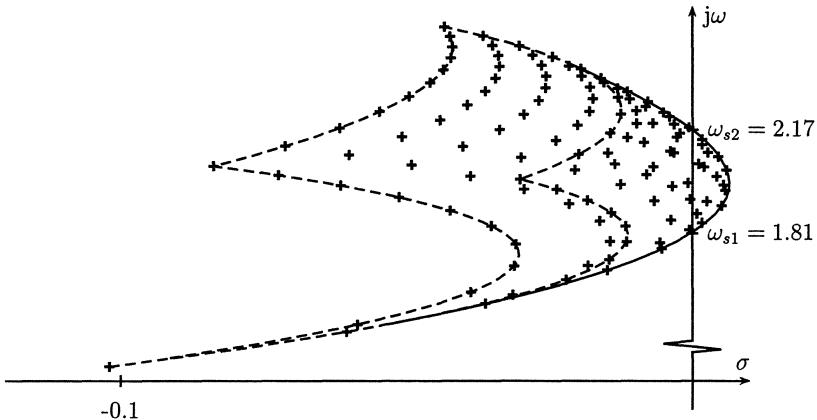


Fig. 4.5: Roots of the polynomial family (4.4.10)

i.e. the Jacobian condition  $\det \mathbf{J}(\omega, q) = 2(q_1 - q_2) = 0$  is satisfied for  $q = q_1 = q_2$ .

The equation  $g(-\omega^2, q) = 0$  yields  $\omega^2 = 2(1 + q)$  and with this substitution for  $\omega^2$  the first equation becomes

$$h(-\omega^2, q) = -2 + r^2 + 4q - 2q^2 = 0$$

The equation is satisfied for  $q = 1 \pm r/\sqrt{2}$  and the singular frequencies are  $\omega_{s1,2}^2 = 2(2 \pm r/\sqrt{2})$ .

For the value  $r = 0.5$  chosen in Fig. 4.5 the singular frequencies are  $\omega_{s1} = 1.81$ ,  $\omega_{s2} = 2.17$ . For  $r = 0$  the two singular frequencies become identical  $\omega_{s1} = \omega_{s2} = 2$ . In this case it is important again that the frequency  $\omega = 2$  is included in a grid on  $\omega$ .  $\square$

*Example 4.11.* Consider the interval polynomial family

$$P(s, q_1, q_2) = \{ p(s, q_1, q_2) = q_1 + q_2 s + s^2 \mid q_1 > 0, q_2 > 0 \} \quad (4.4.11)$$

Only the equation

$$h(-\omega^2, q_1) = q_1 - \omega^2 = 0$$

depends on the frequency and the entire imaginary axis  $\omega^2 = q_1$ ,  $q_1 > 0$  is covered by the singular frequencies.  $\square$

This section has introduced the notion of singular frequencies. It has also presented an algebraic procedure for determining these frequencies. In addition, this section has tried to motivate why singular frequencies can play important roles in robust stability analysis. This motivation will be reinforced by examples contained in the following two sections, and we will come back to singular frequencies in later chapters.

## 4.5 Parameter Space Problem Formulation

This section introduces the notion of a parameter space stability domain. Given an uncertain polynomial  $p(s, \mathbf{q})$ , the corresponding parameter space stability domain is defined to be the set of parameter vectors  $\mathbf{q}$  such that  $p(s, \mathbf{q})$  is stable. This set is denoted by  $Q_{stable}$ . The emphasis of the presentation is on efficiently obtaining a graphical description of  $Q_{stable}$ . The stability test for a plant family  $P(s, Q)$  is then to check if  $Q \subset Q_{stable}$ .

In 1876, Vishnegradsky [175] investigated the construction of  $Q_{stable}$  when  $p(s, \mathbf{q})$  is third order and only two uncertain parameters are present, i.e.  $\mathbf{q} = [q_1 \ q_2]^T$ . Vishnegradsky also introduced the idea of graphically representing  $Q_{stable}$ . Using a method based on the Boundary Crossing Theorem, Frazer and Duncan [69] showed how  $Q_{stable}$  might be graphically determined for the general case of an  $n$ -th order polynomial with two uncertain parameters. At the core of the Frazer-Duncan method is the need to find all  $\mathbf{q} = [q_1 \ q_2]^T$  that are roots of the symbolically computed  $n \times n$  Hurwitz determinant. The drawback of this method is that  $\det \mathbf{H}_n$  is normally a high order polynomial of  $q_1$  and  $q_2$  even for relatively simple uncertainties. In 1947 when Neimark [134] presented his *D-decomposition* method for construction of  $Q_{stable}$ , general computational tools for solving Frazer and Duncan's equations were not available. Neimark's method is also based on the Boundary Crossing Theorem, but it has computational advantages. If the uncertainties are of simple form, then the Neimark method finds  $Q_{stable}$  using straightforward computations even if  $p(s, \mathbf{q})$  is a high order polynomial. Further historical background and descriptions of more general uses of parameter space methods can be found in Šiljak [154]. A parameter space method has been introduced to the design of robust control systems by Ackermann [3]. This section will focus only on construction of  $Q_{stable}$ .

Using the Boundary Crossing Theorem we proceed in two steps. First, find the parameters  $Q_\omega$  that lead to a polynomial with roots on the imaginary axis. Second, test a nominal vector  $\mathbf{q}^0$ . If it is stable, then stability in the neighborhood of  $\mathbf{q}^0$  is preserved under all continuous variations of  $\mathbf{q}$  that do not intersect  $Q_\omega$ . In the first step we find

$$Q_\omega = \{ \mathbf{q} \mid p(j\omega, \mathbf{q}) = 0 \text{ for some } \omega \geq 0 \}$$

where  $p(j\omega, \mathbf{q}) = \operatorname{Re} p(j\omega, \mathbf{q}) + j \operatorname{Im} p(j\omega, \mathbf{q})$ .

$$\begin{aligned} \operatorname{Re} p(j\omega, \mathbf{q}) &= a_0(\mathbf{q}) - a_2(\mathbf{q})\omega^2 + a_4(\mathbf{q})\omega^4 - \dots \\ \operatorname{Im} p(j\omega, \mathbf{q}) &= \omega(a_1(\mathbf{q}) - a_3(\mathbf{q})\omega^2 + a_5(\mathbf{q})\omega^4 - \dots) \end{aligned}$$

The parameters  $\mathbf{q} \in Q_\omega$  simultaneously make the real and imaginary parts of  $p(j\omega, \mathbf{q})$  equal to zero.

For  $\omega \neq 0$  the parameters that yield  $p(j\omega, \mathbf{q}) = 0$  are the simultaneous solutions of

$$\begin{aligned} \operatorname{Re} p(j\omega, \mathbf{q}) &= a_0(\mathbf{q}) - a_2(\mathbf{q})\omega^2 + a_4(\mathbf{q})\omega^4 - \dots = 0 \\ \frac{1}{\omega} \operatorname{Im} p(j\omega, \mathbf{q}) &= a_1(\mathbf{q}) - a_3(\mathbf{q})\omega^2 + a_5(\mathbf{q})\omega^4 - \dots = 0 \end{aligned} \tag{4.5.1}$$

For  $\omega = 0$  the set  $Q_\omega$  consists of the solutions of

$$a_0(\mathbf{q}) = 0 \quad (4.5.2)$$

The assumption  $a_n(\mathbf{q}) > 0$  for all  $\mathbf{q} \in Q$  produces a third part of  $Q_\omega$  for  $\mathbf{q}$  such that

$$a_n(\mathbf{q}) = 0 \quad (4.5.3)$$

(As discussed in connection with Remark 4.1 we do not make the sophisticated distinction between a validity boundary and a stability boundary.)

The set  $Q_\omega$  contains the boundary of  $Q_{stable}$ , so once  $Q_\omega$  is found, most of the work in determining the stability domain is complete. The final step of finding the interior of the stability domain is easily carried out based on the first condition of the Boundary Crossing Theorem. Examples in this section will illustrate the complete construction of  $Q_{stable}$ .

Before giving examples, the process of solving the two simultaneous equations (4.5.1) shall be discussed. The systematic way to solve the two equations is to eliminate one of the variables either  $\omega^2$  or one of the parameters in  $\mathbf{q}$ . Elimination of  $\omega^2$  by the resultant method (see Appendix B) would essentially give Frazer and Duncan's method. Elimination of one of the parameters by the resultant or, if possible, by linear techniques, gives Neimark's method. The two equations (4.5.1) are solved for two of the uncertain parameters, say  $q_1$  and  $q_2$ . Thus,  $p(j\omega, \mathbf{q}) = 0$ ,  $\omega > 0$  is represented in the form

$$\begin{aligned} q_1 &= q_1(\omega, q_3, q_4, \dots, q_\ell) \\ q_2 &= q_2(\omega, q_3, q_4, \dots, q_\ell) \end{aligned} \quad (4.5.4)$$

For a grid on  $\omega, q_3, q_4, \dots, q_\ell$  the boundary points in the  $(q_1, q_2)$ -plane are determined and plotted. Some or all of the uncertain parameters  $q_i$  may be controller parameters as in the following example.

*Example 4.12.* Recall the characteristic polynomial of the crane under state feedback given in (4.4.2). For fixed values of  $m_L$  and  $\ell$ , the goal is to find all  $k_2$  and  $k_3$  such that all the roots of the characteristic polynomial have a real part smaller than  $-1$ . This is equivalent to finding all  $k_2$  and  $k_3$  such that the shifted polynomial  $p(s, k_2, k_3)$  from (4.4.2) is stable. To find these gains, the first step is to determine  $Q_\omega$  by solving the simultaneous equations (4.5.1). Since these equations are linear in  $k_2$  and  $k_3$ , they can be rewritten in matrix form

$$\mathbf{J}(\omega) \begin{bmatrix} k_2 \\ k_3 \end{bmatrix} = \begin{bmatrix} -4000 - 400\ell - 10m_L + (1000 + 900\ell + 10m_L)\omega^2 - 100\ell\omega^4 \\ (2000 + 1000\ell + 20m_L)\omega - 400\ell\omega^3 \end{bmatrix}$$

where  $\mathbf{J}(\omega)$  is the Jacobian matrix from (4.4.8). For all  $\omega > 0$  such that  $\det \mathbf{J}(\omega) \neq 0$ , the solution of these two equations can be written in closed form

$$\begin{bmatrix} k_2 \\ k_3 \end{bmatrix} = [\mathbf{J}(\omega)]^{-1} \begin{bmatrix} -4000 - 400\ell - 10m_L + (1000 + 900\ell + 10m_L)\omega^2 - 100\ell\omega^4 \\ (2000 + 1000\ell + 20m_L)\omega - 400\ell\omega^3 \end{bmatrix} \quad (4.5.5)$$

This equation shows that, in general, each value of  $\omega$  produces one value of  $\mathbf{q} = [k_2 \ k_3]^T$  and that this solution depends continuously on  $\omega$ .

The special values of  $\omega$  such that  $\det \mathbf{J}(\omega) = 0$  must be investigated separately. In Example 4.9, it was shown that the determinant of the Jacobian is nonzero for all positive frequencies if  $\ell \geq 10$ . If  $0 < \ell < 10$ , then the Jacobian is not invertible at a single positive frequency

$$\omega_{s2} = \sqrt{\frac{10 - \ell}{\ell}}$$

The analysis in Example 4.9 showed that there was no simultaneous solution of (4.4.6) and (4.4.7) at  $\omega_{s2}$  if  $\ell \neq 10/3$ . For the case when  $\ell = 10/3$ , it can be seen that the simultaneous solution at the isolated frequency  $\omega_{s2}$  is a line.

$$k_3 = -\frac{20}{3}k_2 + \left( \frac{4000 + 30m_L}{3} \right) \quad (4.5.6)$$

This shows that isolated frequencies can contribute infinite portions of  $Q_\omega$  and therefore contribute significantly to finding the stability domain.

The second step in the stability domain construction is to find all  $k_2$  and  $k_3$  that cause  $p(s, k_2, k_3)$  to have a root at the origin  $s = 0$ . These are determined by solving

$$p(0, k_2, k_3) = a_0(k_2, k_3) = 4000 - 10k_2 - k_3 + 400\ell - k_2\ell + 10m_L = 0$$

The solution set is easily seen to be the line described by the real root boundary

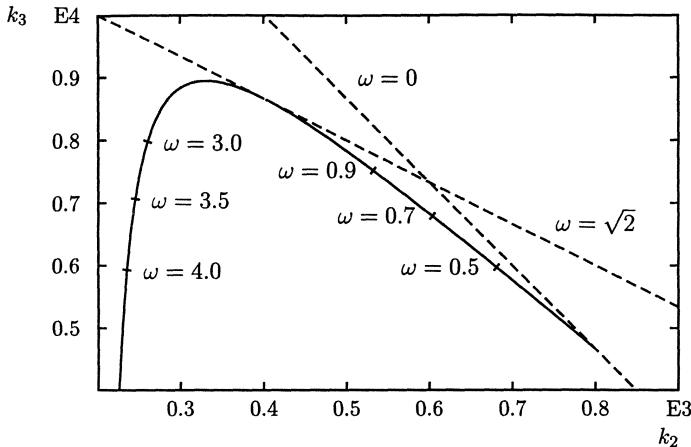
$$k_3 = (-10 - \ell)k_2 + (4000 + 400\ell + 10m_L) \quad (4.5.7)$$

Finally the equation  $a_n(\mathbf{q}) = 100\ell = 0$  yields the boundary  $\ell > 0$ .

For any fixed  $\ell > 0$  and  $m_L$ , equations to describe  $Q_\omega$  are now available. To graphically represent these sets, the values of  $\ell$  and  $m_L$  must be specified. For example, let  $\ell = 10/3$  and  $m_L = 1000$ . Substituting these values in (4.5.7) describes the real root boundary which is represented in Fig. 4.6 by the dashed line with the steepest slope. The portion of  $Q_\omega$  corresponding to nonsingular frequencies, described by (4.5.5), is solved explicitly by inverting  $\mathbf{J}(\omega)$ , i.e.

$$\begin{bmatrix} k_2 \\ k_3 \end{bmatrix} = \begin{bmatrix} \frac{200(4 + \omega^2)}{(1 + \omega^2)} \\ \frac{1000(14 + 34\omega^2 - \omega^4)}{3(1 + \omega^2)} \end{bmatrix}$$

Working with a grid of frequencies, this set was plotted in Fig. 4.6 using a solid curve. Some of the points on this curve are labeled with the frequencies that generated them. This complex root boundary branches off from the real root boundary at a parameter value yielding a double root at  $s = 0$ , i.e. a point that is common to the boundaries for a single root at  $s = 0$  and the boundary for a complex pair with frequency  $\omega = 0$ . The portion of  $Q_\omega$  due to the singular frequency  $\omega_{s2}$  is described by (4.5.6). This solution is represented by the dashed line with the more gradual slope in Fig. 4.6.

Fig. 4.6: Parameter space representation of polynomials with roots on  $j\omega$ -axis

The curves in Fig. 4.6 which represent  $Q_\omega$  divide the figure into six open connected regions. The Boundary Crossing Theorem indicates that if one of these regions contains a stable polynomial, then that region must contain only stable polynomials. Conversely, if one of these regions contains an unstable polynomial, then that region must contain only unstable polynomials. So, by picking one polynomial for each region and testing its stability, the set  $Q_{stable}$  can be completely described. The results show that  $Q_{stable}$  is the unshaded region in Fig. 4.7.  $\square$

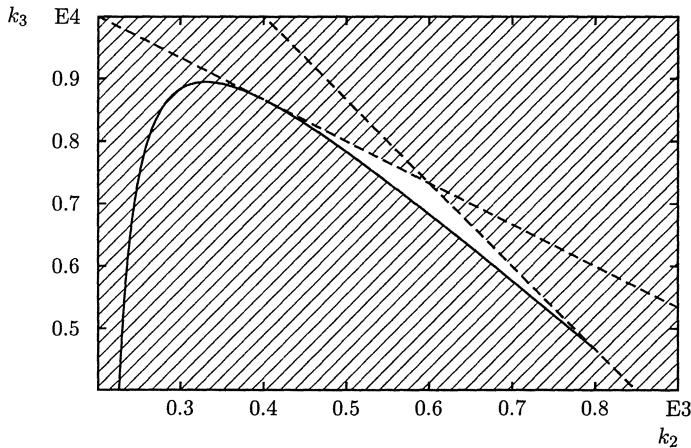


Fig. 4.7: Parameter space representation of stability region (unhatched)

*Example 4.13.* Consider the crane example above with gains in the following ranges.

$$530 \leq k_2 \leq 540$$

$$7575 \leq k_3 \leq 7750$$

Using a close up of Fig. 4.7, this set of gains is represented by the rectangular box in Fig. 4.8. Clearly, some of these gains in the upper right corner of this rectangle correspond to (4.4.2) being unstable. This is corroborated by Fig. 4.4 where the root set for these same values is shown. A major advantage of the parameter space analysis over the root set construction is that it would take very little effort to repeat this stability analysis for a different range of gains. In fact, the unhatched region in Fig. 4.7 is an exact description of the set of all stabilizing gains in this two-parameter case. Note, that “stability” refers to the shifted polynomial (2.2.23) with  $a = 1$ , for the crane this means that all eigenvalues have a real part smaller than  $-1$ .  $\square$

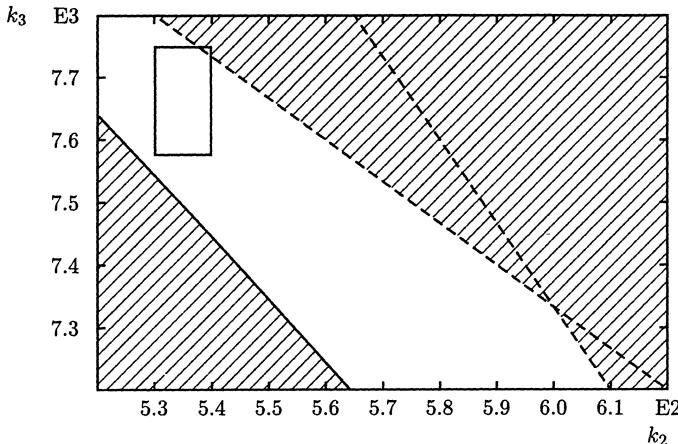


Fig. 4.8: A robust stability test in parameter space. The range of gains represented by the rectangular box is not robustly stable because part of the box intersects the instability region (hatched)

*Example 4.14.* For later comparisons and exercises with other methods we also introduce a simpler numerical example. Consider the uncertain polynomial

$$p(s, \mathbf{q}) = (14 - 0.3q_1 + 2q_2 + 2q_3) + (10 + 2q_1 + 8q_2)s + 10s^2 + 2(1 + q_1)s^3 + s^4$$

For plotting the root boundaries in the  $(q_1, q_2)$ -plane,  $q_3$  has to be fixed. For this affine case (4.5.1) reads

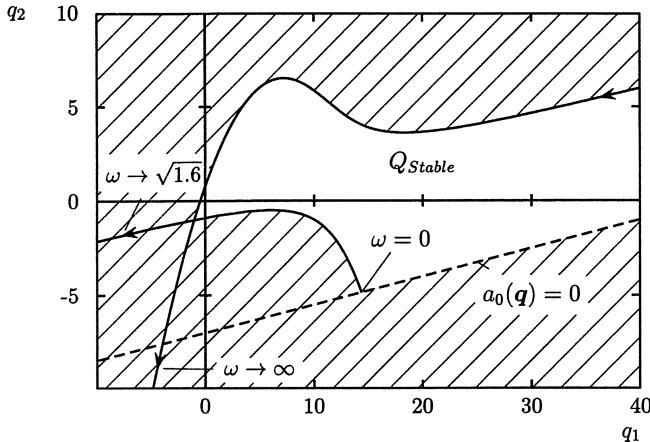
$$\mathbf{J}(\omega)\mathbf{q} = \mathbf{b}(\omega, q_3) \quad (4.5.8)$$

with  $\mathbf{q} = [q_1 \ q_2]^T$  and

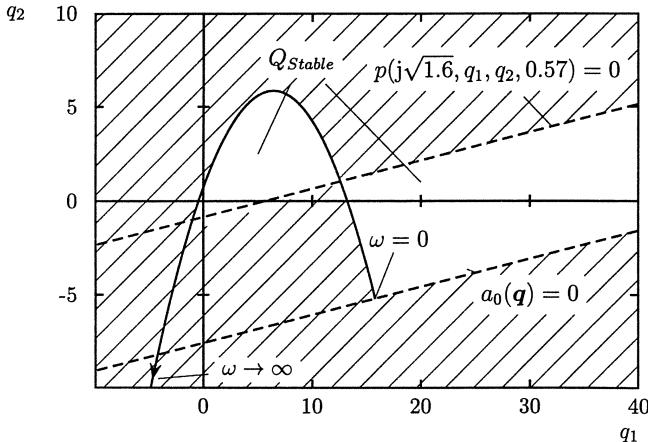
$$\mathbf{J}(\omega) = \begin{bmatrix} -0.3 & 2 \\ (\omega^2 - 1)2 & -8 \end{bmatrix}, \quad \mathbf{b}(\omega, q_3) = \begin{bmatrix} -\omega^4 + 10\omega^2 - 14 - 2q_3 \\ -2\omega^2 + 10 \end{bmatrix}$$

We have  $\det \mathbf{J}(\omega) = 0$  for  $\omega = \omega_1 = \sqrt{1.6}$ . For this frequency the first and second row of (4.5.8) yield

$$\begin{aligned} -0.3q_1 + 2q_2 &= -0.56 - 2q_3 \\ 1.2q_1 - 8q_2 &= 6.8 \end{aligned}$$

Fig. 4.9: The stability boundaries include an asymptote for  $\omega_1 = \sqrt{1.6}$ 

The equations describe two parallel straight lines. For  $q_3 \neq 0.57$  these lines are not identical and as  $\omega$  approaches  $\omega_1$  the intersection of the two lines goes to infinity, i.e. an asymptote occurs for  $\omega = \omega_1$ . For the special value  $q_3 = 0.57$  the two lines are identical and  $\omega_1$  is an isolated frequency.

Fig. 4.10: The stability boundaries include a isolated frequency at  $\omega_1 = \sqrt{1.6}$  for  $q_3 = 0.57$ 

The complex root boundary is obtained from  $\mathbf{q} = [q_1 \ q_2]^T = \mathbf{J}(\omega)^{-1}\mathbf{b}(\omega, q_3)$  with

$$\begin{aligned} q_1 &= (-10\omega^2 + 79)/5 \\ q_2 &= (-5\omega^4 + 47\omega^2 - 52)/10 \end{aligned}$$

and the real root boundary  $a_0(\mathbf{q}) = 0$  yields

$$-0.15q_1 + q_2 + 6.43 = 0$$

The condition  $a_n(\mathbf{q}) > 0$  is satisfied everywhere, i.e. there is no boundary  $a_n(\mathbf{q}) = 0$ . Fig. 4.9 shows the resulting boundary for  $q_3 = 0$  and Fig. 4.10 shows the case  $q_3 = 0.57$ . In the case  $q_3 = 0$  the stable region  $Q_{stable}$  is simply connected, in the case  $q_3 = 0.57$  the stable region  $Q_{stable}$  is disconnected.  $\square$

As the example above has shown, the parameter space approach to stability analysis can be very useful and has some advantages. The main drawback of the parameter space method is that it is best suited to uncertain polynomials with only two uncertain parameters. The method can be extended to polynomials with more than two parameters, but a considerable increase in effort is required. This extension is beyond the scope of this section.

## 4.6 Frequency Plot Problem Formulation

This section shall present a stability analysis method based on the graphical use of frequency plots. The complex number taken by a polynomial when evaluated at  $s = j\omega$  will be called its *value* at the frequency  $\omega$ . Computing and plotting the value of a polynomial for all nonnegative frequencies creates the polynomial's frequency plot. The frequency plot of a polynomial is referred to as a Mikhailov plot. Inspection of this plot can determine the stability or instability of the polynomial. This section will review the classical use of this method for known polynomials and present its extension for families of polynomials.

Stability conditions stated in terms of frequency plots have a long history. In their most general form, these conditions were given by Cauchy's principle of the argument around 1829. For the specific case of polynomials, conditions of this form also follow from the Hermite-Biehler Theorem [82, 38] (see also [72], p. 228). The graphical use of the argument principle was introduced to the engineering community by Nyquist [135] in 1932. In 1938, Mikhailov [129] gave simpler graphical conditions for analyzing stability of known polynomials. In 1944 and 1947, Leonhard [116] and Cremer [48], respectively, gave similar conditions. For this reason, the Mikhailov plot is sometimes called the Cremer-Leonhard plot. Extensions of this method from known polynomials to families of polynomials are also available. In the 1950s, a frequency plot method for families of polynomials was originated by Curtis. This method was made known to the control community by Zadeh and Desoer [181]. This approach has recently been labeled by Barmish [29] as the *value set* approach using the Zero Exclusion Theorem (Theorem 4.10). It is not fully clear which author should get the credit for this theorem. An assertion close to Theorem 4.10 is given in a book by Zadeh and Desoer [181]. Also Frazer and Duncan [69] were not far from this result. A historical review of the Zero Exclusion Theorem was given in a survey by Barmish [28]. Barmish also uses the theorem extensively in his forthcoming book on Robust Control [30].

This section will review both the Mikhailov stability conditions and the principles of the value set approach.

*Theorem 4.9. (Mikhailov, Leonhard and Cremer)*

The polynomial

$$p(s) = a_0 + a_1 s + \cdots + a_n s^n, \quad a_n > 0 \quad (4.6.1)$$

is stable if and only if the frequency plot  $p(j\omega)$ ,  $0 \leq \omega < \infty$  satisfies the following two conditions:

1.  $p(j0) = a_0 > 0$ , i.e. the plot starts on the positive real axis
2. As  $\omega$  increases, the plot of  $p(j\omega)$  encircles the origin in a counterclockwise direction and its phase goes to  $n\frac{\pi}{2}$  for  $\omega \rightarrow \infty$ .

□

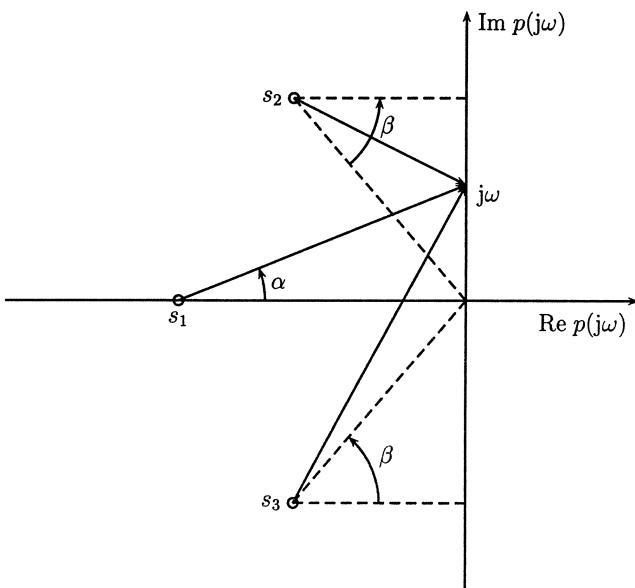


Fig. 4.11: The variation of  $\arg(j\omega - s_i)$

*Proof.*

To prove necessity we first assume that the polynomial is stable and show that the Mikhailov conditions must be satisfied. The first condition,  $a_0 > 0$ , is satisfied for a stable polynomial. The second condition follows from the argument principle. The stable polynomial  $p(s)$  can be factorized:

$$p(s) = a_n \prod_{i=1}^n (s - s_i) = a_n \prod_{i=1}^n \|j\omega - s_i\| e^{j\arg(j\omega - s_i)}$$

where all  $s_i$  have a negative real part. The argument of the product

$$p(j\omega) = a_n \prod_{i=1}^n (j\omega - s_i)$$

is the sum of the arguments of  $(j\omega - s_i)$ ,  $i = 1, 2, \dots, n$ . If a root is real, say  $s_1$ , then  $\alpha := \arg(j\omega - s_1)$  increases from zero to  $\frac{\pi}{2}$  as  $\omega$  goes from zero to  $\infty$  (see Fig.+4.11). If a root is complex, say  $s_2$ , then  $s_3 = \bar{s}_2$  is also a root of  $p(s)$ . The argument of  $(j\omega - s_2)$  changes from  $-\beta$  to  $\frac{\pi}{2}$ , the argument of  $(j\omega - s_3)$  from  $\beta$  to  $\frac{\pi}{2}$  as  $\omega$  goes from zero to  $\infty$ . So the total argument variation of a conjugate complex root pair is  $2\frac{\pi}{2}$ . Adding the contribution of each root the total argument variation of all roots is  $n\frac{\pi}{2}$ . Thus,  $p(j\omega)$  must encircle the origin corresponding to the second condition.

To prove sufficiency, assume that the Mikhailov conditions are satisfied. It must be shown that the polynomial is stable. We will show the equivalent statement: Whenever a polynomial is unstable it will violate the Mikhailov conditions. Assume that the polynomial has  $m$  roots in the right half plane and  $n - m$  roots in the left half plane. By the argument principle each root in the right half plane contributes a change of the argument of  $-\pi/2$  as  $\omega$  goes from zero to infinity. Thus, the total phase change is  $(n - 2m)\pi/2$  and the second Mikhailov condition is violated. For the special case of a root on the positive imaginary axis it has to be circumvented by a small semicircle into the left half plane in order to count the root correctly as unstable. Thus, the phase jumps by  $-\pi$  as  $\omega$  crosses the imaginary root (Fig. 4.12).

□

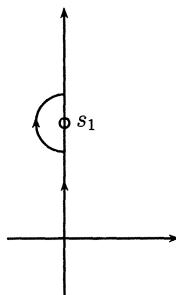


Fig. 4.12: The variation of  $\arg(j\omega - s_i)$  for unstable roots

Note, that the above proof also shows that the phase of the Mikhailov plot for a stable polynomial is monotonically increasing i.e.

$$\frac{\partial \arg\{p(j\omega)\}}{\partial \omega} > 0 \text{ for all } \omega \quad (4.6.2)$$

For every polynomial the total variation of the phase of  $p(j\omega)$  of  $(n - 2m)\pi/2$  determines the number  $n - m$  of left half plane roots and the number  $m$  of right half plane roots.

Note also, that for one or more unstable roots the phase of the Mikhailov plot cannot exceed  $(n - 1)\frac{\pi}{2}$ , i.e. once the Mikhailov plot has reached the  $n$ -th quadrant as  $\omega$  increases, stability is guaranteed. This condition gives an upper bound  $\omega^+$  for the frequencies  $\omega$  that must be tested, where

$$\arg p(j\omega^+) = (n - 1)\frac{\pi}{2} \quad (4.6.3)$$

The minimal distance between the Mikhailov plot of a stable polynomial and the origin is not an absolute measure for the stability margin because  $p(s)$  may be multiplied with an arbitrary factor without changing its roots. This distance is, however, a relative measure that indicates frequencies that are critical for stability. If the Mikhailov plot passes through the origin, then the polynomial has a root on the imaginary axis at the corresponding frequency.

The Mikhailov condition is often interpreted in another form. The phase condition is equivalent to requiring the frequency plot to intersect the axes in the following order: positive real (for  $\omega = 0$ ), positive imaginary, negative real, negative imaginary, positive real, . . . , until  $n$  such intersections have been made. This formulation is equivalent to the Hermite–Biehler Theorem [82, 38] that requires an “interlacing property” of the roots of  $\text{Re } p(j\omega) = 0$  and  $\text{Im } p(j\omega) = 0$ .

*Example 4.15.* Given the frequency plot of a polynomial, the Mikhailov conditions indicate at a glance whether or not the polynomial is stable. As examples, recall the shifted characteristic polynomial of the crane from (4.4.2), and let  $\ell = 10/3$  and  $m_L = 1000$ . For  $k_2 = 540$  and two different choices of the gain  $k_3$ , this shifted polynomial  $p(s, 540, k_3)$  is

$$p_1(s) = p(s, 540, 7575) = \frac{1675}{3} + \frac{1850}{3}s + 1025s^2 + \frac{1400}{3}s^3 + \frac{1000}{3}s^4 \quad (4.6.4)$$

and

$$p_2(s) = p(s, 540, 7750) = \frac{1150}{3} + \frac{2900}{3}s + 850s^2 + \frac{1400}{3}s^3 + \frac{1000}{3}s^4 \quad (4.6.5)$$

Bounded portions of the frequency plots  $p_1(j\omega)$  and  $p_2(j\omega)$  are shown in Fig. 4.13. Some points on each curve are labeled with the frequency that produced them. As always happens with nontrivial polynomials, the remainder of the two curves head off to infinity as  $\omega$  increases to  $+\infty$ . It is clear that the plot for the fourth order polynomial  $p_1(j\omega)$  meets the Mikhailov criteria, and hence,  $p(s, 540, 7575)$  is stable. It can be seen that the plot  $p_2(j\omega)$  intersects the axes in the order: positive real, positive imaginary, positive imaginary, positive real. The intersection order clearly violates the Mikhailov conditions, and therefore, the polynomial  $p(s, 540, 7750)$  is unstable. This analysis is corroborated by Fig. 4.8 where  $p(s, 540, 7575)$  and  $p(s, 540, 7750)$  correspond to the lower and upper right corners, respectively, of the parameter space rectangle.  $\square$

An obvious way to use the Mikhailov conditions for a family of polynomials is to compute the frequency plot of each polynomial in the family and to check the Mikhailov conditions one polynomial at a time. As was discussed in the root set section, the family normally has an infinite number of members, so this approach is not really possible. In

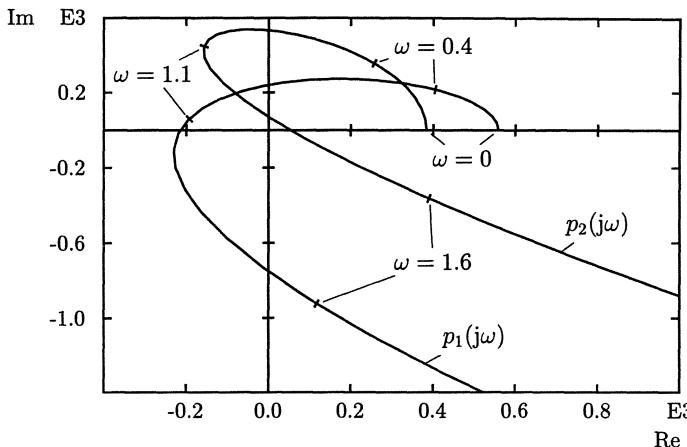


Fig. 4.13: Frequency plots of two polynomials

practical terms, however, computing the frequency plots for a sufficiently dense grid of the parameter set is adequate. To avoid large numbers of figures, the frequency plots are typically placed together on a single graph. This procedure is illustrated by two examples.

*Example 4.16.* Again, recall the shifted characteristic polynomial of the crane from (4.4.2) with  $\ell = 10/3$  and  $m_L = 1000$ .

$$\begin{aligned} p(s, k_2, k_3) = & \left( \frac{46000 - 40k_2 - 3k_3}{3} \right) + \left( \frac{-76000 + 60k_2 + 6k_3}{3} \right)s + \\ & + (14000 - 10k_2 - k_3)s^2 + \left( \frac{-4000 + 10k_2}{3} \right)s^3 + \frac{1000}{3}s^4 \end{aligned} \quad (4.6.6)$$

For the range of gains,

$$530 \leq k_2 \leq 540 \quad (4.6.7)$$

$$7575 \leq k_3 \leq 7750 \quad (4.6.8)$$

a 21 by 21 point grid was used to compute the set of all possible frequency plots. This collection of plots is shown in Fig. 4.14. For this range of gains, the polynomial is not robustly stable because Fig. 4.14 shows that at least one of the plots intersects the origin and hence violates the Mikhailov conditions.

This example was repeated using a smaller range of gains

$$530 \leq k_2 \leq 540 \quad (4.6.9)$$

$$7575 \leq k_3 \leq 7700 \quad (4.6.10)$$

The correspondingly smaller collection of frequency plots were computed and displayed in Fig. 4.15. Since the individual Mikhailov plots cannot be distinguished in the figure,

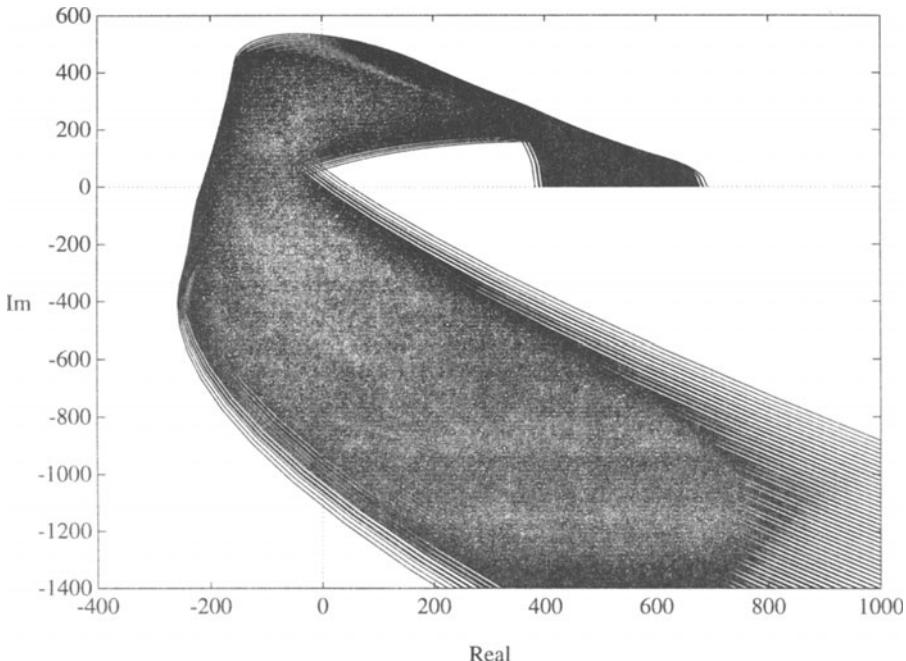


Fig. 4.14: Frequency plots of an unstable family of polynomials (The black surface in the figure results from very dense discrete lines)

it is not obvious whether or not the polynomial is robustly stable for this range of gains. The fact is that it is robustly stable, and the simplest way to prove this is to use the Boundary Crossing Theorem. The family defined by this range of gains is known to contain at least one stable polynomial (4.6.1). Now, a member of this family has a root on the  $j\omega$ -axis if and only if there exists a pair of gains  $k_2$  and  $k_3$  in the given ranges and a frequency  $\omega$  such that  $p(j\omega, k_2, k_3) = 0$ . Fig. 4.15 shows that the frequency plots do not intersect the origin and hence none of the complex numbers  $p(j\omega, k_2, k_3)$  equal zero. This implies there are no possible roots on the  $j\omega$ -axis, and hence by the Boundary Crossing Theorem, the polynomial is robustly stable for the given range of gains. The line of reasoning used in these two examples is true in general and will be stated in theorem form.  $\square$

*Theorem 4.10.*

Let  $P(s, Q)$  be a set of polynomials as given in (4.0.1–4.0.7). The set  $P(s, Q)$  is robustly stable if and only if

- 1) there exists a stable polynomial  $p(s, q) \in P(s, Q)$ ,

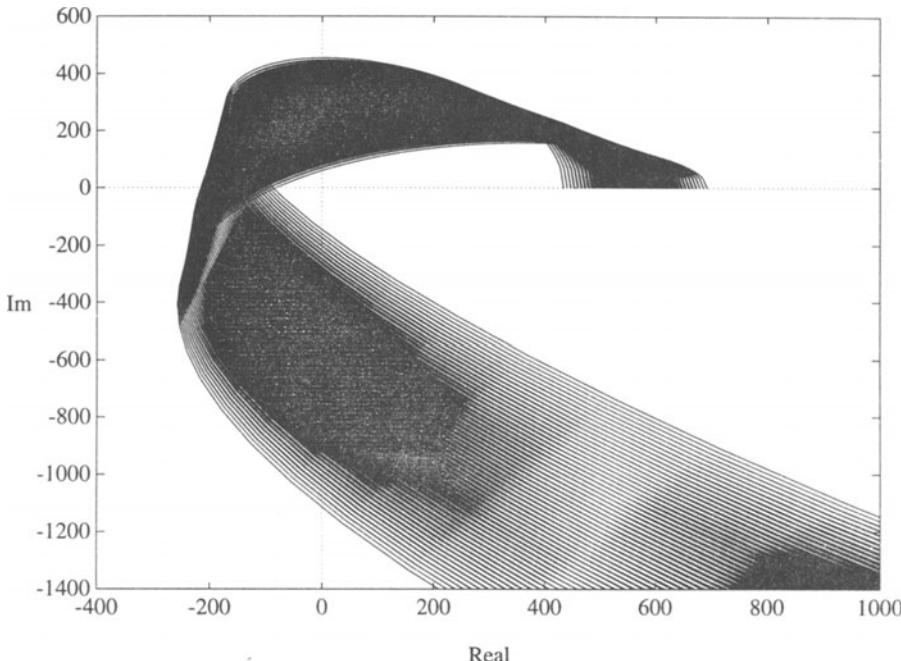


Fig. 4.15: Frequency plots of a stable family of polynomials

- 2) there does not exist a  $\mathbf{q} \in Q$  such that the frequency plot  $p(j\omega, \mathbf{q})$ ,  $\omega \geq 0$ , intersects the origin.

□

The gridding approach suggested in this section has one of the same drawbacks as computing the multiparameter root set. It is quite easy to select a grid that would cause the computations to take an excessively long time. For this reason, it is generally better to use a slightly different approach in computing the set of frequency plots.

Rather than to compute the frequency plot  $p(j\omega, \mathbf{q})$ ,  $\omega \geq 0$ , for each  $\mathbf{q}$  on a grid of  $Q$ , it is advisable to compute the *value set*

$$\mathcal{P}(j\omega, Q) = \{ p(j\omega, \mathbf{q}) \in \mathbb{C} \mid \mathbf{q} \in Q \}$$

for each  $\omega$  on a grid of frequencies from 0 to  $+\infty$ . This slight modification makes it possible to reap significant computational savings by exploiting the structure of the uncertain polynomial under investigation. Chapter 6 will cover this issue in detail. At this point, a value set of one of the previous examples is briefly investigated.

*Example 4.17.* Consider the polynomial (4.6.6). Because the gains enter the coefficients in an affine manner, value sets of this polynomial are always four sided convex polygons. In addition, the vertices of these polygons are determined by the extreme values of the gains. This are the four complex numbers  $p(j\omega, k_2^-, k_3^-)$ ,  $p(j\omega, k_2^-, k_3^+)$ ,  $p(j\omega, k_2^+, k_3^-)$ , and  $p(j\omega, k_2^+, k_3^+)$ . From just these four gain combinations rather than from a large grid of gains, it is possible to completely determine the value set at each frequency. For  $\omega = 0.85$  and for the large range of gains (4.6.7) and (4.6.8), the value set is the polygon bounded by the solid line in Fig. 4.16. To test stability, construction of the value set could be repeated for a grid of frequencies to get the set of all possible frequency plots. The collection of value sets would indicate stability or instability as stated in the theorem below.  $\square$

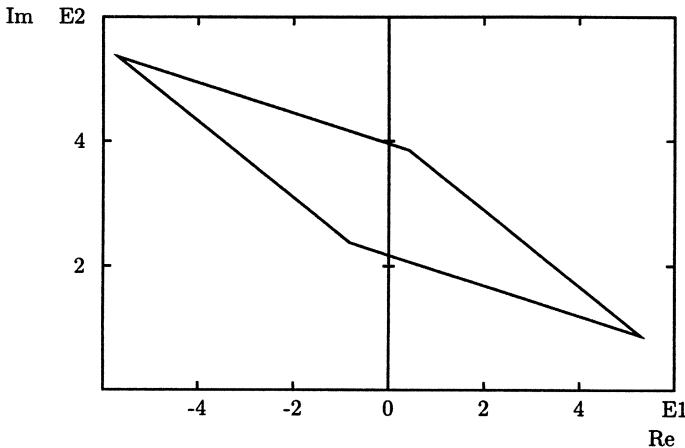


Fig. 4.16: A typical value set for an uncertain polynomial with affine parameter dependencies

*Theorem 4.11. (Zero Exclusion Theorem)*

Given a polynomial family  $P(s, Q)$  as in (4.0.1–4.0.7). The set  $P(s, Q)$  is robustly stable if and only if

- 1) there exists a stable polynomial  $p(s, q) \in P(s, Q)$ ,
- 2)  $0 \notin \mathcal{P}(j\omega, Q)$  for all  $\omega \geq 0$ .

$\square$

Since this theorem is a frequency domain application of the Boundary Crossing Theorem, it is subject to the effects of singular frequencies. In any use of this theorem to directly test the stability of a polynomial family, it would be wise to identify the singular frequencies. These singular frequencies should then be included in the grid of frequencies used in testing stability.

### *Summary*

This chapter has presented four different methods for testing the stability of a polynomial family. The root set, algebraic, and two frequency plot approaches are all formulated to handle the general class of families (4.0.1–4.0.7) given in the introduction. The parameter space approach also handles the general class of families but can only effectively treat two uncertain parameters at a time. Each of the four methods has its relative advantages and disadvantages. Some qualities of each method shall be briefly reviewed.

Robust stability analysis by root set construction was given first. In practice, the root set can only be approximated by use of a parameter set grid. An approximation does not contain all roots in the root set, but every root in an approximation is in the root set. For this reason, the approximation may clearly indicate instability even when the level of accuracy is quite poor. This fact and immunity to the effects of singular frequencies are strengths of this method. To conclude that a system is robustly stable, the root set approximation normally needs to be fairly accurate. Unfortunately, no exact rules for obtaining a desired level of accuracy are available. What is known is that a tradeoff between accuracy and computational time exists. Often a parameter grid that appears reasonable from an accuracy point of view will lead to intolerably long computational times. The possibility of excessive computations is the number one drawback of the root set approach.

The algebraic approach to robust stability analysis is the second method that was given. An analysis using this approach hinges on showing that a single determinant is always nonzero. The Bialas test used this condition to provide a simple and complete stability test for any polynomial with a single affine uncertainty. For more general classes of uncertainties, the ability to use the determinant condition will vary from example to example. In some cases, the condition will clearly indicate stability or instability. In other cases, the determinant condition will be too complicated to work with. A major drawback of this approach is that excessive complexity may occur even when the uncertainties are relatively simple in form.

The third stability analysis approach given in this chapter was the parameter space method. When the parameter uncertainties are not very complicated, this approach is relatively easy to use. The main difficulty is the need to deal with singular frequencies. The graphical representation of the stability region provided by this method is very appealing and highly informative. Unfortunately, the general applicability of this attractive approach is severely limited by the restriction to two uncertain parameters.

The final stability analysis method given in this chapter was the frequency plot approach. Instead of gridding all uncertain parameters it suffices to grid the frequency and to check the value set of the polynomial for zero exclusion at all frequencies.

None of the stability analysis methods given in this chapter are perfect. All of the methods can require relatively large amounts of computational effort. The next chapter will look at classes of uncertain polynomials for which much of this effort can be avoided.

## 4.7 Exercises

- 4.1. Plot the root set of the uncertain polynomial in (4.1.13) for the parameter range

$$10 \text{ [kg]} \leq m_L \leq 5000 \text{ [kg]}$$

As a check, note that the computed root set must contain Fig. 4.2 as a subset.

- 4.2. Replot the root set of the uncertain polynomial in (4.1.19) in two parts. First, plot the root set for shorter rope lengths

$$7 \text{ [m]} \leq \ell \leq 9.5 \text{ [m]}$$

and then for longer rope lengths

$$9.5 \text{ [m]} \leq \ell \leq 12 \text{ [m]}$$

Note, that the union of the two root sets must equal Fig. 4.3. Determine which of the two root sets is farther from the  $j\omega$ -axis. What, if anything, does this relative distance indicate?

- 4.3. Using Hurwitz matrices, test the stability of each of the three polynomials (4.1.7). Do the results agree with those indicated by Fig. 4.1?
- 4.4. Consider the uncertain shifted polynomial in 4.4.2. For  $m_L = 1000$ ,  $\ell = 10/3$ ,  $k_2 = 560$ , and  $k_3 = 7500$ , determine the stability of this polynomial using Hurwitz determinants. Does this result agree with that found by checking the point  $k_2 = 560$  and  $k_3 = 7500$  against the parameter space stability region in Fig. 4.8? Now, with  $m_L$ ,  $\ell$ ,  $k_2$ , and  $k_3$ , all unspecified, symbolically compute the determinant of the largest Hurwitz matrix. Noting that lengths and masses must be positive, does the symbolic determinant give any obvious indications of stability of the uncertain shifted polynomial? Setting  $m_L = 1000$ ,  $\ell = 10/3$ , and  $k_2 = 560$ , use the determinant to find all values of  $k_3$  around 560 that maintain stability. Do the results agree with Fig. 4.8?
- 4.5. Recall that the uncertain system in Example 4.12 had a nonzero singular frequency only for  $\ell = 10/3$ . For this length, the nonzero singular frequency contributed a large portion of the parameter space stability boundary shown in Fig. 4.7. For any infinitesimal change in  $\ell$  away from  $10/3$ , the contribution of the nonzero singular frequency will suddenly vanish. This suggests that the parameter space stability region might change dramatically for a tiny change in  $\ell$ . To check whether or not this actually happens, find the parameter space stability region for  $\ell$  slightly greater and slightly less than  $10/3$ . Did dramatic changes in the stability region occur?
- 4.6. Recall that the collection of Mikhailov plots for the unstable family of polynomials (4.6.6)–(4.6.8) is shown in Fig. 4.14. This collection of Mikhailov plots intersects

the origin, so there must exist at least one frequency such that the boundary of the corresponding value set contains the origin. In an attempt to find such a frequency, plot several different value sets for this family of polynomials. If this search does not produce a value set whose boundary contains the origin, then try to estimate a range of frequencies in which a zero inclusion in the boundary is likely to occur. Watching the value set change as frequency changes will help in this range estimation.

- 4.7. Exercise 4.6 is a tricky problem to get exact results for. If finite precision numerical calculations are used, then an extremely tight range of frequencies can be found, but exact frequencies where the boundaries of the value set contain the origin can not be found. Since numerical calculations will not work, more analytical thought is required to find exact frequencies. To this end, recalling the warning in this chapter about frequency domain methods should give a useful hint on how one might find an exact frequency. Once the desired frequencies are found, plot the corresponding value sets. In finding the frequencies and in generating the plot, remember to carry the calculations as far as possible using infinite precision symbolic calculations.
- 4.8. Check the following uncertain polynomials for singular frequencies
  - a) The characteristic polynomial of the crane given in (2.2.18) with fixed parameters as in Example 4.8.
  - b) The characteristic polynomial of the car steering example given in (1.2.2) with the uncertain parameters virtual mass  $\tilde{m}$  and velocity  $v$ .
- 4.9. Consider the uncertain polynomial  $p(s, q) = 8.64 + 3s + 3s^2 + s^3 + q(8 + s + s^2)$ 
  - a) For what range of  $q$  is the polynomial stable?
  - b) For which frequencies is the stability boundary crossed?
- 4.10. Consider the crane with state feedback and the following parameters:  
 $m_L = 2, m_C = 2, g = 10, k_1 = 0.4, k_2 = 3, k_3 = -10, k_4 = 0.3$ .
  - a) For which rope lengths is the crane stable?
  - b) What frequencies occur at the stability boundary?

## 5 Testing Sets

In Chapter 4 several testing methods were presented for checking stability of a family of polynomials

$$P(s, Q) = \{p(s, \mathbf{q}) \mid \mathbf{q} \in Q\} \quad (5.0.1)$$

generated by an uncertain polynomial defined over  $Q$

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + \dots + a_{n-1}(\mathbf{q})s^{n-1} + a_n(\mathbf{q})s^n, \quad \mathbf{q} \in Q \quad (5.0.2)$$

As in Chapter 4, real and continuous coefficient functions  $a_i(\mathbf{q})$  are assumed with

$$a_n(\mathbf{q}) > 0 \quad (5.0.3)$$

for all  $\mathbf{q} \in Q$  where  $Q$  is a parameter box

$$Q = \{\mathbf{q} \mid q_i \in [q_i^-; q_i^+], \quad i = 1, \dots, \ell\} \quad (5.0.4)$$

The robustness tests considered so far were all based on the complete set of uncertain parameters, thus, the testing set for the robustness checks of  $p(s, \mathbf{q})$  over  $Q$  was  $Q$  itself. Here, the question arises whether it suffices to check only a subset of  $Q$ . Consider for example the polynomial

$$p(s, \mathbf{q}) = q_1 + q_2 s + s^2 \quad (5.0.5)$$

uncertain in two parameters  $q_1$  and  $q_2$  in a two-dimensional box

$$Q = \{\mathbf{q} \mid q_i \in [q_i^-; q_i^+], \quad i = 1, 2\} \quad (5.0.6)$$

There is a simple robustness result:  $P(s, Q)$  is stable for all  $\mathbf{q} = [q_1 \ q_2]^T$  in  $Q$  if and only if  $q_1 > 0$  and  $q_2 > 0$ . This is trivially true if and only if  $q_1^- > 0$  and  $q_2^- > 0$ . Thus, a simple robustness test for the uncertain second order polynomial is: the polynomial (5.0.5) is robustly stable over the box  $Q$  if and only if the vertex

$$\mathbf{q}^{--} := [q_1^- \ q_2^-]^T \quad (5.0.7)$$

of  $Q$  yields a stable polynomial (in short:  $\mathbf{q}^{--}$  is stable). Hence, to check stability, only a single point in the uncertainty box  $Q$  has to be checked. The set

$$Q_T := \{\mathbf{q}^{--}\} \subset Q \quad (5.0.8)$$

is a testing set for the robustness check of the uncertain polynomial (5.0.5) over  $Q$ . A general definition for a testing set will now be given:

*Definition 5.1.* A set  $Q_T \subset Q$  is called a *testing set* (of  $Q$ ) if the stability of the polynomial family  $P(s, Q_T)$  implies the stability of  $P(s, Q)$ .  $\square$

In the example above, obviously each subset of  $Q$  containing the point  $\mathbf{q}^{--}$  is a testing set. The testing set  $Q_T = \{\mathbf{q}^{--}\}$  is minimal in the sense that there exists no testing set with less elements.

The above example leads to the main question of the present chapter: For what properties of an uncertain polynomial  $p(s, \mathbf{q})$  over an  $\ell$ -dimensional box  $Q$  do there exist testing sets  $Q_T$  properly contained in the uncertainty box  $Q$ ? From a practical point of view testing sets are of interest which drastically reduce the number of elements compared to those of  $Q$ . For instance, classes of uncertain polynomials would be interesting where the testing set consists only of its edges or even is a finite set, for instance the vertices of the box. Still better would be a finite testing set independent of the number  $\ell$  of uncertain parameters.

In Section 5.1 it will be shown that indeed the special class of interval polynomials has a finite testing set of only four elements. Assuming polynomial coefficients depending affinely on the uncertain parameters leads in Section 5.2 to a testing set that is not finite. However, there exists a pretty small testing set that consists only of the edges of the parameter box  $Q$ , i.e. the testing set consists of a one-dimensional continuum. For general nonlinear polynomial coefficients such nice testing set results do not exist. It will be shown in Section 5.3 that if the polynomial coefficients depend multilinearly on the parameters  $q_i$ , then at least the convex hull of the value set can be constructed easily. In Section 5.4 it will be shown that the set of points in the operating domain for which a Jacobian determinant vanishes is (together with the edges of  $Q$ ) a testing set for polynomials  $p(s, \mathbf{q})$  with nonlinear dependency on  $\mathbf{q}$ . The Jacobian condition can be used for value set construction and algebraic stability tests.

## 5.1 Interval Polynomials: Kharitonov's Theorem

In this section we introduce the special class of interval polynomials for which a simple robust stability result exists.

*Example 5.1.* In Example 2.7 the characteristic polynomial (2.2.17) of the crane with state feedback was derived. If all the parameters except the load mass  $m_L$  are fixed, at the values of Example 4.3, then the uncertain polynomial is by (4.1.13)

$$p(s, m_L) = 0.6 + 2s + (2.6 + 0.001m_L)s^2 + 2s^3 + s^4 \quad (5.1.1)$$

The coefficient  $a_2$  depends on the load mass

$$m_L \in [50; 2395] [\text{kg}] \quad (5.1.2)$$

and the coefficient function

$$a_2(m_L) = 2.6 + 0.001m_L \quad (5.1.3)$$

assumes all values in the interval

$$a_2 \in [2.65 ; 4.995] \quad (5.1.4)$$

The polynomial coefficient  $a_2$  is uncertain in this interval. The other coefficients

$$a_0 = 0.6, a_1 = 2, a_3 = 2, a_4 = 1 \quad (5.1.5)$$

formally can be considered as coefficient functions that are constant with their values included in the intervals

$$a_0 \in [0.6 ; 0.6], \quad a_1 \in [2 ; 2], \quad a_3 \in [2 ; 2], \quad a_4 \in [1 ; 1] \quad (5.1.6)$$

The coefficient vector

$$\mathbf{a} = [a_0 \dots a_4]^T \quad (5.1.7)$$

of the characteristic polynomial (5.1.1) is now uncertain within the box

$$\begin{aligned} \mathcal{A} := \{ \mathbf{a} \mid & a_0 \in [0.6 ; 0.6], \quad a_1 \in [2 ; 2], \quad a_2 \in [2.65 ; 4.995], \\ & a_3 \in [2 ; 2], \quad a_4 \in [1 ; 1] \} \end{aligned} \quad (5.1.8)$$

We observe that the independent variable  $m_L$  appears just in one single coefficient function and therefore, any polynomial coefficient is independent from all other coefficients. Thus, the set

$$\mathcal{A}(m_L) := \{ \mathbf{a}(m_L) \mid m_L \in [50 ; 2395] \} \quad (5.1.9)$$

is the whole box  $\mathcal{A}$  of (5.1.8),

$$\mathcal{A}(m_L) = \mathcal{A} \quad (5.1.10)$$

The polynomial (5.1.1) is an example of an interval polynomial.  $\square$

*Example 5.2.* If in the characteristic polynomial (4.1.14) of Example 4.3 also the load mass is constant,  $m_L = 2395$  [kg], but the controller coefficient  $k_2$  is free in the interval

$$k_2 \in [1000 ; 3000] \quad (5.1.11)$$

then the closed-loop characteristic polynomial is

$$p(s, k_2) = 0.6 + 0.001k_2 s + 4.995s^2 + 0.001k_2 s^3 + s^4 \quad (5.1.12)$$

The independent variable  $k_2$  appears in two coefficient functions,  $a_1 = a_3 = 0.001k_2$ . Thus, the two coefficients  $a_1, a_3$  are not mutually independent. The range of the coefficient function  $\mathbf{a}(k_2) = [a_0(k_2) \ a_1(k_2) \ a_2(k_2) \ a_3(k_2) \ a_4(k_2)]^T$ , i.e.

$$\mathcal{A}(k_2) := \{ \mathbf{a}(k_2) \mid k_2 \in [1000 ; 3000] \} \quad (5.1.13)$$

is not a box. Thus, the polynomial (5.1.12) is not an interval polynomial.  $\square$

*Definition 5.2.* A polynomial

$$p(s, \mathbf{a}) = a_0 + a_1 s + \dots + a_n s^n \quad (5.1.14)$$

with the uncertain coefficient vector

$$\mathbf{a} := [a_0 \ a_1 \ \dots \ a_n]^T \quad (5.1.15)$$

is called an *interval polynomial*, if  $\mathbf{a}$  ranges over the box

$$\mathcal{A} := \{ \mathbf{a} \mid a_i \in [a_i^-; a_i^+], \ i = 0, 1, \dots, n \} \quad (5.1.16)$$

□

For an interval polynomial the uncertain parameters are identified with the polynomial coefficients  $a_i$ , that means  $a_i$  is not considered as a function of other parameters. Then each uncertain coefficient in (5.1.14) is independent of all other coefficients. An interval polynomial generates the polynomial family

$$P(s, \mathcal{A}) = \{ p(s, \mathbf{a}) \mid \mathbf{a} \in \mathcal{A} \} \quad (5.1.17)$$

In 1953 Faedo [63] formulated the problem of a necessary and sufficient condition for robust stability of an interval polynomial. Faedo obtained only a sufficient condition. In 1979, Kharitonov published a surprisingly simple necessary and sufficient stability condition for interval polynomials [106]. Kharitonov's result is that a testing set  $\mathcal{A}_T \subset \mathcal{A}$  exists which consists of only *four* points, that means it is independent on the number  $\ell$  of uncertain parameters. Thus, only *four* polynomials of the continuum  $\mathcal{A}$  have to be checked for robust stability.

*Theorem 5.1. (Kharitonov)*

The polynomial family

$$P(s, \mathcal{A}) = \{ p(s, \mathbf{a}) = a_0 + a_1 s + \dots + a_n s^n \mid \mathbf{a} \in \mathcal{A} \}, \ a_n > 0 \quad (5.1.18)$$

is stable if and only if the following four polynomials are stable:

$$\begin{aligned} p^{+-}(s) &= a_0^+ + a_1^- s + a_2^- s^2 + a_3^+ s^3 + a_4^+ s^4 + \dots \\ p^{++}(s) &= a_0^+ + a_1^+ s + a_2^- s^2 + a_3^- s^3 + a_4^+ s^4 + \dots \\ p^{-+}(s) &= a_0^- + a_1^+ s + a_2^+ s^2 + a_3^- s^3 + a_4^- s^4 + \dots \\ p^{--}(s) &= a_0^- + a_1^- s + a_2^+ s^2 + a_3^+ s^3 + a_4^- s^4 + \dots \end{aligned} \quad (5.1.19)$$

□

The polynomials (5.1.19) are called Kharitonov polynomials. The superscripts indicate the upper and lower bounds of the coefficients  $a_0$  and  $a_1$ .

*Remark 5.1.* A simple rule for constructing the Kharitonov polynomials is the “Kharitonov melody” ... two high, two low, two high, .... The upper and lower bounds occur in the four polynomials as

$$\begin{array}{ccccccccc} + & - & - & + & + & - & - & + \\ + & + & - & - & + & + & - & - \\ - & + & + & - & - & + & + & - \\ - & - & + & + & - & - & + & + \end{array}$$

□

The original proof of Theorem 5.1 by Kharitonov is complicated. Meanwhile, a much simpler proof was found [21], which is based on the zero exclusion from the value set of  $p(j\omega, \mathbf{a})$ . First, we prove the following lemma, [50]:

*Lemma.*

For each fixed  $\omega = \omega^* \geq 0$ , the value set  $P(j\omega^*, \mathcal{A}) = \{p(j\omega^*, \mathbf{a}) \mid \mathbf{a} \in \mathcal{A}\}$  is a rectangle with edges parallel to the coordinate axes and with vertices determined by the values of the four Kharitonov polynomials  $p^{+-}(j\omega^*)$ ,  $p^{++}(j\omega^*)$ ,  $p^{-+}(j\omega^*)$ ,  $p^{--}(j\omega^*)$ , see Fig. 5.1.

□

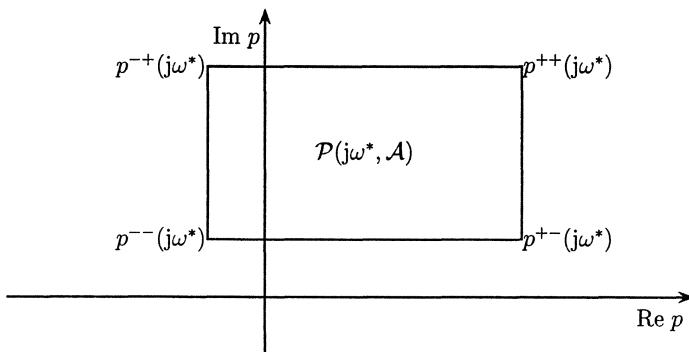


Fig. 5.1: Value set of an interval polynomial for fixed  $\omega = \omega^*$

*Proof.*

For all  $\omega \geq 0$  and all  $\mathbf{a} \in \mathbb{R}^{n+1}$  the real and imaginary part of  $p(j\omega, \mathbf{a}) = \text{Re } p(j\omega, \mathbf{a}) + j\text{Im } p(j\omega, \mathbf{a})$  have the lower and upper bounds

$$a_0^- - a_2^+ \omega^2 + a_4^- \omega^4 - \dots \leq \operatorname{Re} p(j\omega, \mathbf{a}) \leq a_0^+ - a_2^- \omega^2 + a_4^+ \omega^4 - \dots \quad (5.1.20)$$

$$\omega(a_1^- - a_3^+ \omega^2 + a_5^- \omega^4 - \dots) \leq \operatorname{Im} p(j\omega, \mathbf{a}) \leq \omega(a_1^+ - a_3^- \omega^2 + a_5^+ \omega^4 - \dots) \quad (5.1.21)$$

Since  $\operatorname{Re} p(j\omega, \mathbf{a})$  is a function of the even indexed parameters  $a_i$  and  $\operatorname{Im} p(j\omega, \mathbf{a})$  is a function of the odd indexed parameters  $a_i$ , the two upper bounds (the two lower bounds) are mutually independent. Since the two functions are continuous, the set  $P(j\omega^*, \mathcal{A})$  must be a rectangle. Its edges are parallel to the coordinate axes.

The four Kharitonov polynomials, defined in Theorem 5.1, can be represented as follows

$$\begin{aligned} p^{+-}(s) &= a_0^+ - a_2^- \omega^2 + a_4^+ \omega^4 - \dots + j\omega(a_1^- - a_3^+ \omega^2 + a_5^- \omega^4 - \dots) \\ p^{++}(s) &= a_0^+ - a_2^- \omega^2 + a_4^+ \omega^4 - \dots + j\omega(a_1^+ - a_3^- \omega^2 + a_5^+ \omega^4 - \dots) \\ p^{-+}(s) &= a_0^- - a_2^+ \omega^2 + a_4^- \omega^4 - \dots + j\omega(a_1^+ - a_3^- \omega^2 + a_5^+ \omega^4 - \dots) \\ p^{--}(s) &= a_0^- - a_2^+ \omega^2 + a_4^- \omega^4 - \dots + j\omega(a_1^- - a_3^+ \omega^2 + a_5^- \omega^4 - \dots) \end{aligned} \quad (5.1.22)$$

Obviously for each  $\omega \geq 0$  the vertices of the rectangle  $P(j\omega, \mathcal{A})$  are the values of the four Kharitonov polynomials.

□

Using the lemma we now prove Kharitonov's theorem.

*Proof.*

The necessary part of the theorem is trivial. To prove sufficiency assume that the four Kharitonov polynomials are stable. Then the Mikhailov plot (see Section 4.6) of each Kharitonov polynomial starts on the positive real axis and circles the origin in counterclockwise direction until its phase is  $n\pi/2$ . Since the edges of the value set are parallel to the coordinate axis and their end points satisfy the Mikhailov stability condition, the origin cannot enter into the value set through an edge of the value set. Hence, all Mikhailov curves for  $p(j\omega, a)$  with  $a \in \mathcal{A}$  are stable and  $P(s, \mathcal{A})$  is stable.

□

In accordance with our earlier Definition 5.1 of a testing set  $\mathcal{A}_T$  (of  $\mathcal{A}$ ), we have for an interval polynomial the following testing set containing only four elements:

$$\mathcal{A}_T = \{\mathbf{a}^{+-}, \mathbf{a}^{++}, \mathbf{a}^{-+}, \mathbf{a}^{--}\} \quad (5.1.23)$$

with

$$\begin{aligned} \mathbf{a}^{+-} &:= [a_0^+ \ a_1^- \ a_2^- \ a_3^+ \ a_4^+ \ \dots]^T \\ \mathbf{a}^{++} &:= [a_0^+ \ a_1^+ \ a_2^- \ a_3^- \ a_4^+ \ \dots]^T \\ \mathbf{a}^{-+} &:= [a_0^- \ a_1^+ \ a_2^+ \ a_3^- \ a_4^- \ \dots]^T \\ \mathbf{a}^{--} &:= [a_0^- \ a_1^- \ a_2^+ \ a_3^+ \ a_4^- \ \dots]^T \end{aligned} \quad (5.1.24)$$

We ask now if there are testing sets with less elements. The next theorem [22] answers this question.

*Theorem 5.2.* (Anderson, Jury, Mansour)

For an interval polynomial

$$p(s, \mathbf{a}) = a_0 + a_1 s + \dots + a_n s^n \quad (5.1.25)$$

with the uncertain parameter vector

$$\mathbf{a} = [a_0 \ a_1 \ \dots \ a_n]^T \quad (5.1.26)$$

in a box

$$\mathcal{A} = \{\mathbf{a} \mid a_i \in [a_i^-; a_i^+], \quad a_i^- > 0, \quad i = 0, 1, \dots, n\} \quad (5.1.27)$$

a testing set  $\mathcal{A}_T$  is

$$\begin{aligned} \mathcal{A}_T &= \{\mathbf{a}^{+-}, \mathbf{a}^{++}, \mathbf{a}^{-+}, \mathbf{a}^{--}\} && \text{for } n > 5 \\ \mathcal{A}_T &= \{\mathbf{a}^{+-}, \mathbf{a}^{++}, \mathbf{a}^{-+}\} && \text{for } n = 5 \\ \mathcal{A}_T &= \{\mathbf{a}^{+-}, \mathbf{a}^{++}\} && \text{for } n = 4 \\ \mathcal{A}_T &= \{\mathbf{a}^{+-}\} && \text{for } n = 3 \end{aligned}$$

□

For  $n = 2$  and  $n = 1$  the condition  $a_i^- > 0$  is necessary and sufficient.

*Proof.*

Let  $n = 3$  and assume that  $p^{+-}(j\omega) := p(j\omega, \mathbf{a}^{+-})$  is stable. Then, its Mikhailov plot traverses through the quadrants I, II, and III, as illustrated by Fig. 5.2 and with  $a_0^- > 0$  the same is true for all  $p(j\omega, \mathbf{a}) \in P(j\omega, \mathcal{A})$ . Let  $n = 4$  and assume

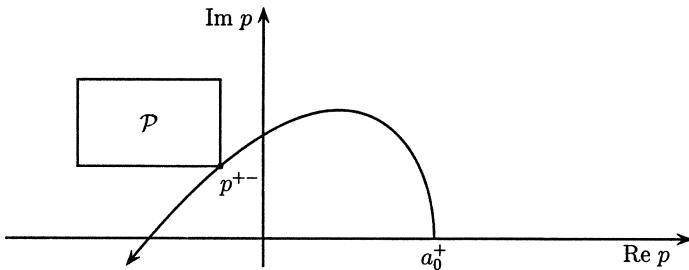


Fig. 5.2: For polynomials of degree three only  $p^{+-}(s)$  has to be tested

that both  $p^{+-}(j\omega)$  and  $p^{++}(j\omega)$  are stable as illustrated by the Mikhailov plots in Fig. 5.3. Then with  $a_0^- > 0$  the same is true for the entire rectangle  $\mathcal{P}$ . Finally, for  $n = 5$  also  $p^{-+}(j\omega)$  must be stable to guarantee that all  $p \in \mathcal{P}$  are stable, see Fig. 5.4. For  $n > 4$  the condition  $a_0^- > 0$  needs not be tested separately, it is implied by stability for  $\mathbf{a}^{-+}$ .

□

For testing the stability of a polynomial, plotting its Mikhailov plot is not the only way. Using the Hurwitz test or factorizing the polynomials are other possibilities.

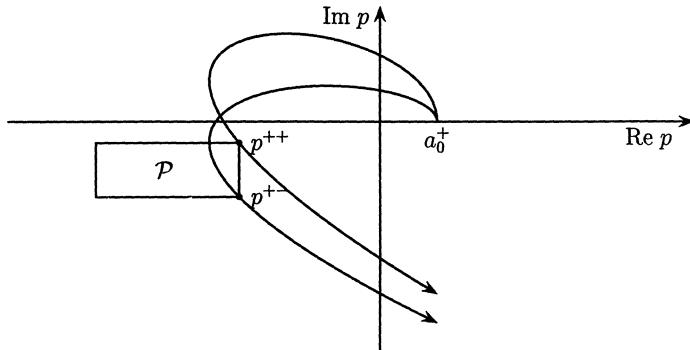


Fig. 5.3: For polynomials of degree four only  $p^{+-}(s)$  and  $p^{++}(s)$  have to be tested

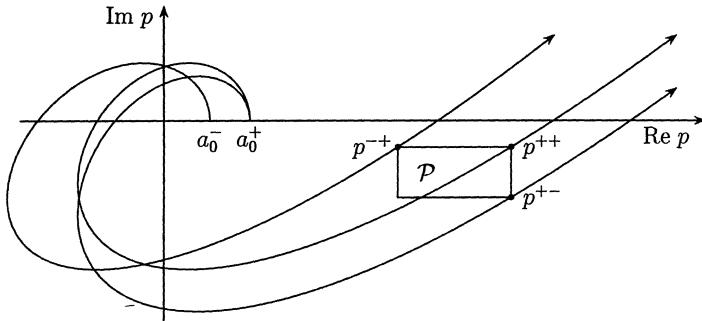


Fig. 5.4: For polynomials of degree five  $p^{+-}(s)$ ,  $p^{++}(s)$  and  $p^{-+}(s)$  have to be tested

*Example 5.3.* Recall the characteristic polynomial of the crane with state feedback (Example 4.3) with the fixed physical parameters  $g = 10, m_C = 1000, \ell = 10, m_L = 1000$ , with fixed feedback gains  $k_1 = 600, k_2 = 2000$ , and nominal values  $k_3 = -10000$  and  $k_4 = 0$ . The controller parameters may vary in the intervals  $k_3 \in [-20000; 0]$  and  $k_4 \in [-10000; 10000]$ . We check the stability of the polynomial family. With the given parameter values the characteristic polynomial is

$$p(s, k_3, k_4) = 6000 + 20000s + (26000 - k_3)s^2 + (20000 - k_4)s^3 + 10000s^4 \quad (5.1.28)$$

This obviously is an interval polynomial. Because of the degree four, only  $p^{+-}(s)$  and  $p^{++}(s)$  have to be tested.

$$p^{+-}(s) = 2000(3 + 10s + 23s^2 + 5s^3 + 5s^4) \quad (5.1.29)$$

$$p^{++}(s) = 2000(3 + 10s + 23s^2 + 15s^3 + 5s^4) \quad (5.1.30)$$

Both polynomials are stable as can for instance be shown by a Hurwitz test. Therefore, the entire polynomial family is stable.  $\square$

Consider now the general uncertain polynomial

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + \dots + a_{n-1}(\mathbf{q})s^{n-1} + a_n(\mathbf{q})s^n, \quad \mathbf{q} \in Q \quad (5.1.31)$$

which not necessarily has the interval property. In vectorial notation we write  $\mathbf{a}(\mathbf{q})$  for the continuous coefficient functions. In general, the set

$$\mathcal{A}(Q) := \{ \mathbf{a}(\mathbf{q}) \mid \mathbf{q} \in Q \} \quad (5.1.32)$$

is not a box and therefore, (5.1.31) is not an interval polynomial. On the other hand, there are complicated coefficient functions  $\mathbf{a}(\mathbf{q})$  that yield a box  $\mathcal{A}(Q) = \mathcal{A}$ .

*Example 5.4.* Consider the polynomial

$$p(s, \mathbf{q}) = (1 + q_1^2) + (q_2 e^{q_3^2} + q_3^2)s + s^2, \quad q_i \in [1; 2], \quad i = 1, 2, 3 \quad (5.1.33)$$

with three uncertain parameters

$$\mathbf{q} = [q_1 \ q_2 \ q_3]^T \quad (5.1.34)$$

At a first glance this polynomial looks complicated for a robustness analysis. But note that each independent variable  $q_i$  appears only in a single coefficient function,

$$\begin{aligned} a_0(\mathbf{q}) &= a_0(q_1) = 1 + q_1^2 \\ a_1(\mathbf{q}) &= a_1(q_2, q_3) = q_2 e^{q_3^2} + q_3^2 \\ a_2(\mathbf{q}) &= a_2 = 1 \end{aligned} \quad (5.1.35)$$

Therefore, the coefficient functions are independent of each other. It follows that the coefficient function,  $\mathbf{a}(\mathbf{q}) = [a_0(\mathbf{q}) \ a_1(\mathbf{q}) \ a_2(\mathbf{q})]^T$  generates a value set

$$\mathcal{A}(Q) := \{ \mathbf{a}(\mathbf{q}) \mid q_i \in [1; 2], \quad i = 0, 1, 2 \} \quad (5.1.36)$$

which is a box

$$\mathcal{A}(Q) = \mathcal{A} = \{ \mathbf{a} \mid a_i \in [a_i^-; a_i^+], \quad i = 0, 1, 2 \} \quad (5.1.37)$$

with

$$a_i^- = \min_Q a_i(\mathbf{q}), \quad a_i^+ = \max_Q a_i(\mathbf{q}) \quad (5.1.38)$$

Thus, having determined these box boundaries we can apply Kharitonov's theorem for answering the robustness question. For an uncertain polynomial given in the general form

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + \dots + a_{n-1}(\mathbf{q})s^{n-1} + a_n(\mathbf{q})s^n, \quad \mathbf{q} \in Q \quad (5.1.39)$$

( $a_i$  continuous in  $\mathbf{q}$ ,  $Q$  a box) it is very easy to see if it has a hidden interval property: If each independent (uncertain) variable  $q_i$  appears just in a single coefficient function, then it is of the interval type. However, for applying the Kharitonov robustness test it remains to determine the interval boundaries (5.1.38). This task can be difficult.  $\square$

In cases of parameters entering only in one coefficient, Kharitonov's Theorem provides a necessary and sufficient condition for stability of the polynomial family. If, however, an uncertain parameter enters into more than one of the coefficients, then these coefficients are mutually dependent. If we overbound dependent coefficients by their individual extremal values, then only a sufficient stability criterion is obtained. This fact is illustrated by the following example.

*Example 5.5.* Consider the polynomial

$$p(s, q_1, q_2) = a_0 + a_1(q_1, q_2)s + a_2(q_1, q_2)s^2 + a_3s^3 \quad (5.1.40)$$

with

$$\begin{aligned} a_0 &= 1 \\ a_1 &= 3 - 2q_1 - 0.5q_2 \\ a_2 &= 0.5 + q_1 + 1.5q_2 \\ a_3 &= 1 \end{aligned} \quad (5.1.41)$$

where the parameters  $q_1, q_2$  are uncertain within the intervals

$$q_i \in [0 ; 1], \quad i = 1, 2 \quad (5.1.42)$$

The two coefficient functions  $a_1, a_2$  are not mutually independent. Thus, the value set  $\mathcal{A}(Q)$  is not a box, i.e. the given polynomial is not an interval polynomial. The box

$$\hat{\mathcal{A}} = \{\mathbf{a} \mid a_i \in [a_i^- ; a_i^+], \quad i = 1, 2, 3\} \quad (5.1.43)$$

with

$$\begin{aligned} a_0^- &= 1, & a_0^+ &= 1 \\ a_1^- &= 0.5, & a_1^+ &= 3 \\ a_2^- &= 0.5, & a_2^+ &= 3 \\ a_3^- &= 1, & a_3^+ &= 1 \end{aligned} \quad (5.1.44)$$

overbounds the value set  $\mathcal{A}(Q)$ ,

$$\hat{\mathcal{A}} \supset \mathcal{A}(Q) \quad (5.1.45)$$

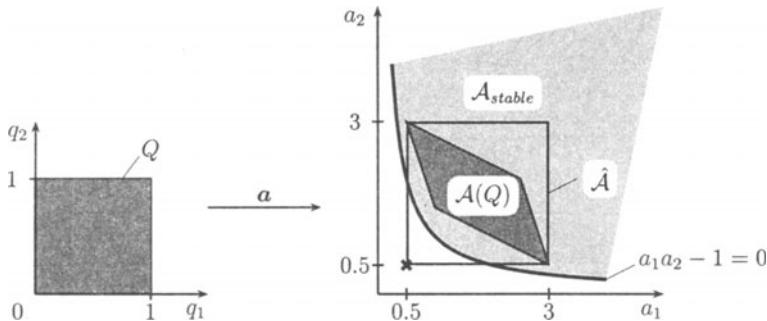
For the robustness test of (5.1.40) to the box  $\hat{\mathcal{A}}$  we only have to check the single Kharitonov polynomial

$$p^{+-}(s) = a_0^+ + a_1^- s + a_2^- s^2 + a_3^+ s^3 = 1 + 0.5s + 0.5s^2 + s^3 \quad (5.1.46)$$

This polynomial is unstable. Thus,  $p(s, q_1, q_2)$  is not robustly stable over the box  $\hat{\mathcal{A}}$ . With this result we still do not know if the polynomial is robustly stable over  $\mathcal{A}(Q)$ . We now make a robustness analysis in the coefficient space using the Hurwitz stability test. The stability boundaries in the  $(a_1, a_2)$ -plane are determined by the equation

$$a_1 a_2 - 1 = 0 \quad (5.1.47)$$

The polynomial is Hurwitz stable for all coefficients  $a_1, a_2$  in the region  $\mathcal{A}_{stable}$  of Fig. 5.5. As we already know from the above Kharitonov test, the box  $\hat{\mathcal{A}}$  is not contained in  $\mathcal{A}_{stable}$ . However, if we map the uncertainty box  $Q$  into the coefficient space we observe that  $\mathcal{A}(Q)$  is contained in  $\mathcal{A}_{stable}$ . Thus,  $p(s, \mathbf{a}(q_1, q_2))$  is robustly stable over  $Q$ . This example demonstrates that overbounding  $\mathcal{A}(Q)$  by a box  $\hat{\mathcal{A}}$  and then applying Kharitonov's theorem for  $p(s, \mathbf{a})$  over the box  $\hat{\mathcal{A}}$  can give a very conservative result.  $\square$

Fig. 5.5: Overbounding the value set  $\mathcal{A}(Q)$  by a box  $\hat{\mathcal{A}}$ 

*Remark 5.2.* Example 5.5 has affine coefficient functions and a general testing set for this class of polynomial families will be derived in Section 5.2. Before we do that we discuss what conclusions can be drawn for the affine case from stability of all vertices of the  $Q$ -box. An interior point  $\mathbf{q}$  of the  $\ell$ -dimensional  $Q$ -box may be represented by its *barycentric coordinates*. A physical interpretation is that we can make  $\mathbf{q}$  the center of gravity (or barycenter) of the box by assigning positive masses to the vertices of the  $Q$ -box. For any point  $\mathbf{q}$  outside the  $Q$ -box at least one of these masses would be negative and if all vertex masses are positive then  $\mathbf{q}$  is inside the box.

Mathematically speaking,  $\mathbf{q} \in Q$  if and only if there exist  $\lambda_m \geq 0$ ,  $m = 1, 2, \dots, 2^\ell$ , such that

$$\mathbf{q} = \sum_{m=1}^{2^\ell} \lambda_m \mathbf{q}_m \quad (5.1.48)$$

where the  $\mathbf{q}_m$  are the vertices of the  $Q$ -box. The sum of the masses is normalized by

$$\sum_{m=1}^{2^\ell} \lambda_m = 1 \quad (5.1.49)$$

Consider now an uncertain polynomial

$$p(s, \mathbf{q}) = [1 \ s \ \dots \ s^n] \ \mathbf{a}_n(\mathbf{q})$$

with the coefficient vector

$$\mathbf{a}_n(\mathbf{q}) = [a_0(\mathbf{q}) \ a_1(\mathbf{q}) \ \dots \ a_n(\mathbf{q})]^T$$

and assume that the uncertain parameters  $\mathbf{q}$  enter affinely, i.e.

$$\mathbf{a}(\mathbf{q}) = \mathbf{a}^0 + \mathbf{F}\mathbf{q}$$

Now substitute the barycentric coordinates (5.1.48) for  $\mathbf{q}$

$$\mathbf{a}(\mathbf{q}) = \mathbf{a}^0 + \mathbf{F} \sum_{m=1}^{2^\ell} \lambda_m \mathbf{q}_m$$

and with (5.1.49)

$$\mathbf{a}(\mathbf{q}) = \sum_{m=1}^{2^\ell} \lambda_m (\mathbf{a}^0 + \mathbf{F} \mathbf{q}_m) \quad (5.1.50)$$

Assume that all vertices of the  $Q$ -box are stable, i.e. all polynomial coefficients vectors  $\mathbf{a}^0 + \mathbf{F} \mathbf{q}_m$ ,  $m = 1, 2, \dots, 2^\ell$  lead to Hurwitz polynomials. Then they must have positive coefficients and with  $\lambda \geq 0$ ,  $\sum_{m=1}^{2^\ell} \lambda_m = 1$  the same applies to  $\mathbf{a}(\mathbf{q})$ . We conclude that the necessary stability condition  $a_i(\mathbf{q}) > 0$ ,  $i = 0, 1, \dots, n$  is satisfied for all  $\mathbf{q} \in Q$ .

By Theorem 4.5 and (4.3.3) the sufficient stability condition that remains to be tested is  $\det \mathbf{H}_{n-1}(\mathbf{q}) \neq 0$  for all  $\mathbf{q} \in Q$ , or with the inequality sign of the Hurwitz theorem

$$\inf_{\mathbf{q} \in Q} \det \mathbf{H}_{n-1}(\mathbf{q}) > 0 \quad (5.1.51)$$

With the definition of the Hurwitz matrix and (5.1.50) we have

$$\mathbf{H}_{n-1}(\mathbf{q}) = \sum_{m=1}^{2^\ell} \lambda_m \mathbf{H}_{n-1}(\mathbf{q}_m) \quad (5.1.52)$$

$\mathbf{H}_{n-1}(\mathbf{q})$  may have a negative determinant although all vertex Hurwitz determinants  $\mathbf{H}_{n-1}(\mathbf{q}_m)$  have positive determinants. Thus, stability for all  $\mathbf{q} \in Q$  cannot be concluded from stability of the vertices of  $Q$ . In the affine Example 4.12 the nonconvex stability region is shown in Fig. 4.8. It is easy to choose a  $Q$ -box that has four stable vertices but is not contained in  $Q_{stable}$ .  $\square$

*Remark 5.3.* The proof of Kharitonov's Theorem was easy after the observation that the value set  $p(j\omega, \mathcal{A})$  of an interval polynomial for fixed  $\omega$  is a rectangle with edges parallel to the real and imaginary axes. A natural question is: Are there other classes of polynomials which have such a rectangular value set. Indeed it is easy to show that polynomials with *even-odd decoupling*

$$p(s, \mathbf{q}, \mathbf{r}) = h(s^2, a_0(\mathbf{q}), a_2(\mathbf{q}), \dots) + sg(s^2, a_1(\mathbf{r}), a_3(\mathbf{r}), \dots)$$

have the above property.

The uncertain parameters  $\mathbf{q}$  enter only in the even order coefficients, the uncertain parameters  $\mathbf{r}$  enter only in the odd ordered coefficients. Then, as in (5.1.8) and (5.1.9), the bounds of the real and imaginary parts are mutually independent and the value set is a rectangle with edges parallel to the real and imaginary axis.

$$\begin{aligned} h^-(\omega^2, a_0(\mathbf{q}), a_2(\mathbf{q}), \dots) &\leq \operatorname{Re} p(j\omega, \mathbf{q}) \leq h^+(\omega^2, a_0(\mathbf{q}), a_2(\mathbf{q}), \dots) \\ \omega g^-(\omega^2, a_1(\mathbf{r}), a_3(\mathbf{r}), \dots) &\leq \operatorname{Im} p(j\omega, \mathbf{q}) \leq \omega g^+(\omega^2, a_1(\mathbf{r}), a_3(\mathbf{r}), \dots) \end{aligned}$$

The determination of  $h^-, h^+, g^-$  and  $g^+$  is easy if the uncertain parameters  $\mathbf{q}$  and  $\mathbf{r}$  enter affinely into the coefficients  $a_i$ . Then a feasible testing set consists of all combinations of extremal values of  $\mathbf{q}$  and  $\mathbf{r}$ , see Panier et al. [137].  $\square$

## 5.2 Affine Coefficients: Edge Theorem

The next step is to investigate the polynomial family

$$P(s, Q) = \{ p(s, q_1, q_2, \dots, q_\ell) = p_0(s) + \sum_{i=1}^{\ell} q_i p_i(s) \mid q_i \in [q_i^-; q_i^+], \quad i = 1, 2, \dots, \ell \} \quad (5.2.1)$$

The coefficients depend linearly on the uncertain parameter vector  $\mathbf{q} = [q_1 \ q_2 \ \dots \ q_\ell]^T$ . More precisely, it is an affine dependency because of the additional constant terms, i.e. each coefficient  $a_i$  has the form

$$a_i = a_i^0 + f_{1i}q_1 + f_{2i}q_2 + \dots + f_{\ell i}q_\ell, \quad i = 0, 1, \dots, n \quad (5.2.2)$$

The parameters  $q_i$  vary in an  $\ell$ -dimensional box  $Q$ :

$$q_i \in [q_i^-; q_i^+], \quad i = 1, 2, \dots, \ell \quad (5.2.3)$$

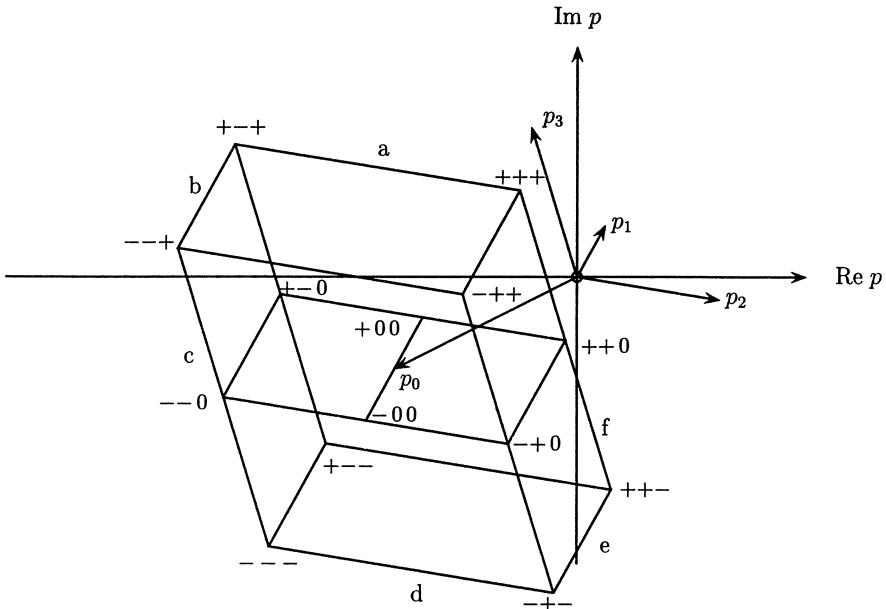


Fig. 5.6: The construction of the value set with three parameters and fixed  $\omega$

The polynomial family  $P(s, Q) = \{ p(s, \mathbf{q}) \mid \mathbf{q} \in Q \}$  represents a parallelepiped (the generalization of the parallelogram to higher-dimensional spaces) in the coefficient space because of the affine mapping (5.2.2), see for example the parallelogram  $\mathcal{A}(Q)$  in Fig. 5.5. Its vertices and edges are generated by vertices and edges of the  $Q$ -box. If all parameter values  $q_i$  take their minimum or maximum value then the corresponding

polynomial is called a vertex polynomial. If exactly one of these parameters varies between its minimum and maximum value while the remaining  $\ell - 1$  parameters stick either to their minimal or maximal values, then this polynomial family is called an edge polynomial. The number of vertices and edges of this parallelepiped may be smaller than the number of vertices and edges of the original box  $Q$  depending on the numbers  $\ell$  and  $n$ . For  $\ell > n$  not all edges of the box are mapped to edges of the parallelepiped. Some edges are mapped into the interior. One might assume that a necessary and sufficient condition for stability would be that the vertices of the parallelepiped (or even a subset) are stable. The necessity is trivial but this set is too small for sufficiency. A testing set was found by Bartlett et al. [35]. It consists of a finite set of one-dimensional tests.

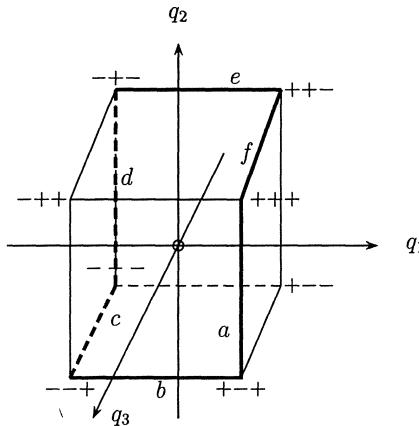


Fig. 5.7: The edges  $a - f$  generate the boundary of the value set of Fig. 5.6

*Theorem 5.3. (Edge Theorem of Bartlett, Hollot and Huang)*

The polynomial family  $P(s, Q) = \{ p(s, \mathbf{q}) = \sum_{i=0}^n a_i(\mathbf{q})s^i \mid \mathbf{q} \in Q \}$  with affine coefficient functions  $a_i(\mathbf{q})$  and  $Q = \{ \mathbf{q} \mid q_i \in [q_i^-; q_i^+], i = 1, 2, \dots, \ell \}$  is stable if and only if the edges of  $Q$  are stable.

□

For notational convenience of the proof, the original box is scaled to the unit box with sidelength 2 by the transformation

$$\tilde{q}_i = \frac{2q_i - q_i^+ - q_i^-}{q_i^+ - q_i^-}, \quad i = 1, 2, \dots, \ell \quad (5.2.4)$$

It maps  $q_i = q_i^-$  to  $\tilde{q}_i = -1$  and  $q_i = q_i^+$  to  $\tilde{q}_i = +1$ . The tilde is now omitted and we have without loss of generality  $q_i \in [-1; +1]$ .

Before proving Theorem 5.3 the value set  $\mathcal{P}(j\omega, Q)$  with affine coefficient functions will be constructed. In the case of two parameters (see Example 4.14) the value set  $\mathcal{P}(j\omega, Q)$  for fixed  $\omega$  is a parallelogram in the complex plane.

Consider now a third parameter, i.e. we have the affine polynomial family

$$\{ p(s, q_1, q_2, q_3) = p_0(s) + q_1 p_1(s) + q_2 p_2(s) + q_3 p_3(s) \mid q_i \in [-1; +1], \quad i = 1, 2, 3 \} \quad (5.2.5)$$

For fixed frequency  $s = j\omega^*$  we calculate the four complex numbers  $p_0(j\omega^*)$ ,  $p_1(j\omega^*)$ ,  $p_2(j\omega^*)$ ,  $p_3(j\omega^*)$  and draw four vectors at the origin which represent these numbers (see Fig. 5.6).

Starting from  $p_0$ , add the vectors  $q_1^+ p_1$  and  $q_1^- p_1$  and obtain the points  $+00$  resp.  $-00$ . The segment  $[+00; -00]$  is the value set for the polynomial family  $p(s, q_1, 0, 0)$ . In  $+00$  we add  $q_2^+ p_2$  (arriving at  $++0$ ) resp.  $q_2^- p_2$  (arriving at  $+ - 0$ ) and repeat this procedure at  $-00$ . We obtain the parallelogram  $+ + 0, + - 0, - - 0, - + 0$ , which corresponds to the polynomial family  $p(s, q_1, q_2, 0)$ . The last step is adding  $q_3^+ p_3$  resp.  $q_3^- p_3$  at all four vertices to get a hexagon  $+++, + -, - -, - + -, + + -$ , which is the value set of the polynomial family (5.1.20) for the fixed frequency  $\omega^*$ .

Denote the six edges of the value set, with  $a$  to  $f$ , see Fig. 5.6. They are the image of six pairwise adjacent edges of the  $Q$ -box as indicated in Fig. 5.7. The other six edges and also two vertices are mapped into the interior of the hexagon. Observe that vertices and edges which contribute to the boundary of the hexagon are not the same with varying frequency. Changing the frequency leads to different values of  $p_i(j\omega)$ ,  $i = 0, 1, 2, 3$ , where  $p_0(j\omega)$  determines the midpoint of the hexagon and the vectors  $p_1(j\omega)$ ,  $p_2(j\omega)$  and  $p_3(j\omega)$  determine the size and orientation. The transition frequencies where the image of an edge of the  $Q$ -box changes from the boundary into the interior and vice versa may be calculated by comparing the phases of  $p_1(j\omega)$ ,  $p_2(j\omega)$  and  $p_3(j\omega)$ . If two of them are equal then a transition occurs. At the transition frequency the value set  $\mathcal{P}(j\omega, Q)$  degenerates from a hexagon to a parallelogram.

Adding a further term  $q_4 p_4(s)$  to (5.2.5) requires adding  $q_4^+ p_4$  resp.  $q_4^- p_4$  which yields an octagon. Obviously, the value set of an affine family of polynomials for fixed frequency is always a convex polygon with at most  $2\ell$  vertices (this number may decrease for special frequencies). Two opposite vertices are parallel. This special kind of polygon is called a *parpolygon*. With these preparations we are now ready to prove the edge theorem.

*Proof.*

The necessity is obvious. We use the zero exclusion principle for the proof of sufficiency. With varying frequency the shape of the parpolygon is changing and it is moving around the origin. If the polynomial family is unstable then there is a frequency where the boundary of the parpolygon, i.e. an edge goes through the origin of the  $p$ -plane. Because this edge is the image of an edge of the  $Q$ -box, the instability can be recognized by testing all edges of the  $Q$ -box.

□

*Remark 5.4.* It is not necessary to test all edges, however, the amount of calculation to determine the critical edges is about the same as to test all edges. A procedure which avoids the mapping of superfluous edges was given by Fu [70]. It reduces the problem of the computation of all  $2^\ell$  extremal points to the computation of  $\ell$  points followed by a sorting problem.  $\square$

*Remark 5.5.* The edge theorem is valid for a more general class of polynomial families. It was shown in [35], that a “polytope” of polynomials is robustly stable if and only if all its edges are stable. A polytope of polynomials is the higher-dimensional generalization of a convex polygon in two dimensions. It is the convex hull of a finite number of points. A polytope of polynomials can be written in the form

$$\begin{aligned} P_\ell &= \text{conv} \{ p_1(s), p_2(s), \dots, p_\ell(s) \} = \\ &= \{ p(s) = \sum_{i=1}^{\ell} \lambda_i p_i(s), \quad \sum_{i=1}^{\ell} \lambda_i = 1, \quad \lambda_i \geq 0, \quad i = 1, 2, \dots, \ell \} \end{aligned}$$

Each edge corresponds to polynomials  $p(s, \mathbf{q})$  where exactly one parameter  $q_i$  varies between its given limits while the remaining  $\ell - 1$  parameters take their minimum or maximum value. Note that the  $\lambda_i$  are again barycentric coordinates as in (5.1.48). There, however, a special polytope (box) in  $\mathbf{q}$ -space was considered. The above formulation applies to an arbitrary polytope in the space of coefficient vectors  $\mathbf{a}$ .  $\square$

The edge theorem reduces the robustness analysis of an affine polynomial family to a finite set of one-dimensional tests. This is a drastic reduction of the testing set compared with the complete set. An  $\ell$ -dimensional box has  $2^\ell$  vertices. From each vertex start  $\ell$  edges. The number  $\ell 2^\ell$  would count each edge two times, thus, the total number of edges is  $\ell 2^{\ell-1}$ . For some values of  $\ell$  the number of vertices and edges is given in Table 5.1.

$\ell$	$2^\ell$ vertices	$\ell 2^{\ell-1}$ edges
1	2	1
2	4	4
3	8	12
4	16	32
5	32	80
10	1024	5120
20	1048576	10485760

Table 5.1: Number of vertices and edges of an  $\ell$ -dimensional box

For a large number of parameters the number of edges to be tested becomes prohibitively large. In any case, it is of interest to use an efficient test for an edge.

Consider an edge with end points  $p_a$  and  $p_b$ , i.e. the polynomial family

$$P(s, Q) = \{ (1 - q_1)p_a(s) + q_1 p_b(s) \mid q_1 \in [0; 1] \} \quad (5.2.6)$$

The Bialas test of Theorem 4.7 can be executed by a standard eigenvalue program for the calculation of all eigenvalues of the Bialas matrices  $(\mathbf{H}_{n-1}^a)^{-1} \mathbf{H}_{n-1}^b$ . Note that for each edge only the  $n - 1$  eigenvalues of a matrix must be computed. This is much more efficient than using the root locus test for the edge (5.2.6) which requires factorizing many polynomials depending on the choice of the grid. Plotting the eigenvalues gives a graphical warning of closeness to instability if a pair of complex eigenvalues is close to the negative real axis.

*Example 5.6.* Recall the characteristic polynomial of Example 4.12 with  $q_3 = 0$

$$p(s, q_1, q_2) = (14 - 0.3q_1 + 2q_2) + (10 + 2q_1 + 8q_2)s + 10s^2 + 2(1 + q_1)s^3 + s^4 \quad (5.2.7)$$

and  $q_1 \in [15; 25]$ ,  $q_2 \in [0; 3.6]$ . Corresponding to the four vertices we have the four vertex polynomials

$$\begin{aligned} p^{(1)}(s, 15, 0) &= 9.5 + 40s + 10s^2 + 32s^3 + s^4 \\ p^{(2)}(s, 15, 3.6) &= 16.7 + 68.8 + 10s^2 + 32s^3 + s^4 \\ p^{(3)}(s, 25, 0) &= 6.5 + 60s + 10s^2 + 52s^3 + s^4 \\ p^{(4)}(s, 25, 3.6) &= 13.7 + 88.8s + 10s^2 + 52s^3 + s^4 \end{aligned}$$

with the Hurwitz matrices

$$\mathbf{H}^{(1)} = \begin{bmatrix} 32 & 40 & 0 \\ 1 & 10 & 9.5 \\ 0 & 32 & 40 \end{bmatrix}, \quad \mathbf{H}^{(2)} = \begin{bmatrix} 32 & 68.8 & 0 \\ 1 & 10 & 6.7 \\ 0 & 32 & 68.8 \end{bmatrix} \quad (5.2.8)$$

$$\mathbf{H}^{(3)} = \begin{bmatrix} 52 & 60 & 0 \\ 1 & 10 & 6.5 \\ 0 & 52 & 40 \end{bmatrix}, \quad \mathbf{H}^{(4)} = \begin{bmatrix} 52 & 88.8 & 0 \\ 1 & 10 & 13.7 \\ 0 & 52 & 88.8 \end{bmatrix} \quad (5.2.9)$$

The Bialas test yields

$$\begin{aligned} \text{eigenvalues of } \mathbf{H}^{(1)}(\mathbf{H}^{(2)})^{-1} &= \{13.04, 0.62, 1.00\} \\ \text{eigenvalues of } \mathbf{H}^{(4)}(\mathbf{H}^{(2)})^{-1} &= \{1.77, -1.85 \pm 0.67j\} \\ \text{eigenvalues of } \mathbf{H}^{(1)}(\mathbf{H}^{(3)})^{-1} &= \{5.22, 0.65, 0.29\} \\ \text{eigenvalues of } \mathbf{H}^{(4)}(\mathbf{H}^{(3)})^{-1} &= \{0.82, 1.00, 1.00\} \end{aligned}$$

All Bialas eigenvalues are shown in Fig. 5.8.

The second set contains a complex root pair close to the negative real axis. This indicates that the edge from  $p^{(2)}(s)$  to  $p^{(4)}(s)$  is close to the instability region. This result is confirmed if we plot the  $Q$ -box into the region  $Q_{stable}$  of Fig. 4.9.

□

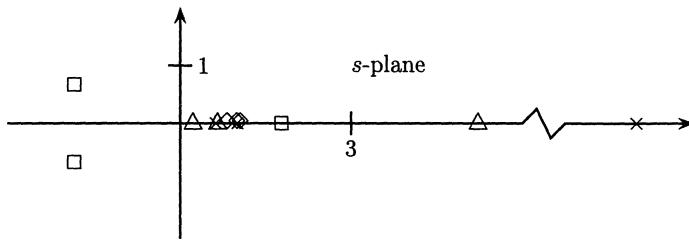


Fig. 5.8: The eigenvalues of the Bialas matrices corresponding to the four edge polynomials, marked with  $\times$ ,  $\square$ ,  $\triangle$  and  $\diamond$

Some more complicated polynomials  $p(s, \mathbf{q})$  can be factorized as

$$p(s, \mathbf{q}) = f(\mathbf{q})\bar{p}(s, \mathbf{q}) \quad (5.2.10)$$

where  $\bar{p}(s, \mathbf{q})$  is affine in  $\mathbf{q}$  and  $f(\mathbf{q}) \neq 0$  for all admissible  $\mathbf{q}$ . Then  $p(s, \mathbf{q})$  and  $\bar{p}(s, \mathbf{q})$  have the same roots and we can use the edge theorem. Such occurrences are called *hidden affine cases*, see also (2.7.5) and Exercise 2.5.

*Example 5.7.* Recall the characteristic polynomial of Example 4.3 with  $m_C = 1000$  and the uncertain parameters  $m_L$  and  $\ell$ . With the output feedback gains  $k_1 = 500$ ,  $k_2 = 2769$ ,  $k_3 = -21557$  and  $k_4 = 0$  the characteristic polynomial is

$$p(s, \ell, m_L) = \frac{5}{\ell} + \frac{27.69}{\ell}s + [0.5 + \frac{31.557}{\ell} + \frac{0.01m_L}{\ell}]s^2 + \frac{2.769}{\ell}s^3 + s^4$$

which is multilinear in  $m_L$  and  $1/\ell$ . Introducing new uncertain parameter  $q_1 = \ell$ ,  $q_2 = m_L$  and multiplying both sides by  $q_1 = \ell > 0$ , we have

$$q_1 p(s, q_1, q_2) = 5 + 27.69s + (31.557 + 0.5q_1 + 0.01q_2)s^2 + 2.769s^3 + q_1 s^4 \quad (5.2.11)$$

Now the coefficient functions are affine in  $q_1$  and  $q_2$ . □

### 5.3 A Warning Example

The last two sections have shown that there exist testing sets  $Q_T$  for interval polynomials and for polynomials with affine coefficient dependencies which are proper subsets of  $Q$ . In the first case, the testing set consists of four special vertex polynomials and in the second case of all edge polynomials of the  $Q$ -box.

A more general polynomial family has multilinear coefficient functions, that are functions with terms like  $q_1 q_2$ ,  $q_2 q_3$ ,  $q_1 q_2 q_3$  and so on but no terms like  $q_1^2$  or  $q_1 q_2^2$ . For this type of coefficient functions  $a_i(\mathbf{q})$ , in general there do not exist proper testing sets of  $Q$ , as we will demonstrate in the next example [9].

*Example 5.8.* Consider the third degree uncertain polynomial

$$p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + a_2(\mathbf{q})s^2 + s^3 \quad (5.3.1)$$

with the multilinear coefficient functions

$$a_0(\mathbf{q}) = \ell(\ell - 1) + r^2 + 2(\ell + 1) \sum_{i=1}^{\ell} q_i + 2 \sum_{i=1}^{\ell-1} \sum_{j=i+1}^{\ell} q_i q_j \quad (5.3.2)$$

$$a_1(\mathbf{q}) = \ell + \sum_{i=1}^{\ell} q_i \quad (5.3.3)$$

$$a_2(\mathbf{q}) = a_1(\mathbf{q}) \quad (5.3.4)$$

Let  $\mathbf{q} \in Q^+ = \{ \mathbf{q} \mid q_i > 0, i = 1, 2, \dots, \ell \}$ , then all coefficients  $a_i(\mathbf{q})$  are positive and the only remaining Hurwitz stability conditions is  $a_1(\mathbf{q})a_2(\mathbf{q}) - a_0(\mathbf{q}) > 0$ . It is easily verified that

$$a_1(\mathbf{q})a_2(\mathbf{q}) - a_0(\mathbf{q}) = \sum_{i=1}^{\ell} (q_i - 1)^2 - r^2 \quad (5.3.5)$$

The resulting stability condition

$$\sum_{i=1}^{\ell} (q_i - 1)^2 > r^2 \quad (5.3.6)$$

is satisfied outside a ball of radius  $r$  and center  $\mathbf{q}^0 = [1 \ 1 \ \dots \ 1]^T$ .

Next, we let  $r$  go to zero. Thus, the unstable ball shrinks to an isolated unstable point  $\mathbf{q}^0$ . With the exception of this point, the polynomial is stable for all  $\mathbf{q} \in Q^+$ . Now choose an arbitrary uncertainty domain  $Q \subset Q^+$  such that  $\mathbf{q}^0 \in Q$ .  $Q$  is not robustly stable because it contains an isolated unstable point. It is stable otherwise. Generically, no subset is suited to show the stability of  $Q$ . An example of this type was given already in 1961 bei Truxal [164].  $\square$

The above example shows that generically for nonlinear coefficient functions also interior points of  $Q$  must be tested for stability. A brute force approach is gridding of the  $Q$ -box and a large number of stability tests for the grid points  $Q_g$ . It would however not suffice to do numerical tests on the grid. An isolated unstable point as in the above example would be missed by the grid and instability would remain undiscovered. Practically a graphical representation of the results helps to get a warning of the “closeness to instability”. The graphics can be obtained for example by factorizing each grid point polynomial and displaying the root set as shown in Section 4.1. A more efficient test avoiding the factorization is based on Theorem 4.5. The worst case operating conditions  $\mathbf{q}_w$  on the grid  $Q_g$  is determined by selecting

$$\mathbf{q}_w = \min_{\mathbf{q} \in Q_g} \det \mathbf{H}_n(\mathbf{q}) \quad (5.3.7)$$

If necessary, a small neighborhood of  $\mathbf{q}_w$  must then be investigated. Note that by (4.3.3)  $\det \mathbf{H}_n = a_0 \det \mathbf{H}_{n-1}$  can be used to simplify the calculation. The determinant of the Hurwitz matrix must be evaluated and simplified first by a computer algebra program.

Gridding approaches are feasible only for a small number of parameters. For example let  $\ell = 5$  and each parameter interval  $q_i \in [q_i^-; q_i^+]$ ,  $i = 1, 2, \dots, 5$  be represented by ten grid points. Then  $10^5$  evaluations of  $\det \mathbf{H}_n(\mathbf{q})$  must be made and searched for the worst case  $\mathbf{q}_w$ . In the affine case with  $\ell = 5$  we have to test 80 edges, see Table 5.1.

We try to avoid gridding  $Q$  as far as possible, therefore, we will use the frequency domain methods of Sections 4.4 through 4.6 in the following chapters. Then the frequency  $\omega$  must be gridded for the execution of the robust stability test.

## 5.4 Jacobi Conditions

In Example 5.8 we can choose a box  $Q$  such that its surface (vertices, edges, faces etc.) is stable, but an interior point is unstable. For the case of two parameters the value sets will be investigated and the Jacobian conditions will be introduced. This will yield a further testing set result.

*Example 5.9.* Consider the uncertain polynomial (5.3.1) for  $\ell = 2$ ,  $r = 0.5$ .

$$p(s, q_1, q_2) = 2q_1q_2 + 6q_1 + 6q_2 + 2.25 + (q_1 + q_2 + 2)s + (q_1 + q_2 + 2)s^2 + s^3$$

with  $q_1 \in [0.3; 2.5]$  and  $q_2 \in [0; 1.7]$ . The value set  $\mathcal{P}(j\omega, q_1, q_2)$  for the fixed frequency  $\omega^* = 2$  is determined. An approximation of the value set may be obtained by gridding  $q_2$ . For fixed  $q_2 = q_2^*$  the coefficient functions are affine, i.e. a line segment parallel to the  $q_1$ -axis maps into a line segment in the  $p$ -plane.

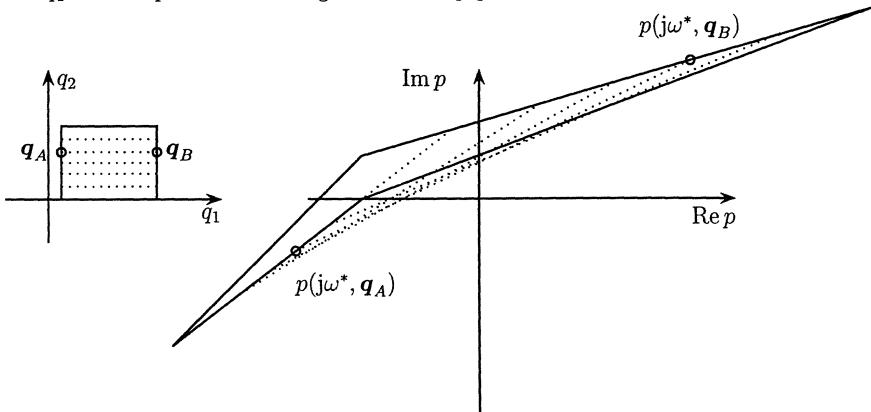


Fig. 5.9: A straight line segment parallel to the  $q_1$ -axis is mapped to a straight line segment

Fig. 5.9 shows the image of the box boundary in solid lines and the image of the line segment  $q_A - q_B$  sweeping stepwise over the interval  $q_2 \in [0; 1.7]$  in dotted lines. There

are interior points of  $Q$  which are mapped to the boundary of the value set. A necessary condition for an interior point of  $Q$  to map into a boundary point of  $P(j\omega^*, q_1, q_2)$  is that the Jacobian determinant of  $p(j\omega^*, q_1, q_2)$  vanishes.  $\square$

*Definition 5.3.* Consider two real functions of two variables  $x = x(q_1, q_2)$ ,  $y = y(q_1, q_2)$ . The *Jacobi matrix*  $\mathbf{J}$  (or *Jacobian*) of the functions is defined by

$$\mathbf{J} := \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} \end{bmatrix}$$

The corresponding determinant  $J = \det \mathbf{J}$  is called *Jacobi determinant*.  $\square$

The significance of the Jacobian for the case of affine functions is first explained. The functions

$$\begin{bmatrix} x(q_1, q_2) \\ y(q_1, q_2) \end{bmatrix} = \begin{bmatrix} \operatorname{Re} p(j\omega^*, q_1, q_2) \\ \operatorname{Im} p(j\omega^*, q_1, q_2) \end{bmatrix} = \begin{bmatrix} a_0(\omega^*) + a_1(\omega^*)q_1 + a_2(\omega^*)q_2 \\ b_0(\omega^*) + b_1(\omega^*)q_1 + b_2(\omega^*)q_2 \end{bmatrix} \quad (5.4.1)$$

are affine functions of  $q_1$  and  $q_2$  for fixed  $\omega$ . In short notation we have

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{J} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} + \begin{bmatrix} a_0(\omega^*) \\ b_0(\omega^*) \end{bmatrix} = \mathbf{e}_0 + q_1 \mathbf{e}_1 + q_2 \mathbf{e}_2 \quad (5.4.2)$$

with the Jacobian determinant

$$J = a_1 b_2 - a_2 b_1 \quad (5.4.3)$$

Let  $J \neq 0$ , then every  $Q$ -box of the  $(q_1, q_2)$ -plane is mapped via (5.4.2) to a parallelogram in the  $(x, y)$ -plane spanned by the two vectors  $\mathbf{e}_1$  and  $\mathbf{e}_2$  which start at the endpoint of  $\mathbf{e}_0$ , see Fig. 5.10.

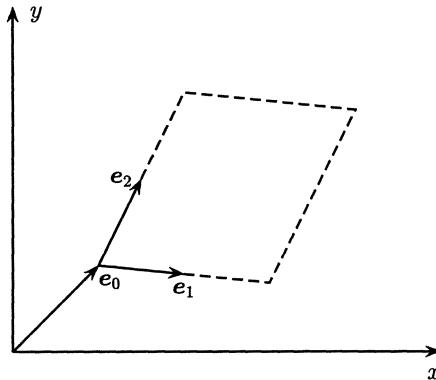
If  $J > 0$ , then the orientation remains unchanged, i.e. if we move around the image of the edges of the  $Q$ -box in the order  $p^{--}, p^{+-}, p^{++}, p^{-+}$  then the image of the  $Q$ -box is on the left side. If  $J < 0$  then the orientation is changed, the image of the  $Q$ -box is to the right side. An exception occurs if  $J = 0$ . Then the vectors  $\mathbf{e}_1$  and  $\mathbf{e}_2$  are linear dependent and the image of the  $Q$ -box reduces to a segment.

Now assume that  $x = x(q_1, q_2)$ ,  $y = y(q_1, q_2)$  are multilinear in  $q_1$  and  $q_2$ . Instead of (5.4.1) we now have

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \operatorname{Re} p(j\omega^*, q_1, q_2) \\ \operatorname{Im} p(j\omega^*, q_1, q_2) \end{bmatrix} = \begin{bmatrix} a_0(\omega^*) + a_1(\omega^*)q_1 + a_2(\omega^*)q_2 + a_{12}(\omega^*)q_1 q_2 \\ b_0(\omega^*) + b_1(\omega^*)q_1 + b_2(\omega^*)q_2 + b_{12}(\omega^*)q_1 q_2 \end{bmatrix} \quad (5.4.4)$$

and the corresponding Jacobian is

$$\mathbf{J} = \begin{bmatrix} a_1 + a_{12}q_2 & a_2 + a_{12}q_1 \\ b_1 + b_{12}q_2 & b_2 + b_{12}q_1 \end{bmatrix} \quad (5.4.5)$$

Fig. 5.10: The image of a  $Q$ -box with  $J \neq 0$ 

with

$$J = (a_1 b_2 - a_2 b_1) + (a_1 b_{12} - a_{12} b_1)q_1 + (a_{12} b_2 - a_2 b_{12})q_2 \quad (5.4.6)$$

The Jacobi determinant is now a function of  $q_1$  and  $q_2$ . If  $J \neq 0$  for all  $q \in Q$ , then the value set for fixed  $\omega^*$  is a quadrangle. The vertices are the images of the vertices of the  $Q$ -box. The orientation of the quadrangle depends on the sign of  $J$ .

An interesting case occurs if  $J = 0$  for some  $q \in Q$ . From (5.4.6) it can be recognized that these points lie on a straight line (Jacobi line) which intersects the  $Q$ -box. For the construction of the complete value set this segment additionally has to be mapped.

*Example 5.10.* Consider again Example 5.9. This polynomial family is unstable inside the ball  $(q_1 - 1)^2 + (q_2 - 1)^2 - 0.5^2 = 0$  (see Example 5.4). The mapping equations are

$$\begin{bmatrix} \operatorname{Re} p(j\omega, q_1, q_2) \\ \operatorname{Im} p(j\omega, q_1, q_2) \end{bmatrix} = \begin{bmatrix} 2.25 - 2\omega^2 + (6 - \omega^2)q_1 + (6 - \omega^2)q_2 + 2q_1q_2 \\ \omega(2 - \omega^2) + \omega q_1 + \omega q_2 \end{bmatrix} \quad (5.4.7)$$

The Jacobian is

$$\mathbf{J} = \begin{bmatrix} 6 - \omega^2 + 2q_2 & 6 - \omega^2 + 2q_1 \\ \omega & \omega \end{bmatrix} \quad (5.4.8)$$

and its determinant is

$$J = 2\omega(q_2 - q_1) \quad (5.4.9)$$

Instead of gridding  $q_2$  and mapping the segments as in Example 5.9 we have to map only the edges of  $Q$  and the Jacobi line  $q_1 = q_2 = q$ . The real and imaginary parts  $x$  and  $y$  depend only on the parameter  $q$  and we have

$$\begin{aligned} x &= 2.25 - 2\omega^2 + (12 - 2\omega^2)q + 2q^2 \\ y &= \omega(2 - \omega^2) + 2\omega q \end{aligned}$$

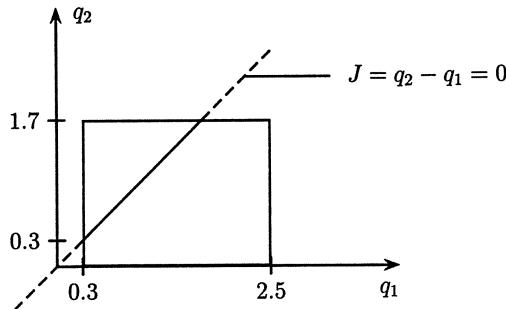


Fig. 5.11: The  $Q$ -box and the Jacobi line have common points, so additionally the Jacobi line has to be mapped

with  $q \in [0.3; 1.7]$  corresponding to the intersection points of the Jacobi line with the  $Q$ -box, see Fig. 5.11. The image of the Jacobi line is not a straight line segment, therefore, this line has to be gridded. The parametric representation shows that the image of the Jacobi line for fixed  $\omega$  is a part of a parabola because of the quadratic dependency on  $q$ . Fig. 5.12 shows the complete value set for  $\omega = 2.2$  combining the nonconvex quadrangle and the part of the parabola arising from the condition  $J = 0$ .  $\square$

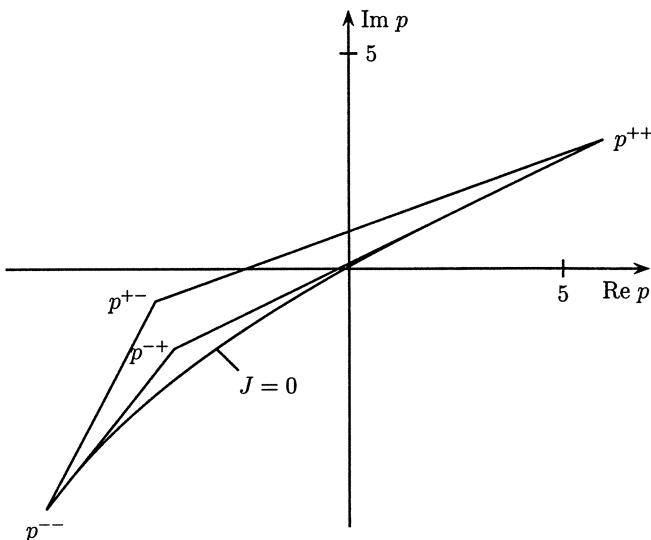


Fig. 5.12: Value set combining the images of the edges and the Jacobi line

In this example the equation of the Jacobi line is frequency independent. Therefore, we only have to test one fixed Jacobi segment (the segment of the Jacobi line that

intersects  $Q$ ) in addition to the edges as the testing set. In general however, the Jacobi line is frequency dependent and the interior points that contribute to the boundary of the value set must be identified for each frequency.

*Remark 5.6.* If one of the two functions (5.4.4) depends only on one parameter (say  $x$  is independent of  $q_2$ ) then the mapping equations are

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \operatorname{Re} p(j\omega^*, q_1) \\ \operatorname{Im} p(j\omega^*, q_1, q_2) \end{bmatrix} = \begin{bmatrix} a_0(\omega^*) + a_1(\omega^*)q_1 \\ b_0(\omega^*) + b_1(\omega^*)q_1 + b_2(\omega^*)q_2 + b_{12}(\omega^*)q_1q_2 \end{bmatrix} \quad (5.4.10)$$

The edges parallel to the  $q_2$ -axis are mapped to segments parallel to the imaginary axis. The corresponding Jacobian matrix is

$$\mathbf{J} = \begin{bmatrix} a_1 & 0 \\ b_1 + b_{12}q_2 & b_2 + b_{12}q_1 \end{bmatrix} \quad (5.4.11)$$

with

$$J = a_1(b_2 + b_{12}q_1) = 0 \quad (5.4.12)$$

The Jacobi line  $q$

$$q_1^* = -b_2/b_{12}$$

is parallel to the imaginary axis. Despite of the fact that the Jacobian line intersects the  $Q$ -box, the boundary of the value set is already determined by the edges. A classification of the type of boundaries of the value set in the bilinear case can be found in [8].  $\square$

In the above discussion  $\ell = 2$  was assumed, i.e. there are only two uncertain parameters  $q_1, q_2$  and only one bilinear term  $q_1q_2$ . The next definition is the extension of Definition 5.3 to  $\ell$  variables.

*Definition 5.4.* Let  $x = x(q_1, q_2, \dots, q_\ell)$ ,  $y = y(q_1, q_2, \dots, q_\ell)$  be two real functions of  $\ell$  variables. Then the *Jacobi matrix*  $\mathbf{J}$  (or *Jacobian*) of the functions  $x$  and  $y$  is defined by

$$\mathbf{J} := \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} & \cdots & \frac{\partial x}{\partial q_\ell} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} & \cdots & \frac{\partial y}{\partial q_\ell} \end{bmatrix}$$

$\square$

The construction of the value set for  $\ell$  parameters requires again mapping the edges. Instead of the pairs  $[q_1 \ q_2]^T$  for which  $J = 0$  first the points  $[q_1 \ q_2 \ \dots \ q_\ell]^T$  for which  $\mathbf{J}$  has rank  $< 2$  have to be determined. This means that all  $2 \times 2$ -subdeterminants of  $\mathbf{J}$  must vanish. The solution vectors of this system of equations contribute to the boundary of the value set. These vectors are in general not easy to describe. Only in special cases a satisfying description is possible. In most practical cases gridding the whole  $Q$ -box is more effective than solving the system of equations. A method which reduces the number of grid points for special cases will be shown in the next chapter.

## 5.5 The Mapping Theorem

For multilinear polynomial families the “mapping theorem” of Desoer [181] yields a useful sufficient stability condition that can restrict frequency gridding to a subset of the nonnegative frequencies. The robust stability test is by zero exclusion from the value set, see Theorem 4.11. The mapping theorem gives a simple description of the convex hull of the value set.

*Theorem 5.4. (Mapping Theorem of Desoer)*

The convex hull of the value set  $\mathcal{P}(j\omega^*, Q)$  of a polynomial with multilinear coefficient functions is the convex hull of the images of the vertices of  $Q$ .

□

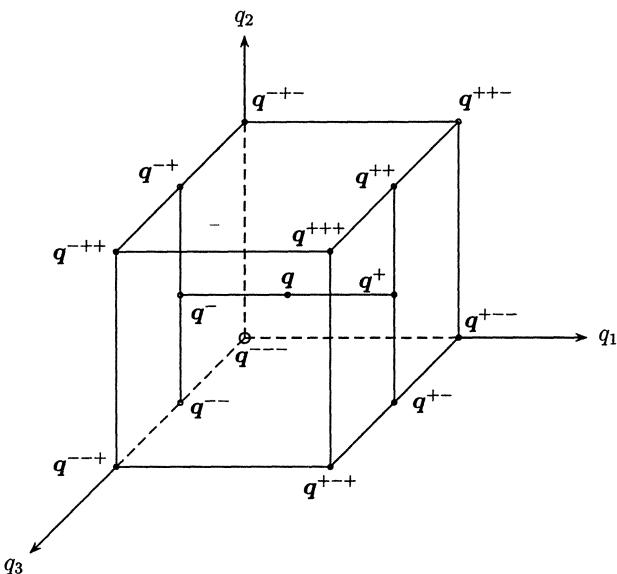


Fig. 5.13: Notations used in the proof of Theorem 5.4

*Proof.*

Without loss of generality let  $q_i \in [0; 1]$ ,  $i = 1, 2, \dots, \ell$ . We will demonstrate the proof of the mapping theorem for three parameters  $q_1, q_2, q_3$ . The generalization to  $\ell$  parameters is obvious (induction over  $\ell$ ). In four steps we show that the image of an interior point  $p$  of the  $Q$ -box is contained in the convex hull of the

images of the eight vertices (refer to Fig. 5.13). Let  $\text{conv}\{p_1, p_2, \dots, p_m\}$  denote the convex hull of  $m$  points that is the set

$$\begin{aligned} \text{conv}\{p_1, p_2, \dots, p_m\} := \\ \{p \mid p = \sum_{i=1}^m \lambda_i p^{(i)}, \quad \sum_{i=1}^m \lambda_i = 1, \quad \lambda_i \geq 0, \quad i = 1, 2, \dots, m\} \end{aligned} \quad (5.5.1)$$

The polynomials  $p^{(i)} = p(\mathbf{q}^{(i)})$  are generated by the corresponding vectors  $\mathbf{q}^{(i)}$  as indicated in Fig. 5.13. For fixed  $q_2 = q_2^*$  and  $q_3 = q_3^*$  and  $q_1 \in [0; 1]$  we have

$$p(j\omega^*, q_1, q_2^*, q_3^*) = (1 - q_1)p^- + q_1p^+ \quad (5.5.2)$$

where  $p^- = p(j\omega^*, 0, q_2^*, q_3^*)$ ,  $p^+ = p(j\omega^*, 1, q_2^*, q_3^*)$ . Using the notation of (5.5.1), (5.5.2) may be written as

$$p \subset \text{conv}\{p^-, p^+\} \quad (5.5.3)$$

Analogously, we have

$$p^- \subset \text{conv}\{p^{--}, p^{-+}\}, \quad p^+ \subset \text{conv}\{p^{+-}, p^{++}\} \quad (5.5.4)$$

The third step is combining (5.5.3) and (5.5.4):

$$p \subset \text{conv}\{p^{--}, p^{-+}, p^{+-}, p^{++}\} \quad (5.5.5)$$

and the last step gives

$$p \subset \text{conv}\{p^{---}, p^{--+}, p^{--+}, p^{--+}, p^{---}, p^{--+}, p^{++-}, p^{+++}\} \quad (5.5.6)$$

□

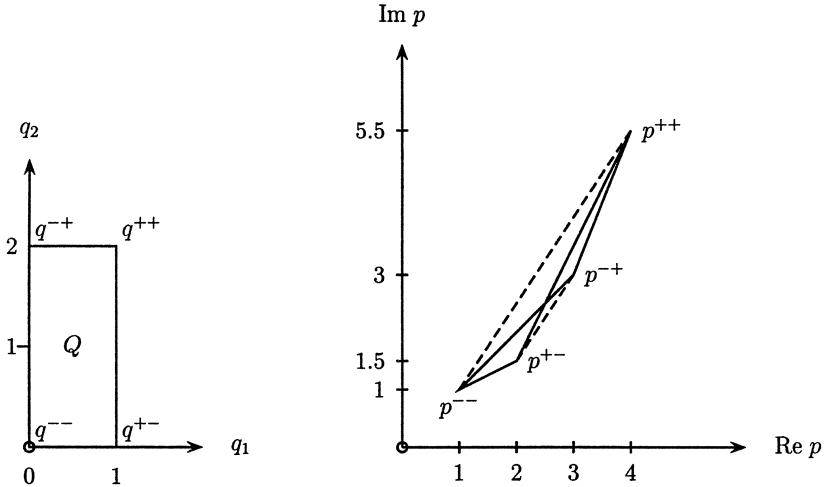
*Example 5.11.*

$$p(s, q_1, q_2) = (1 + q_1 + q_2) + (1 + 0.5q_1 + q_2 + q_1q_2)s \quad (5.5.7)$$

$$Q = \{q_1, q_2 \mid q_1 \in [0; 1], q_2 \in [0; 2]\} \quad (5.5.8)$$

With  $s = j\omega$  the images of the vertices of  $Q$  are

$$\begin{aligned} p^{++} &= p(j\omega, 1, 2) &= 4 + 5.5j\omega \\ p^{+-} &= p(j\omega, 1, 0) &= 2 + 1.5j\omega \\ p^{--} &= p(j\omega, 0, 0) &= 1 + j\omega \\ p^{-+} &= p(j\omega, 0, 2) &= 3 + 3j\omega \end{aligned}$$

Fig. 5.14: Mapping of the edges of  $Q$  and the convex hull of the value set

For  $\omega = 1$  the images of the four vertices of  $Q$  are shown in the  $p$ -plane of Fig. 5.14 as  $p^{++}, p^{+-}, p^{--}, p^{-+}$ .

The convex hull is the quadrangle  $p^{++}, p^{--}, p^{+-}, p^{-+}$ . Along the edges, i.e. for fixed  $q_1$  or fixed  $q_2$ ,  $p$  is linear in the varying parameter, i.e. the image of an edge of  $Q$  is a straight line segment connecting the images of the corresponding vertices of  $Q$ . The edges are plotted in solid lines in Fig. 5.14. Obviously the edges belong to the value set. If the image of a vertex belongs to the convex hull, as  $\overline{p^{--} p^{+-}}$ , then these segments belong to the boundary of the value set, as  $\overline{p^{++} p^{-+}}$  and  $\overline{p^{--} p^{++}}$ . The dotted lines  $\overline{p^{--} p^{++}}$  and  $\overline{p^{+-} p^{-+}}$  complete the convex hull of the value set.  $\square$

*Example 5.12.* The crane as given in (1.1.6) with  $\ell \in [8; 16] \text{ [m]}$ ,  $m_C \in [100; 2000] \text{ [kg]}$ ,  $m_L = 2000 \text{ [kg]}$  and  $g = 10 \text{ [ms}^{-2}\text{]}$  with the control law

$$u = -[500 \ 100 \ -100 \ 0] \mathbf{x}$$

has to be checked for Hurwitz stability. Its characteristic polynomial is

$$p(s, m_C, \ell) = 5000 + 1000s + (20100 + 500\ell + 10m_C)s^2 + 100\ell s^3 + \ell m_C s^4$$

We can verify stability easily for a nominal point in the given domain of interest, for instance  $\ell = 8 \text{ [m]}$  and  $m_C = 1000 \text{ [kg]}$ . The frequency is gridded and for each grid point  $s = j\omega^*$  the convex hull of the value set is constructed. The result is shown in Fig. 5.15. For higher frequencies (which are not shown for reasons of scale of the figure) the quadrangles move away from the origin and into the fourth quadrant. The frequency grid for this fourth degree polynomial can be terminated as soon as the entire convex hull lies in the fourth quadrant.

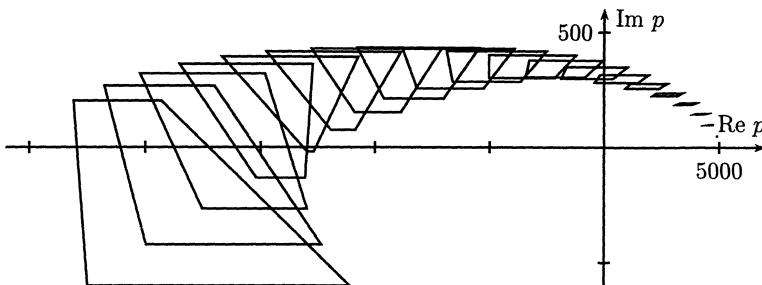


Fig. 5.15: Convex hulls for the given example with  $\omega^* = k \cdot 0.05$ ,  $k = 0, 1, \dots, 20$

Obviously the origin is excluded from the convex hull of the value sets at all frequencies. Thus, the origin is also excluded from the value sets at all frequencies and the system is stable.  $\square$

If the origin is included in the convex hull of the value sets for a frequency band  $\omega \in \Omega$ , then the mapping theorem does not give a conclusive answer about the stability of the polynomial family. A necessary and sufficient stability condition is then obtained by the actual construction of the value set for a grid on  $\omega \in \Omega$  and checking for zero exclusion from the value set. For frequencies  $\omega \notin \Omega$ , the value set need not be constructed in view of the sufficient condition obtained from the convex hull of the value set.

An interesting use of the mapping theorem is in domain splitting algorithms [53, 151, 152]. The idea of domain splitting is introduced by the following example.

*Example 5.13.* Consider the characteristic polynomial

$$p(s, q_1, q_2) = 3 + 2s + (0.25 + 2q_1 + 2q_2)s^2 + 0.5(q_1 + q_2)s^3 + q_1 q_2 s^4$$

with uncertain parameters  $q_i \in [1; 5]$ ,  $i = 1, 2$ . The vertex polynomials for  $s = j$  (i.e.  $\omega = 1$ ) yield

$$\begin{aligned} p^{--} &= p(j, q_1^-, q_2^-) = -0.25 + j \\ p^{-+} &= p(j, q_1^-, q_2^+) = -4.25 - j \\ p^{++} &= p(j, q_1^+, q_2^+) = 7.75 - 3j \\ p^{+-} &= p(j, q_1^+, q_2^-) = -4.25 - j \end{aligned}$$

The convex hull of these complex points is displayed in Fig. 5.16. It contains the origin. The edges  $p^{--} p^{-+}$  (identical with  $p^{--} p^{+-}$ ) and  $p^{++} p^{-+}$  (identical with  $p^{++} p^{+-}$ ) are images of edges of the  $Q$ -box, i.e. they are part of the boundary of the value set. The edge  $p^{--} p^{++}$  does not originate from an edge of the  $Q$ -box.

In a first step the operating domain is split into two subdomains  $Q_1 = \{q_1, q_2 \mid q_1 \in [1; 3], q_2 \in [1; 5]\}$  and  $Q_2 = \{q_1, q_2 \mid q_1 \in [3; 5], q_2 \in [1; 5]\}$ , see Fig. 5.18. The system is stable if and only if it is stable for both  $Q_1$  and  $Q_2$ . The convex hulls for both sets are constructed. The origin is now only included in  $\text{conv } \mathcal{P}(j, Q_1)$ . The resulting

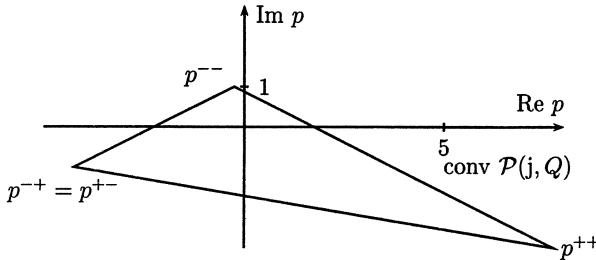
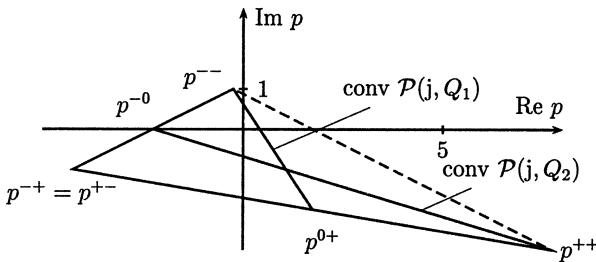
Fig. 5.16: Convex hull of the value set for  $s = j$ 

Fig. 5.17: Convex hulls of the value sets after the first partitioning step

convex hulls are displayed in Fig. 5.17 with the hull from the preceding step shown in dashed lines.

$Q_1$  is partitioned again into  $Q_3 = \{q_1, q_2 \mid q_1 \in [1; 3], q_2 \in [1; 3]\}$  and  $Q_4 = \{q_1, q_2 \mid q_1 \in [1; 3], q_2 \in [3; 5]\}$ , as shown in Fig. 5.18. The convex hulls produced by the subdomains  $Q_2$ ,  $Q_3$  and  $Q_4$  no longer contain the origin. The union of the convex hulls is displayed in Fig. 5.19.

□

Parameters entering polynomially into the coefficients of the characteristic polynomial can be overbounded by substitution, for example  $q_1^2 = q_1 q_1^*$ , and pretending that  $q_1$  and  $q_1^*$  vary independently in the interval  $[q_1^-; q_1^+]$ . Details and algorithms for the domain splitting approach can be found in [53, 151, 152]. In addition to the outer bound of the value set (sufficient condition) the algorithms also generate an inner bound (necessary condition) from the fact that images of the additional edges must belong to the value set. In Example 5.13 these are the line segments from  $p^{0-}$  to  $p^{0+}$  and from  $p^{-0}$  to  $p^{00}$ . The algorithm stops when either a sufficient condition for zero exclusion is satisfied or a necessary condition for zero exclusion is violated.

### Summary

In Chapter 5 several useful testing sets were introduced. For interval polynomials of degree one and two positivity of the coefficients guarantees stability, for higher degrees we have the following testing sets  $Q_T$

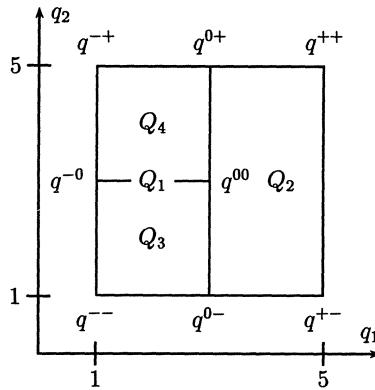
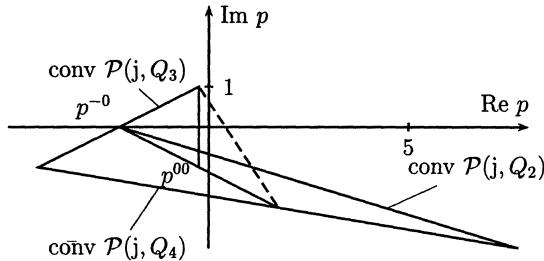
Fig. 5.18: Partitioning of the uncertainty domain  $Q$ 

Fig. 5.19: Convex hulls of the value sets after the second partitioning step

$n$	$Q_T$
3	$p^{+-}$
4	$p^{+-}, p^{++}$
5	$p^{+-}, p^{++}, p^{-+}$
6	$p^{+-}, p^{++}, p^{-+}, p^{--}$

For  $n > 6$  the number of polynomials to be tested does not increase. The test of four polynomials suffices for arbitrary  $n$ . For polynomial families with affine coefficient functions on a parameter box it suffices to check the edges of the  $Q$ -box for stability. The above tests are necessary and sufficient conditions stability tests for polynomial families. For the case of multilinear coefficient functions only a sufficient test is available, it is satisfied if the origin is excluded from the convex hull of the value sets  $\mathcal{P}(j\omega, Q)$  at all frequencies. The convex hull is spanned by the polynomials corresponding to the vertices of  $Q$ . In the two-parameter case (i.e. the bilinear case) it is feasible to calculate those interior points of  $Q$  that map into boundary points of the value set.

## 5.6 Exercises

5.1. Check the following uncertain polynomials for robust Hurwitz stability:

a)

$$p(s, q_1, q_2, q_3) = (1 - q_1) + (-3 + q_2)s - q_3 s^2 - s^3$$

with  $q_1 \in [2 ; 3]$ ,  $q_2 \in [1 ; 2]$  and  $q_3 \in [3 ; 4]$ .

b)

$$p(s, q) = (3 - q) + (2 + 3q)s + (5 - 2q)s^2 + (3 + q)s^3$$

with  $q \in [0 ; 2]$ .

c)

$$p(s, q_1, q_2, q_3) = (3 + q_1) + (4 - q_2)s + 21s^2 + (6 + 2q_3)s^3 + s^4$$

with  $q_1 \in [-1 ; 1]$ ,  $q_2 \in [0 ; 2]$  and  $q_3 \in [1 ; 2]$ .

d)

$$p(s, q_1, q_2) = (1 + 8q_1) + (2 + q_1 + q_2)s + (2 + q_1)s^2 + s^3$$

with  $q_1 \in [0 ; 4]$  and  $q_2 \in [0 ; 1]$ .

e)

$$p(s, q_1, q_2) = (6 + q_1 + q_2) + (5 + q_1 + 4q_2)s + 5s^2 + (1 + q_1)s^3 + 0.5(1 - q_2)s^4$$

with  $q_1 \in [-0.4 ; 0.4]$  and  $q_2 \in [-0.7 ; 0.7]$ .

5.2. Consider Example 5.9 with  $\ell = 2$  and  $r = 0.1$ .

a) Plot the stable and unstable regions in the  $(q_1, q_2)$ -plane.

b) Let  $q_1 \in [0 ; 1]$ ,  $q_2 \in [0 ; 1]$ . Which interior points map to the boundary of the value set?

5.3. Consider the polynomial family  $P(s, Q) = \{ p(s, q_1, q_2, q_3) = \prod_{i=1}^3 (s - q_i) \mid q_i \in [-\sqrt{3} ; \sqrt{3}] \}$ . Plot the convex hull of the value set for  $\omega^* = 0.5$ .

5.4. Consider the characteristic polynomial (4.4.2) of Example 4.8. The polynomial family is represented by

$$p(s, q_1, q_2) = p_0(s) + q_1 p_1(s) + q_2 p_2(s) \quad (5.6.1)$$

with

$$p_0(s) = 183 + 50s + 135s^2 + 52s^3 + 40s^4$$

$$p_1(s) = -16 + 24s - 12s^2 + 4s^3$$

$$p_2(s) = -21 + 42s - 21s^2$$

and  $q_1 \in [530 ; 540]$ ,  $q_2 \in [7575 ; 7750]$ . Check stability of the polynomial family by the edge theorem and compare your result with Fig. 4.7.

- 5.5. Let  $q_3 = 0$  in the characteristic polynomial of Example 4.14. Perform an edge test for  $q_1 \in [10 ; 30]$  and plot the corresponding eigenvalues of the Bialas matrices for a)  $q_2 \in [0 ; 3]$ , b)  $q_2 \in [0 ; 5]$ .

## 6 Value Set Construction

In the previous chapter the stability test by zero exclusion from the value set at all frequencies was used as a concept for proofs of Kharitonov's theorem and the edge theorem. For nonlinear parameter dependency there are no such simple results, it may be possible, however, to construct the value set and to use it for the stability test by zero exclusion. We will see that the construction of value sets can be performed extremely fast if the system has a so-called tree structure.

In Chapter 4 it was shown that an uncertain system with characteristic polynomial  $p(s, \mathbf{q})$  is robustly stable if and only if

- there exists a  $\mathbf{q}_0 \in Q$  for which  $p(s, \mathbf{q}_0)$  is stable and
  -
- the value sets  $\mathcal{P}(j\omega, Q)$  do not contain the origin for all frequencies  $\omega \in [0; \infty)$ .

One advantage of this approach is that even a high-dimensional operating domain  $Q$  is always mapped into the two-dimensional complex plane  $\mathbb{C}$ . Therefore, the construction of value sets is well suited for a graphical display. The sets can be visualized for various frequencies on a computer display and the user can investigate by visual inspection of the sets the stability of the uncertain polynomial. If the construction of the value sets is fast enough it is even possible to create a computer animation where the value sets are displayed on the screen with increasing frequency.

### 6.1 Sequential Set Operations

If we are interested in generating the exact value set for nonlinear parameter dependency, dense gridding of the uncertainty domain will often be the only possibility. In certain cases, however, the construction of the value sets can be simplified drastically if the characteristic equation has special properties.

*Example 6.1.* Consider the crane with the control structure as given in Example 2.9. The characteristic polynomial is by (2.3.7)

$$\tilde{p}(s) = (s^2\ell + g)[d(s)(k_1 + m_C s^2) + k_2 s] + d(s)s^2(m_L g - k_3) \quad (6.1.1)$$

The uncertain parameters are  $m_L$ ,  $m_C$ ,  $\ell$ ,  $k_1$ ,  $k_2$ , and  $k_3$ . We want to avoid gridding these six parameters as much as possible. Are there any other ways to generate the value set for a fixed frequency?

Before we start to generate the value set of (6.1.1) we look at an easier problem. Take the term  $k_1 + m_C s^2$  and construct its value set for fixed  $s = j\omega$ . This is quite simple:  $k_1$  ranges in a real interval from  $k_1^-$  to  $k_1^+$ ,  $m_C$  ranges from  $m_C^-$  to  $m_C^+$ . Multiplying  $m_C$  by  $s^2 = -\omega^2$  yields the real interval  $[-m_C^+ \omega^2; -m_C^- \omega^2]$ . Adding both intervals results in  $[-m_C^+ \omega^2 + k_1^-; -m_C^- \omega^2 + k_1^+]$ , which is again real. The next step is to generate the set for  $d(j\omega)(k_1 - m_C \omega^2) + k_2 j\omega$ . The polynomial  $d(j\omega)$  is a complex number for a fixed frequency. If we consider the set of  $k_1 - m_C \omega^2$  as a real line segment in the complex plane, a multiplication by  $d(j\omega)$  means a rotation and scaling of this set corresponding to phase and magnitude of  $d(j\omega)$ . The set of  $k_2 j\omega$  has to be added to this complex line segment. The result is shown in Fig. 6.1 with the vertices

$$p_1 = d(j\omega)(k_1^- - m_C^+ \omega^2) + k_2^- j\omega, \quad p_2 = d(j\omega)(k_1^+ - m_C^- \omega^2) + k_2^- j\omega$$

$$p_3 = d(j\omega)(k_1^+ - m_C^- \omega^2) + k_2^+ j\omega, \quad p_4 = d(j\omega)(k_1^- - m_C^+ \omega^2) + k_2^+ j\omega$$

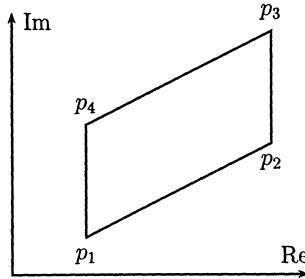


Fig. 6.1: Value set of  $d(s)(k_1 + m_C s^2) + k_2 s$  for a fixed frequency  $s = j\omega$

Similarly, the set of  $s^2\ell + g$  and  $d(s)s^2(m_L g - k_3)$  can be constructed easily. The last construction step is then the multiplication of the sets  $s^2\ell + g$  and  $d(s)(k_1 + m_C s^2) + k_2 s$ . The tools for the execution of this set multiplication will be introduced in the following sections. The main point is that the value set can be constructed sequentially. Much more effort would have been spent if the uncertain parameters had been densely gridded.  $\square$

The previous example showed that the construction is simplified significantly if the polynomial is nicely structured. This is the case if each uncertain parameter appears just once in the equation. Only then, the “subsets” (e.g.  $[-m_C^+ \omega^2 + k_1^-; -m_C^- \omega^2 + k_1^+]$  in Example 6.1) can be generated independently. In Chapter 8 it will be shown that

this structure is likely to appear in transfer functions generated from block diagrams where the uncertainties appear just in one block and nowhere else.

The additions of two-dimensional sets in the previous example are very simple. However, if the sets are more complicated, construction with paper and pencil will become difficult; multiplication of sets is even more complicated. For such difficult cases it would be useful to have an algorithmic description of complex set addition and multiplication. Then, these operations can be performed by a computer program. There already exist some algorithms [24, 81], however, they assume special parameter dependencies or they just work on convex value sets. For our purposes we do not want to approximate the resulting (in general nonconvex) value set by its convex hull. We are rather interested in an exact computation of the value set.

## 6.2 Simplification of Elementary Value Set Operations

For the set operations we assume all value sets to be closed and bounded. The complex set operations we are interested in are addition and multiplication. They are defined as

$$\mathcal{C} = \mathcal{A} + \mathcal{B} = \{a + b \mid a \in \mathcal{A}, b \in \mathcal{B}\} \quad (6.2.1)$$

$$\mathcal{C} = \mathcal{A} \cdot \mathcal{B} = \{a \cdot b \mid a \in \mathcal{A}, b \in \mathcal{B}\} \quad (6.2.2)$$

Every point of set  $\mathcal{A}$  is added to (or multiplied by) each point of set  $\mathcal{B}$ . The following shows how to simplify these operations.

Assume there exists an open neighborhood  $\mathcal{N}(a)$  in the set  $\mathcal{A}$  for a point  $a \in \mathcal{A}$ . If this open set  $\mathcal{N}(a)$  is multiplied by any point  $b \in \mathcal{B}$  this will again result in an open set. Therefore, points in the set  $\mathcal{C} = \mathcal{A} \cdot \mathcal{B}$  for which no open neighborhood in the set  $\mathcal{C}$  exists can only stem from points  $a \in \mathcal{A}$  and  $b \in \mathcal{B}$  which also do not have an open neighborhood in the sets  $\mathcal{A}$  and  $\mathcal{B}$ . These are boundary points,  $a \in \partial\mathcal{A}$  and  $b \in \partial\mathcal{B}$ , thus,

$$\partial\mathcal{C} \subset \partial\mathcal{A} \cdot \partial\mathcal{B} \quad (6.2.3)$$

Similarly, if  $\mathcal{C} = \mathcal{A} + \mathcal{B}$  then

$$\partial\mathcal{C} \subset \partial\mathcal{A} + \partial\mathcal{B} \quad (6.2.4)$$

The main use for constructing value sets is to check the result for exclusion of the origin or any other point. For this purpose it suffices to construct the boundary of the set, for which (6.2.3) and (6.2.4) are exploited.

### Addition

As shown in (6.2.4) the boundary of the set  $\mathcal{C} = \mathcal{A} + \mathcal{B}$  is contained in the set  $\partial\mathcal{A} + \partial\mathcal{B} =: \mathcal{C}_A$ . There may exist points  $a \in \partial\mathcal{A}$  and  $b \in \partial\mathcal{B}$  such that  $c = a + b \notin \partial\mathcal{C}$ : also interior points of the set  $\mathcal{C}$  are generated by the operation  $\partial\mathcal{A} + \partial\mathcal{B}$ . The boundary of the set

$\mathcal{C}_A$  is not necessarily identical to the boundary of set  $\mathcal{C}$ . There may be points on  $\partial\mathcal{C}_A$  which have an open neighborhood in the set  $\mathcal{C}$ . We refer to this set of points as  $\mathcal{C}_{AI}$ . Hence,

$$\partial\mathcal{C}_A = \partial(\partial\mathcal{A} + \partial\mathcal{B}) = \partial\mathcal{C} \cup \mathcal{C}_{AI}, \quad \partial\mathcal{C} \cap \mathcal{C}_{AI} = \emptyset \quad (6.2.5)$$

*Example 6.2.* The value sets  $\mathcal{A}$  and  $\mathcal{B}$  shown in Fig. 6.2 have to be added. The result of

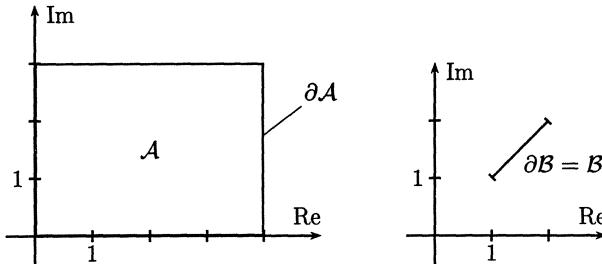


Fig. 6.2: Value sets  $\mathcal{A}$  and  $\mathcal{B}$

$\mathcal{C}_A = \partial\mathcal{A} + \partial\mathcal{B}$  is shown in Fig. 6.3.  $\mathcal{C} = \mathcal{A} + \mathcal{B}$  is displayed in gray, while  $\mathcal{C}_A = \partial\mathcal{A} + \partial\mathcal{B}$  is hatched. The set  $\mathcal{C}_{AI}$  is a boundary of the set  $\mathcal{C}_A$  but not of the set  $\mathcal{C}$ . The inner

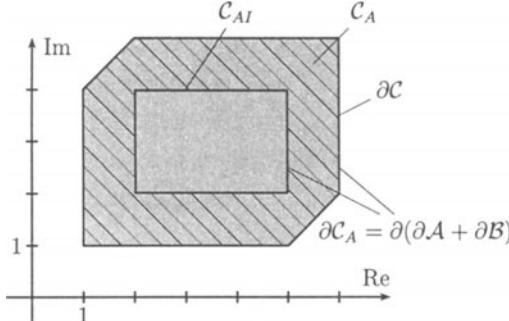


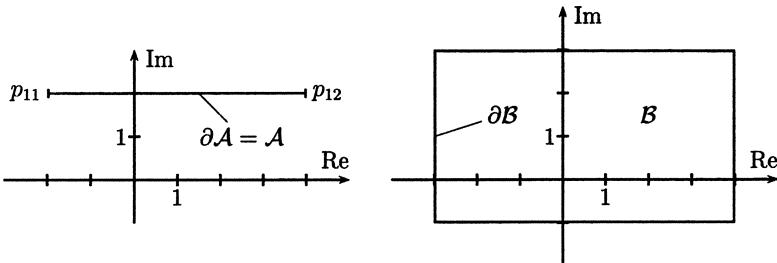
Fig. 6.3: Sum of value sets  $\partial\mathcal{A}$  and  $\partial\mathcal{B}$

rectangular part  $\mathcal{C}_{AI}$  of the boundary of  $\partial\mathcal{A} + \partial\mathcal{B}$  does not belong to  $\partial\mathcal{C}$ . □

### Multiplication

Just like for the addition, we are interested in the boundary of the value set. Analogous to the addition, the boundary of the product  $\mathcal{C}_M := \partial\mathcal{A} \cdot \partial\mathcal{B}$  may contain points with an open neighborhood in the set  $\mathcal{C}$  and therefore cannot be boundary points of  $\mathcal{C}$ . We refer to the set of these points as  $\mathcal{C}_{MI}$ :

$$\partial\mathcal{C}_M = \partial(\partial\mathcal{A} \cdot \partial\mathcal{B}) = \partial\mathcal{C} \cup \mathcal{C}_{MI}, \quad \partial\mathcal{C} \cap \mathcal{C}_{MI} = \emptyset \quad (6.2.6)$$

Fig. 6.4: Value sets  $\mathcal{A}$ ,  $\mathcal{B}$ 

*Example 6.3.* The sets  $\mathcal{A}$  and  $\mathcal{B}$  shown in Fig. 6.4 are to be multiplied. Fig. 6.5 shows the product of the boundaries  $\mathcal{C}_M = \partial\mathcal{A} \cdot \partial\mathcal{B}$ . The set  $\mathcal{C} = \mathcal{A} \cdot \mathcal{B}$  is displayed in gray, the set  $\mathcal{C}_M = \partial(\partial\mathcal{A} + \partial\mathcal{B})$  is hatched. The inner curved part  $\mathcal{C}_{MI}$  of the boundary of  $\partial\mathcal{A} \cdot \partial\mathcal{B}$  does not belong to  $\partial\mathcal{C}$ .

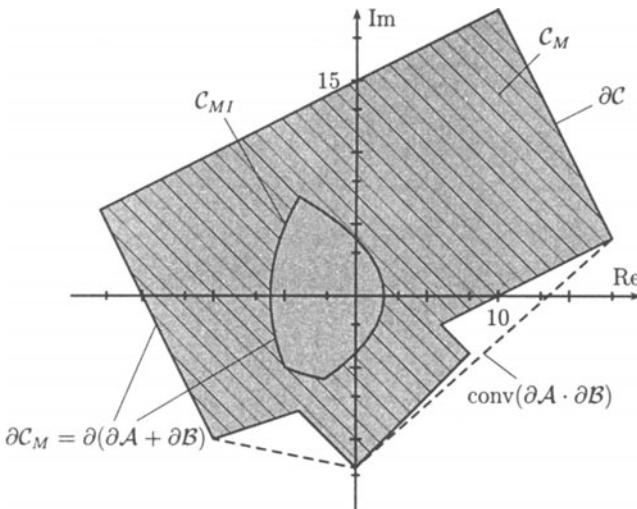


Fig. 6.5: Product of the boundaries of two complex sets

In Fig. 6.5 the convex hull of  $\mathcal{A} \cdot \mathcal{B}$  generated by the Mapping Theorem (Theorem 5.4) is shown with dashed lines.  $\square$

The above example showed that the resulting set  $\mathcal{C} = \mathcal{A} \cdot \mathcal{B}$  may be nonconvex though  $\mathcal{A}$  and  $\mathcal{B}$  are convex. The next example will show that the resulting set  $\mathcal{C} = \mathcal{A} \cdot \mathcal{B}$  may be not simply connected though the sets  $\mathcal{A}$  and  $\mathcal{B}$  are simply connected.

*Example 6.4.* Construct the value set of the polynomial

$$p(s, q) = \prod_{i=1}^3 (s + q_i) \quad (6.2.7)$$

for  $s = 0.5j$  and  $q_i \in [-\sqrt{3}; \sqrt{3}]$ . The value sets of the three subpolynomials  $s + q_i$  are straight lines in the complex plane parallel to the real axes, see Fig. 6.6. Thus, for

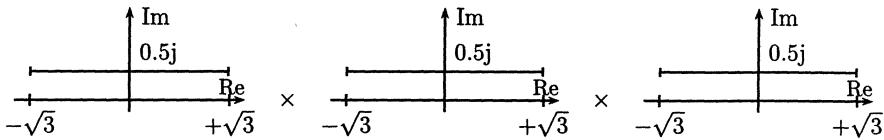


Fig. 6.6: Value sets of the subpolynomials

generating the set  $\mathcal{P}$  three line segments have to be multiplied with each other. The result is shown in Fig. 6.7. The boundary of the set is no longer simply connected. It is

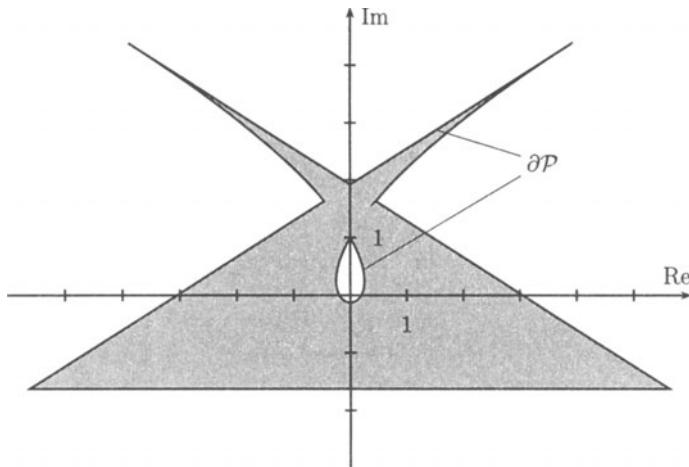


Fig. 6.7: Product of three straight lines

still an open question in which cases the interior boundary of the set may be neglected. One could think of cases where the system is stable, but its value set “wraps” around the origin for a frequency range without containing it. Value sets of the polynomial (6.2.7) for several frequencies are shown in Fig. 6.8.  $\square$

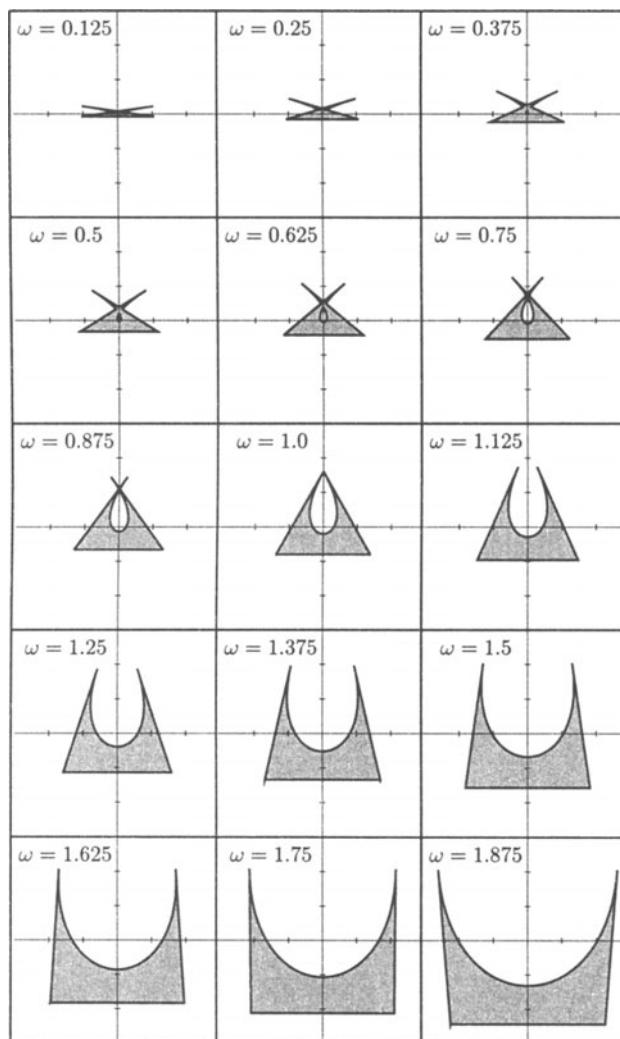


Fig. 6.8: Value sets of the uncertain polynomial (6.2.7)

### 6.3 Computer-Aided Execution of Value Set Operations

So far we were concerned with the simplification of set operations. It was shown that it suffices to operate with the boundaries to add or multiply two sets. If several successive operations have to be performed this is best done by a computer. The necessary algorithms have to be efficient and fast. Computer-aided construction of value sets always means an approximation of the actual set caused by numerical errors.

A first problem is a suitable way of representing the set boundaries in the computer. Here we could think of an analytic computation of the boundaries. Then, the boundary is represented by piecewise continuous and differentiable curves. If two sets are multiplied, not only the order of these curves will be increased but also the number of curve segments forming the boundary. For further operations it will become more and more complicated to determine the correct representation of the resulting boundaries.

For our purpose the boundaries are approximated by polygons. Example 6.2 and Example 6.3 showed that interior points are also generated with each operation. These have to be eliminated before the next set operation to get again a polygon representation of the resulting set boundary.

Before an operation  $\mathcal{A} + \mathcal{B}$  or  $\mathcal{A} \cdot \mathcal{B}$ , both boundaries of  $\mathcal{A}$  and  $\mathcal{B}$  have to be approximated by polygons. Both these polygons consist of line segments like the ones shown in Fig. 6.9.

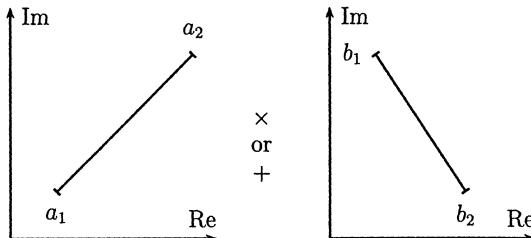


Fig. 6.9: Elementary set operations

For addition and multiplication each line segment representing a part of boundary  $\mathcal{A}$  is added to (or multiplied by) each line segment representing the boundary of  $\mathcal{B}$ . The result is the union of the sets obtained by these elementary operations, which are shown next.

#### Addition

The result of the addition of two line segments is a parallelogram like shown in Fig. 6.10. The vertices are  $c_{ij} = a_i + b_j$ ,  $i, j = 1, 2$ .

#### Multiplication

The line segments displayed in Fig. 6.9 have to be multiplied with each other. The line segments are described by  $a_1 + \alpha(a_2 - a_1)$ ,  $\alpha \in [0; 1]$  and  $b_1 + \beta(b_2 - b_1)$ ,  $\beta \in [0; 1]$ .

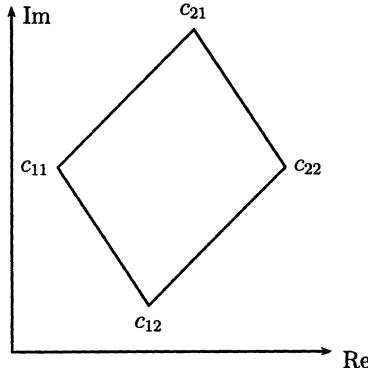


Fig. 6.10: Sum of two line segments

The product is then

$$\mathcal{C} = \{c(\alpha, \beta) \mid \alpha \in [0; 1], \beta \in [0; 1]\}$$

with

$$c(\alpha, \beta) = [a_1 + \alpha(a_2 - a_1)] \cdot [b_1 + \beta(b_2 - b_1)]$$

We can consider this as an uncertain multilinear term in  $\alpha$  and  $\beta$ . Thus, its value set can be constructed easily using the Jacobian (see Chapter 5). Besides the edges of the uncertainty domain, points for which the Jacobian determinant

$$J(\alpha, \beta) = \begin{vmatrix} \frac{\partial \operatorname{Re} c(\alpha, \beta)}{\partial \alpha} & \frac{\partial \operatorname{Re} c(\alpha, \beta)}{\partial \beta} \\ \frac{\partial \operatorname{Im} c(\alpha, \beta)}{\partial \alpha} & \frac{\partial \operatorname{Im} c(\alpha, \beta)}{\partial \beta} \end{vmatrix}$$

vanishes also contribute to the boundary of the resulting set. The resulting curve is a parabola which again is approximated by a polygon. If the Jacobian does not vanish for  $\alpha \in [0; 1]$  and  $\beta \in [0; 1]$ , then the resulting value set is convex. The vertices of this convex set are determined by the four points  $c_{ij} = a_i \cdot b_j$ ,  $i, j = 1, 2$ . Two possible forms of the result are shown in Fig. 6.11.

It is also possible to determine the product of two line segments without computing the Jacobian determinant. First, the four points  $c_{ij}$  are computed and connected in the sequence  $c_{11}-c_{12}-c_{22}-c_{21}-c_{11}$ . If the resulting set is convex, then the final result is already obtained. If the resulting set is non-convex, then one of the line segments has to be gridded and the other line segment is multiplied with these grid points. The result is shown in Fig. 6.12. The parabola generated by the Jacobian is approximated.

#### *Elimination of Interior Points*

For further set operations it is necessary to reduce the set to its boundary. One possibility are modified intersection algorithms as proposed in [141]. Another possibility is the following: The resulting polygons are transmitted to a graphics processor, which writes them into an internal, nonvisible raster display. Then, a contour algorithm determines

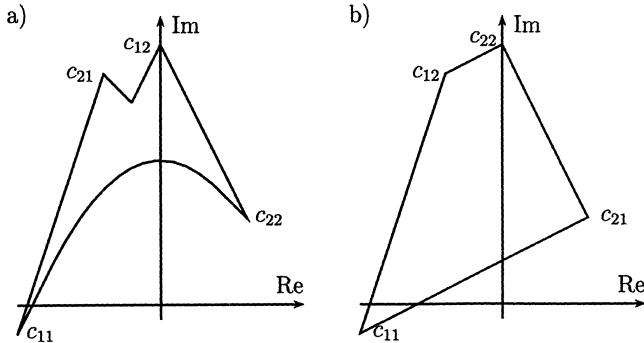


Fig. 6.11: Products of two line segments: a) the Jacobian determinant vanishes, b) the Jacobian determinant does not vanish, i.e. the set is convex

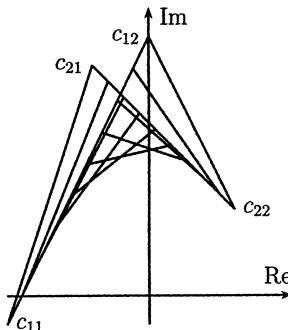


Fig. 6.12: Construction of the product of two line segments by approximation

the boundary of the graphically displayed and rasterized value set. The result is a list of pixel coordinates which can be reduced by eliminating pixels lying on a line. In Fig. 6.13 a part of the union of polygons and its rasterized representation is shown. The gray pixels represent pixels which were found by the contour algorithm. The algorithm first looks for a starting point. Then it searches successively neighboring pixels until it reaches the starting point again. In Fig. 6.13 the algorithm still needs to find four pixels (marked by x) until the starting point is reached and the contour is closed.

## 6.4 Color Coding of Value Sets

The construction of value sets gives important information about the system. However, one cannot tell for instance which parameter combinations yield the point in the set with the smallest distance from the origin. For a controller design the knowledge of these critical plants is very useful. A possibility to identify critical operating conditions

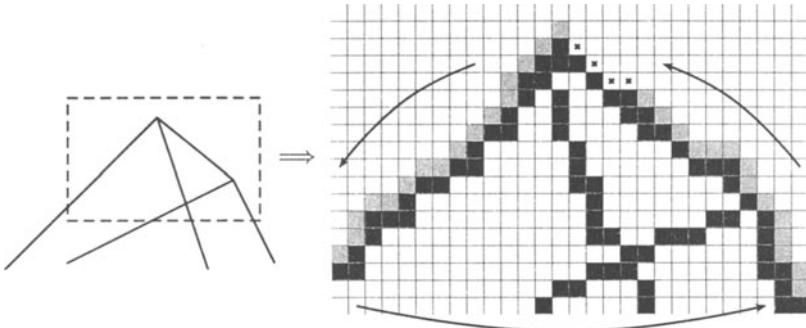


Fig. 6.13: Contour algorithm

is color coding the value set. In the case of two parameters a color is assigned to each vertex point of the selected rectangular operating plane. Then, for each interior point the corresponding color can be interpolated. The colored operating domain is then mapped into the complex plane via the characteristic polynomial.

*Example 6.5.* Consider the bus O 305 with the data given in Table 1.3. The characteristic polynomial of the yaw dynamics is

$$p(s, \tilde{m}, v) = 270 \cdot 10^9 + 16670\tilde{m}v^2 + 1.08 \cdot 10^6\tilde{m}vs + \tilde{m}^2v^2s^2 \quad (6.4.1)$$

The uncertain parameters velocity  $v$  and virtual mass  $\tilde{m}$  vary in the intervals  $v \in [3; 20] \text{ [m} \cdot \text{s}^{-1}]$  and  $\tilde{m} \in [9950; 32000] \text{ [kg]}$ . The yaw dynamics is stable for positive velocities, independent of parameter values for velocity and virtual mass: The coefficient functions are always positive which is necessary and sufficient for stability for this second order system. In the stability analysis it should be checked if the maximal real part of the eigenvalues of the yaw rate is smaller than  $-0.8$ . Now, the question of robust stability is no longer as easy to answer. The complex frequency  $s$  in (6.4.1) is replaced by  $-0.8 + j\omega$  and for a fixed frequency  $\omega = \omega^*$  the value set can be constructed. The influence of the different operating conditions is illustrated by color coding.

For the color coding each vertex of the uncertainty domain is characterized by a color, for instance the color green indicates minimal velocity and minimal virtual mass. A grid is overlaid and this colored grid displayed in Color plate 1 (at the end of this book) is mapped into the complex plane via the characteristic polynomial (6.4.1). Some color coded sets are shown in Color plate 2. The value set is not only bounded by the images of the edges of the  $Q$ -box, also interior points contribute to this boundary. For small frequencies especially the operating point associated with color blue (high virtual mass, high velocity) is critical, for higher frequencies the operating points with colors green and yellow (low velocity) are critical. For the constructed value sets, zero is excluded. Further constructions of value sets at other frequencies will yield robust stability of the yaw rate for the given operating domain, i.e. the eigenvalues of the yaw rate for the entire operating domain are located to the left of a straight line parallel to the imaginary axis with real part  $-0.8$ .

Interestingly, the value set hardly changes its form with increasing frequency. Viewed from the origin, the part of the boundary of the value set which is produced by interior points of the operating domain always lies on the opposite side of the set. Therefore, in this example, a possible instability would occur on one of the edges.

□

## 6.5 Tree Structured Decomposition

The concept of the tree structure of a polynomial for value set construction was already introduced with Example 6.1. The structure of the characteristic polynomial can also be displayed graphically as a tree. Therefore, we speak of “Tree Structured Decomposition” (TSD). The tree of the uncertain polynomial (6.1.1) is shown in Fig. 6.14. The terms in the bottom line are called subpolynomials. In this simple example each subpolynomial depends on one uncertain parameter and the dependency is linear. In

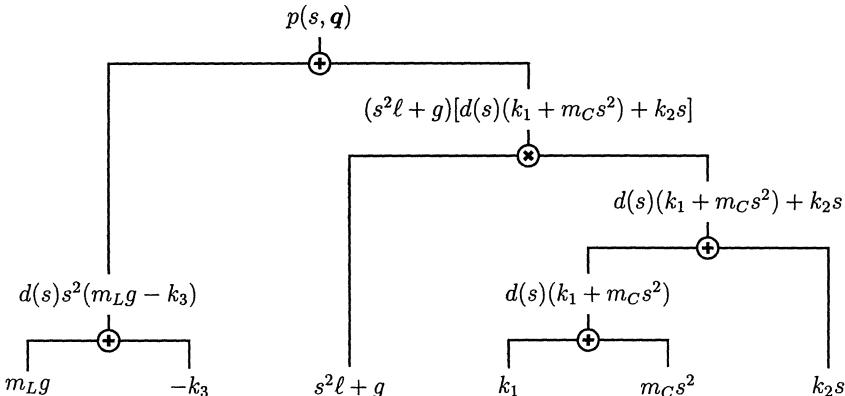


Fig. 6.14: Tree structure of the characteristic polynomial (6.1.1)

general, also multilinear and polynomial dependencies on several parameters may occur. The idea is to obtain subpolynomials with independent uncertainties and small numbers of uncertain parameters. Then, the value sets of these elementary polynomials can be constructed for even more complicated dependencies.

Which polynomials  $p(s, q)$  are structured like polynomial (6.1.1)? In the following, some definitions from [31] are introduced which allow us to define the tree structured decomposition of a polynomial.

The *index set*  $I$  of the  $\ell$  interval parameters is defined as

$$I := \{1, 2, \dots, \ell\} \quad (6.5.1)$$

The partition of an index set  $I$  is defined as

$$\begin{aligned} I_1 \cup I_2 &= I \\ I_1 \cap I_2 &= \emptyset \\ I_1 \neq \emptyset, I_2 \neq \emptyset \end{aligned} \tag{6.5.2}$$

Thus, every element of  $I$  belongs either to  $I_1$  or to  $I_2$ . The *subvector*  $\mathbf{q}^{I_j}$  consists of the components  $q_i$  with  $i \in I_j$ .

We call a polynomial  $p(s, \mathbf{q})$  *sum decomposable*, if there exists an index partition  $I_1, I_2$ , such that

$$p(s, \mathbf{q}) = p_1(s, \mathbf{q}^{I_1}) + p_2(s, \mathbf{q}^{I_2}) \tag{6.5.3}$$

A polynomial  $p(s, \mathbf{q})$  is *product decomposable*, if there exists an index partition  $I_1, I_2$ , such that

$$p(s, \mathbf{q}) = p_1(s, \mathbf{q}^{I_1}) \cdot p_2(s, \mathbf{q}^{I_2}) + p_0(s) \tag{6.5.4}$$

The polynomials  $p_1(s, \mathbf{q}^{I_1})$  and  $p_2(s, \mathbf{q}^{I_2})$  are called *subpolynomials* of  $p(s, \mathbf{q})$ . Further subpolynomials may arise if  $p_1(s, \mathbf{q}^{I_1})$  or  $p_2(s, \mathbf{q}^{I_2})$  are sum or product decomposable. A subpolynomial is called *undecomposable* if neither a sum nor a product decomposition is possible. The uncertain polynomial  $p(s, \mathbf{q})$  is *k-decomposable*, if a sequence of decompositions exists, such that the undecomposable subpolynomials depend on at most  $k$  components  $q_i$ . For  $k = 1$  the polynomial is called *totally decomposable*.

*Example 6.6.*

$$p(s, \mathbf{q}) = (q_1 q_2 + q_3 + q_3^2) + (q_1 - 4q_2 + q_3 + 3)s \tag{6.5.5}$$

In the first step a sum decomposition can be performed which separates the parameters  $q_1$  and  $q_2$  from  $q_3$ . It is

$$p(s, \mathbf{q}) = p_1(s, q_1, q_2) + p_2(s, q_3)$$

with

$$\begin{aligned} p_1(s, q_1, q_2) &= q_1 q_2 + (q_1 - 4q_2)s \\ p_2(s, q_3) &= q_3 + q_3^2 + q_3 s + 3s \end{aligned}$$

Note that the sum decomposition is not unique with respect to the constant term  $3s$ . Any polynomial  $\tilde{p}(s)$  could be added to  $p_1(s, q_1, q_2)$  and subtracted from  $p_2(s, q_3)$ . The polynomial  $p_1(s, q_1, q_2)$  is further product decomposable by adding the term  $(-4s^2)$  and including  $(+4s^2)$  to  $p_0(s)$ , such that

$$p_1(s, q_1, q_2) = p_{11}(s, q_1) \cdot p_{12}(s, q_2) + p_0(s)$$

with

$$\begin{aligned} p_{11}(s, q_1) &= q_1 - 4s \\ p_{12}(s, q_2) &= q_2 + s \\ p_0(s) &= 4s^2 \end{aligned}$$

Hence,  $p(s, \mathbf{q})$  is totally decomposable. This example demonstrates that the degree of the subpolynomials in  $s$  may be higher than the degree of the polynomial itself. For the construction of value sets, larger powers of  $s$  are not difficult to handle, since they are merely complex numbers for a fixed frequency  $s = j\omega^*$ .  $\square$

*Decomposition of Polynomials [153]*

In many cases the tree structure is not obvious, e.g. if we deal with an expanded characteristic polynomial with a high number of uncertain parameters or the system order is too high to recognize a tree structure already while modelling. These cases require a suitable tool that brings to light the structures.

Suppose the polynomial is available in its structured form. For the computation of the value set of the polynomial the value sets of the subpolynomials are combined corresponding to the tree structure. It is important to note that in each step only one operation, addition or multiplication is executed. This means that the tree structure is not determined in one step, rather the operations are applied in several steps to the polynomial and to its subpolynomials until no further decompositions are possible. To regenerate such structures a structure test has to be performed.

First, the polynomial is checked for a sum decomposition. As defined a sum decomposition arises if it is possible to write the polynomial as a sum of two polynomials with disjoint index sets

$$p(s, \mathbf{q}) = p_1(s, \mathbf{q}^{I_1}) + p_2(s, \mathbf{q}^{I_2})$$

Two uncertainty indices  $i$  and  $j$  belong to different index sets  $I_1$  and  $I_2$  if they are not multiplicatively coupled, i.e. there is neither a direct multiplicative coupling  $q_i^{m_i} q_j^{n_j}$  nor an indirect multiplicative coupling via one or more intermediate terms, e.g.  $q_i^{m_i} q_k^{m_k} q_k^{n_k} q_j^{n_j}$ .

For polynomials with few uncertain parameters a sum decomposition can be recognized by inspection.

*Example 6.7.* Given is the uncertain polynomial

$$p(s, \mathbf{q}) = q_1^2 q_2 q_4 + q_3^2 q_6 + q_4 q_7 + q_8 + (q_1 q_4^2 + q_3 q_6^2) s + (q_2 q_4 q_5 + q_7) s^2 \quad (6.5.6)$$

We recognize that the uncertain parameters  $q_3$  and  $q_6$  are multiplied only with each other but not with any other uncertain parameters. Also, the parameter  $q_8$  can be split off. Thus, the polynomial has a sum decomposition with the index sets

$$I_1 = \{1, 2, 4, 5, 7\}, \quad I_2 = \{3, 6\}, \quad I_3 = \{8\} \quad \square$$

For polynomials with many uncertain parameters a systematic test can be executed by a computer algebra program. If a sum decomposable polynomial is written as a multivariate polynomial, for instance in the parameters  $\mathbf{q}^{I_1}$ , then the constant term of this polynomial contains the parameters  $\mathbf{q}^{I_2}$ , while the rest of the polynomial depends only on  $\mathbf{q}^{I_1}$ . The subvector  $\mathbf{q}^{I_1}$  is not known in advance, but it can be determined iteratively. First, the uncertain polynomial is considered as a polynomial in one of the uncertain parameters, say  $q_1$ .

$$p^{(1)}(s, \mathbf{q}) = p_0(s, q_2, \dots, q_\ell) + p_1(s, q_2, \dots, q_\ell) q_1 + p_2(s, q_2, \dots, q_\ell) q_1^2 + \dots$$

Let the index 1 belong to  $I_1$ . In the first iteration,  $I_1^{(1)}$  contains all indices occurring in  $p_1(s, q_2, \dots, q_\ell)$ ,  $p_2(s, q_2, \dots, q_\ell)$ , ..., and  $I_2^{(1)}$  contains all indices occurring in

$p_0(s, q_2, \dots, q_\ell)$ . In further iterations elements of  $I_2^{(1)}$  are shifted to  $I_1^{(1)}$ . For this purpose the polynomial is written as a multivariate polynomial  $p^{(2)}(s, \mathbf{q})$  in the uncertainties with index  $I_1^{(1)}$ ; the uncertainty indices occurring in the constant term form  $I_2^{(2)}$ , all others form  $I_1^{(2)}$ . This process is repeated until the index sets become stationary and disjoint, i.e.  $I_1^{(N+1)} = I_1^{(N)} =: I_1$  and  $I_2^{(N+1)} = I_2^{(N)} =: I_2$ . A sum decomposition exists if the index set  $I_2$  is not empty. The subpolynomial  $p_2(s, \mathbf{q}^{I_2})$  is the constant term of the polynomial  $p^{(N)}(s, \mathbf{q})$  and  $p_1(s, \mathbf{q}^{I_1})$  is formed by the remaining terms. The subpolynomial  $p_1(s, \mathbf{q}^{I_1})$  is not sum decomposable, the subpolynomial  $p_2(s, \mathbf{q}^{I_2})$  may be still further sum decomposable.

*Example 6.8.* For illustration, the sum decomposition of the polynomial (6.5.6) is evaluated by the algorithm. In the first step

$$p^{(1)}(s, \mathbf{q}) = (q_3^2 q_6 + q_4 q_7 + q_8 + q_3 q_6^2 s + (q_2^2 q_4 q_5 + q_7) s^2) + (q_4^2 s) \cdot q_1 + (q_2 q_4) \cdot q_1^2$$

The uncertain parameters in the linear and quadratic terms are  $q_2$  and  $q_4$ , i.e.  $I_1^{(1)} = \{1, 2, 4\}$ ,  $I_2^{(1)} = \{3, 4, 5, 6, 7, 8\}$ . Now, the polynomial  $p(s, \mathbf{q})$  is written as a multivariate polynomial in the uncertainties of  $I_1^{(1)}$ :

$$p^{(2)}(s, \mathbf{q}) = (q_3^2 q_6 + q_8 + q_3 q_6^2 s + q_7 s^2) + q_7 \cdot q_4 + q_5 \cdot q_4 \cdot q_2^2 + s \cdot q_4^2 \cdot q_1 + q_4 \cdot q_2 \cdot q_1^2$$

Here,  $I_1^{(2)} = \{1, 2, 4, 5, 7\}$  and  $I_2^{(2)} = \{3, 6, 7, 8\}$  and the multivariate polynomial in the uncertainties  $I_1^{(2)}$  is

$$p^{(3)}(s, \mathbf{q}) = (q_3^2 q_6 + q_8 + q_3 q_6^2 s) + s^2 \cdot q_7 + q_7 \cdot q_4 + q_5 \cdot q_4 \cdot q_2^2 + s \cdot q_4^2 \cdot q_1 + q_4 \cdot q_2 \cdot q_1^2$$

Now,  $I_1^{(3)} = \{1, 2, 4, 5, 7\}$  and  $I_2^{(3)} = \{3, 6, 8\}$  and the iteration has reached stationary index sets  $I_1 = I_1^{(3)} = I_1^{(4)} = \{1, 2, 4, 5, 7\}$  and  $I_2 = I_2^{(3)} = I_2^{(4)} = \{3, 6, 8\}$ . The subpolynomial  $p_2(s, \mathbf{q}^{I_2})$  is the constant term of the polynomial  $p^{(3)}(s, \mathbf{q})$ , i.e.

$$p_2(s, \mathbf{q}^{I_2}) = q_3^2 q_6 + q_8 + q_3 q_6^2 s$$

and the subpolynomial  $p_1(s, \mathbf{q}^{I_1})$  is

$$p_1(s, \mathbf{q}^{I_1}) = q_1^2 q_2 q_4 + q_4 q_7 + q_1 q_4^2 s + (q_2^2 q_4 q_5 + q_7) s^2$$

The subpolynomial  $p_1(s, \mathbf{q}^{I_1})$  cannot be further sum decomposed,  $p_2(s, \mathbf{q}^{I_2})$  can be further sum decomposed into

$$p_2(s, \mathbf{q}^{I_2}) = (q_3^2 q_6 + q_3 q_6^2 s) + (q_8)$$

The final sum decomposed form of the polynomial (6.5.6) is then

$$p(s, \mathbf{q}) = (q_1^2 q_2 q_4 + q_4 q_7 + q_1 q_4^2 s + (q_2^2 q_4 q_5 + q_7) s^2) + (q_3^2 q_6 + q_3 q_6^2 s) + (q_8)$$

□

If there does not exist a sum decomposition, then the polynomial is checked for a product decomposition.

Suppose the polynomial has the tree structure

$$p(s, \mathbf{q}) = p_1(s, \mathbf{q}^{I_1}) \cdot p_2(s, \mathbf{q}^{I_2}) + p_0(s) \quad (6.5.7)$$

Because of the term  $p_0(s)$  the polynomial  $p(s, \mathbf{q})$  cannot be factorized, i.e. written as a product of polynomials. We can get rid of  $p_0(s)$  if all partial derivatives of the polynomial with respect to the uncertain parameters  $q_i$  are formed:

$$\frac{\partial p(s, \mathbf{q})}{\partial q_i} = p_2(s, \mathbf{q}^{I_2}) \frac{\partial p_1(s, \mathbf{q}^{I_1})}{\partial q_i} \quad \text{if } i \in I_1 \quad (6.5.8)$$

and

$$\frac{\partial p(s, \mathbf{q})}{\partial q_i} = p_1(s, \mathbf{q}^{I_1}) \frac{\partial p_2(s, \mathbf{q}^{I_2})}{\partial q_i} \quad \text{if } i \in I_2 \quad (6.5.9)$$

A product decomposition exists if and only if the partial derivatives of the polynomial with respect to  $q_i$  can be factorized. From the factors, the subpolynomials  $p_1(s, \mathbf{q}^{I_1})$  and  $p_2(s, \mathbf{q}^{I_2})$  can then be determined. The factorization of the polynomials can be executed with computer algebra programs. The constant polynomial  $p_0(s)$  is found by subtracting the product of the two subpolynomials from the polynomial  $p(s, \mathbf{q})$ :

$$p_0(s) = p(s, \mathbf{q}) - p_1(s, \mathbf{q}^{I_1}) \cdot p_2(s, \mathbf{q}^{I_2})$$

Of course the index partitions  $I_1$  and  $I_2$  are not known in advance, but if there exists a product decomposition, then each of the factored forms of the above partial derivatives must yield the same index partition, i.e. an index partition can already be determined from the factored form of one partial derivative and then it has to be verified with the other derivatives.

*Example 6.9.* Given is the polynomial

$$p(s, q_1, q_2, q_3) = \sum_{i=0}^4 a_i(q_1, q_2, q_3) s^i$$

with

$$\begin{aligned} a_0 &= 7 + q_1 + 5q_2 + q_1q_2 - 15q_3 - 3q_1q_3 \\ a_1 &= -38 - 8q_1 + 2q_1^2 + 7q_2 + 2q_1^2q_2 + 10q_2^2 + 2q_1q_2^2 + 19q_3 + 8q_1q_3 - 6q_1^2q_3 \\ a_2 &= 4 + q_2 + 2q_3 + q_1q_3 + (2 - q_1 + 2q_1^2)(-7 + q_2 + 2q_2^2 + 5q_3) \\ a_3 &= -7 + q_2 + 2q_2^2 + 7q_3 - q_1q_3 + 2q_1^2q_3 \\ a_4 &= q_3 \end{aligned}$$

Does there exist a tree structure for this polynomial? A sum decomposition is not possible. This can be seen for instance from the terms  $q_1q_2$  and  $3q_1q_3$  in the constant coefficient.

To check for a product decomposition the partial derivatives of the characteristic polynomial with respect to  $q_1$ ,  $q_2$ , and  $q_3$ , are formed and a factorization is attempted.

$$\text{Factor } \frac{\partial p(s, \mathbf{q})}{\partial q_1} = [1 + (4q_1 - 1)s][1 + q_2 - 3q_3 + (q_2 + 2q_2^2 + 5q_3 - 7)s + q_3s^2] \quad (6.5.10)$$

The partial derivative with respect to  $q_1$  is factorizable. The first factor depends only on  $q_1$  while the second depends on  $q_2$  and  $q_3$ . If the polynomial has a product decomposition, then the index partition must be

$$I_{11} = \{1\}, \quad I_{12} = \{2, 3\} \quad (6.5.11)$$

Now we look at the other two derivatives:

$$\text{Factor } \frac{\partial p(s, \mathbf{q})}{\partial q_2} = [5 + q_1 + (2 - q_1 + 2q_1^2)s + s^2][1 + (4q_2 + 1)s] \quad (6.5.12)$$

$$I_{21} = \{1\}, \quad I_{22} = \{2\} \quad (6.5.13)$$

and

$$\text{Factor } \frac{\partial p(s, \mathbf{q})}{\partial q_3} = [5 + q_1 + (2 - q_1 + 2q_1^2)s + s^2][-3 + 5s + s^2] \quad (6.5.14)$$

$$I_{31} = \{1\}, \quad I_{32} = \emptyset \quad (6.5.15)$$

All derivatives allow a factorization and there does not exist any other index partition than the one we already assumed. Note that the index partitions (6.5.13) and (6.5.15) are not identical to (6.5.11), but the index sets are always subsets of the two sets in (6.5.11). The index sets indicate a product decomposition

$$p(s, \mathbf{q}) = p_1(s, q_1) \cdot p_2(s, q_2, q_3) + p_0(s)$$

In (6.5.10) the partial derivative with respect to  $q_1$  was formed. Hence, the factor in (6.5.10) which does not depend on  $q_1$  is  $p_2(s, q_2, q_3)$ . Equation (6.5.12) is the partial derivative of  $p(s, \mathbf{q})$  with respect to  $q_2$ . Therefore, the subpolynomial  $p_1(s, q_1)$  is the factor in (6.5.12) not depending on  $q_2$ . The subpolynomials can be determined now as

$$p_1(s, q_1) = 5 + q_1 + (2 - q_1 + 2q_1^2)s + s^2$$

and

$$p_2(s, q_2, q_3) = 1 + q_2 - 3q_3 + (q_2 + 2q_2^2 + 5q_3 - 7)s + q_3s^2$$

The constant term  $p_0(s)$  is obtained from

$$p_0(s) = p(s, q_1, q_2, q_3) - p_1(s, q_1) \cdot p_2(s, q_2, q_3) = 2 - 5s + 3s^2$$

The subpolynomial  $p_1(s, q_1)$  is undecomposable, because it only depends on one uncertain parameter, subpolynomial  $p_2(s, q_2, q_3)$ , however, is still further decomposable into a sum of two subpolynomials:

$$p_2(s, q_2, q_3) = [1 + q_2 + q_2(1 + 2q_2)s] + [-3q_3 + (5q_3 - 7)s + q_3s^2]$$

As a result,  $p(s, \mathbf{q})$  is totally decomposable. □

### Finding Tree Structures

The idea of tree structures is physically motivated. In control systems there exist uncertain parameters which act only locally on the system, for example the time constant of an actuator. Frequently, there exists a representation of the characteristic equation such that these parameters appear just once. In contrast, there are global parameters which act on the whole system, for example the oil temperature of a hydraulic system. Global parameters will enter at several places into the system equations. If these parameters are uncertain, the system will not have a tree structure. However, if we assume the number of global parameters to be low in comparison to the other uncertain parameters, gridding of the global parameters is possible. Then, the stability analysis can be performed by exploiting the tree structure which exists now for the remaining uncertain parameters.

Tree structures can be found in a large number of control system configurations. The question is how to recognize these structures. The conventional way of modelling hides tree structures by smearing uncertain parameters over the coefficients of the characteristic polynomial. A first advice is not to unnecessarily manipulate the system equations. For example in mechanical systems the differential equations are represented in the form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{x}} + \mathbf{D}(\mathbf{q})\dot{\mathbf{x}} + \mathbf{K}(\mathbf{q}) = \mathbf{u} \quad (6.5.16)$$

To arrive at a state space representation, this system of equations is premultiplied by the inverse of the mass matrix  $\mathbf{M}(\mathbf{q})$ . This distributes the elements in this matrix all over the system of equations. A better approach is to directly determine the Laplace transform of (6.5.16). The characteristic polynomial is then

$$p(s, \mathbf{q}) = \det(\mathbf{M}(\mathbf{q})s^2 + \mathbf{D}(\mathbf{q})s + \mathbf{K}(\mathbf{q}))$$

This determinant can be evaluated step by step and possible tree structures can be recognized more easily.

*Example 6.10.* In Fig. 6.15 a schematic representation of a mechanical system taken from [18] is given. All elements are assumed to be uncertain. Its differential equations

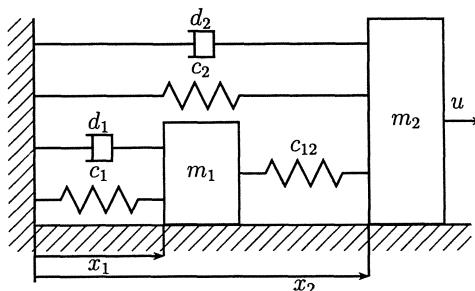


Fig. 6.15: Schematic representation of a mechanical system

are

$$\begin{aligned} m_1 \ddot{x}_1 + d_1 \dot{x}_1 + c_1 x_1 + c_{12}(x_1 - x_2) &= 0 \\ m_2 \ddot{x}_2 + d_2 \dot{x}_2 + c_2 x_2 + c_{12}(x_2 - x_1) &= u \end{aligned}$$

and their Laplace transform is

$$\begin{bmatrix} m_1 s^2 + d_1 s + c_1 + c_{12} & -c_{12} \\ -c_{12} & m_2 s^2 + d_2 s + c_2 + c_{12} \end{bmatrix} \begin{bmatrix} x_1(s) \\ x_2(s) \end{bmatrix} = \begin{bmatrix} 0 \\ u(s) \end{bmatrix}$$

The characteristic polynomial is

$$p(s, \mathbf{q}) = p_1(s, \mathbf{q}^{I_1}, c_{12}) \cdot p_2(s, \mathbf{q}^{I_2}, c_{12}) - c_{12}^2 \quad (6.5.17)$$

with

$$p_i(s, \mathbf{q}^{I_i}, c_{12}) = m_i s^2 + d_i s + c_i + c_{12}, \quad \mathbf{q}^{I_i} = [m_i \ d_i \ c_i]^T, \quad i = 1, 2 \quad (6.5.18)$$

The system does not have a tree structure because the uncertain parameter  $c_{12}$  appears both in  $p_1(s, \mathbf{q})$  and  $p_2(s, \mathbf{q})$ . If, however, this single parameter is assumed to be constant, then the system has a tree structure in the remaining six parameters. The parameter  $c_{12}$  has to be gridded and for each grid point the stability analysis can be done very fast by exploiting the tree structure. This is much better than gridding all seven uncertain parameters. For a given grid point  $c_{12} = c_{12}^*$  and  $s = j\omega^*$  the sets of  $p_i(s, \mathbf{q}^{I_i}, c_{12}^*), i = 1, 2$  are rectangles. These two rectangles have to be multiplied and  $c_{12}^2$  is subtracted. The value set of the characteristic polynomial (6.5.17) for  $m_1 \in [1; 3]$ ,  $d_1 \in [0.5; 2]$ ,  $c_1 \in [1; 2]$ ,  $m_2 \in [2; 5]$ ,  $d_2 \in [0.5; 2]$ ,  $c_2 \in [2; 4]$ , and  $c_{12}^* = 1$ , is given in Fig. 6.16 for  $s = j$ . For a stability analysis the value sets for all frequencies  $0 \leq \omega < \omega_{max}$  must be constructed and checked for zero exclusion. For the example,  $\omega_{max}$  is the frequency where the set is for the first time entirely contained in the fourth quadrant. Fig. 6.17 shows the union of value sets  $\mathcal{P}(j\Omega, Q)$ . Zero is not included in the union, thus, the system is robustly stable.  $\square$

*Remark 6.1.* A stability test of the above mechanical system is trivial. It is a passive system and therefore stable independent of the values for masses, springs, and dampers. In the case, however, one would like to check if the systems guarantees a certain stability margin, for example damping  $D > D_0$  for all  $\mathbf{q} \in Q$ , then the stability test is no longer simple, but the same tree structure can be exploited.  $\square$

*Remark 6.2.* In some cases, like in the previous example, it is necessary to grid a number of uncertain parameters in order to get a tree structured form of the characteristic polynomial. The number of the parameters to be gridded may become quite “large” such that the computation time even with the benefit of the tree structure is getting too high. In the case that parameters have to be gridded, they do not necessarily have to be gridded over their whole operating domain. Consider for example a polynomial which can be written in the form

$$p(s, \mathbf{q}) = p_1(s) \cdot p_2(s) + p_1(s) \cdot p_3(s) + p_2(s) \cdot p_3(s) + p_0(s)$$

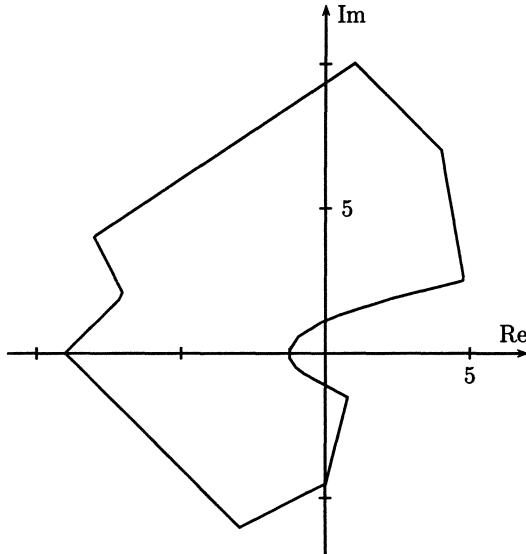


Fig. 6.16: Value set of the mechanical system for  $\omega = 1$

(For convenience the dependency of the polynomials of the uncertain parameters is dropped, i.e.  $p_1(s) = p_1(s, \mathbf{q}^{I_1})$ , etc.) We assume that the value sets of the polynomials  $p_1(s)$ ,  $p_2(s)$ , and  $p_3(s)$ , can be formed. The polynomial  $p(s)$  does not have a tree structure. A tree structure is available if one of the three uncertainty domains is gridded, for instance  $\mathbf{q}^{I_1}$ . Then, the polynomial  $p(s)$  can be written as

$$p(s) = (p_1(s) + p_3(s))(p_2(s) + p_3(s)) - p_3(s)^2 + p_0(s)$$

In this case it is not necessary to grid the entire operating domain of  $\mathbf{q}^{I_3}$ . For a fixed frequency  $\omega = \omega^*$  it suffices to grid the boundary of the value set  $\mathcal{P}_3(j\omega^*)$  and for each grid point  $\partial\mathcal{P}_3^*(j\omega^*)$  the set operation

$$(\mathcal{P}_1(j\omega^*) + \partial\mathcal{P}_3^*(j\omega^*))(\mathcal{P}_2(j\omega^*) + \partial\mathcal{P}_3^*(j\omega^*)) - \partial\mathcal{P}_3^*(j\omega^*) + p_0(j\omega^*)$$

has to be performed. This is sufficient as interior points of the set of  $p_3(j\omega^*)$  can never contribute to the boundary of the set of  $p(j\omega^*)$ . The final value set  $\mathcal{P}(j\omega^*)$  is obtained by the union of value sets for the grid points.  $\square$

## 6.6 Animation of Value Sets

The stability test by zero exclusion from the value set requires inspection of the value sets at all frequencies. In reality, only a finite number of frequency points are used. By

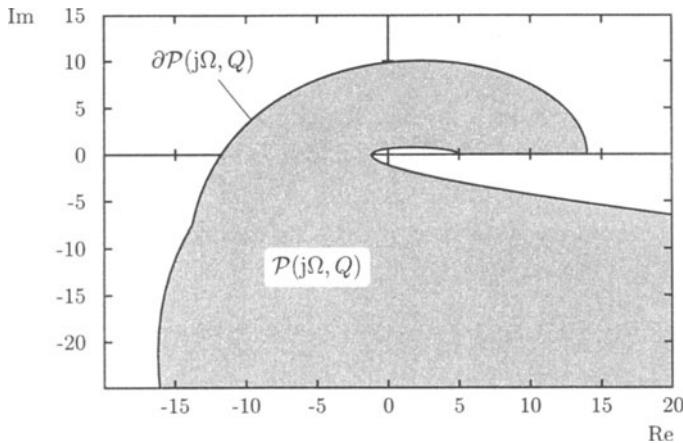


Fig. 6.17: Union of the value sets of the mechanical system

Mikhailov's theorem an upper frequency limit is given by the smallest frequency  $\omega_u$ , where the set is entirely contained in the  $n$ -th quadrant ( $n$  is the order of the system). If the computation of the value sets is fast enough they may be constructed even for a very large number of frequency points in a small amount of time. The test for zero exclusion may be done by an algorithm: If the number of intersections of the polygon with any half ray starting from the origin, for example the positive real axis, is odd, then zero is included. This algorithmic approach may give a yes/no answer about stability and in case of instability some frequency ranges, where zero is included in the set. Other important information, such as the distance of the set from the origin changing with the frequency is lost.

A better method is the graphical visualization of sets for increasing frequency. The engineer can watch a "film" of the animated set on the graphic display of his computer and thereby he performs the zero exclusion test by visual inspection of the set. Some scenes of the animated sequence of the value set of the mechanical system given in Example 6.10 are shown in Fig. 6.18. The animated value set not only gives a yes/no answer about robust stability, but it also shows critical frequency ranges for which the value set comes close to the origin and where better compensation is required. In Fig. 6.18, we recognize that for this example the frequency range  $\omega \in [0.5 ; 1.75]$  seems to be especially critical. In case of a controlled system, controller parameters can be modified during the animation with appropriate input devices (e.g. dial box, joy stick, steering ball) and the engineer can immediately recognize the consequences of this modification. With this interactive search in the space of controller parameters a robustly stabilizing controller can be quickly found [16].

## 6.7 Exercises

- 6.1. Given is the open-loop transfer function

$$g(s, \mathbf{q}) = \frac{6 + 4s + 6s^2}{(s - q_1)(s - q_2)(s - q_3)}$$

with  $q_i \in [-\sqrt{3}; \sqrt{3}]$ ,  $i = 1, 2, 3$ . Construct the value set for  $\omega = 0.5$  of the closed-loop system which is obtained with unit feedback. Compare with Fig. 6.7.

- 6.2. Construct the value set of the mechanical system given in Example 6.10 for  $s = j$  and compare with Fig. 6.16.
- 6.3. Write a Matlab program which produces the value set of the mechanical system for arbitrary frequencies. Create the union of all value sets for a dense frequency grid. Compare with Fig. 6.17.
- 6.4. What is the minimal degree of damping of the mechanical system given in Example 6.10?
- 6.5. Construct the product of the value sets in Fig. 6.4. Compare with Fig. 6.5.
- 6.6. Consider the uncertain characteristic polynomial (5.3.1). Does it admit a TSD
- a) for  $\ell = 2$
  - b) for  $\ell = 3$ ?
- 6.7. Consider the characteristic polynomial of the crane with state feedback. Does it admit a TSD for all seven parameters  $\ell, m_L, m_C, k_1, k_2, k_3$ , and  $k_4$ ?
- 6.8. Construct the value sets of the yaw dynamics of the city bus O 305 with the data given in Table 1.3 for some frequency points of your choice.
- 6.9. Consider the mechanical system given in Example 6.10 with  $m_1 \in [1; 3]$ ,  $d_1 \in [0.5; 2]$ ,  $c_1 \in [1; 2]$ ,  $m_2 \in [2; 5]$ ,  $d_2 \in [0.5; 2]$ ,  $c_2 \in [2; 4]$ , and  $c_{12}^* = 1$ . The position of the mass  $m_1$  has to be controlled by

$$u(s) = -f_C(s)x_1(s)$$

to increase damping of the system. A controller design for the center of the  $Q$ -box yields

$$f_C(s) = \frac{471250(0.5 + 1.9s + 1.7s^2 + s^3)}{19000 + 1450s + 62s^2 + s^3}$$

Is the closed loop system robustly stable for the entire operating domain? What is the maximal degree of damping which is achieved with this controller? Is it better than the minimal damping of the open loop?

The controller coefficients may vary now 20 % around their nominally given values. Is the system still stable then? What is the maximal possible variation in the controller coefficients? Try to solve these problems by constructing value sets.

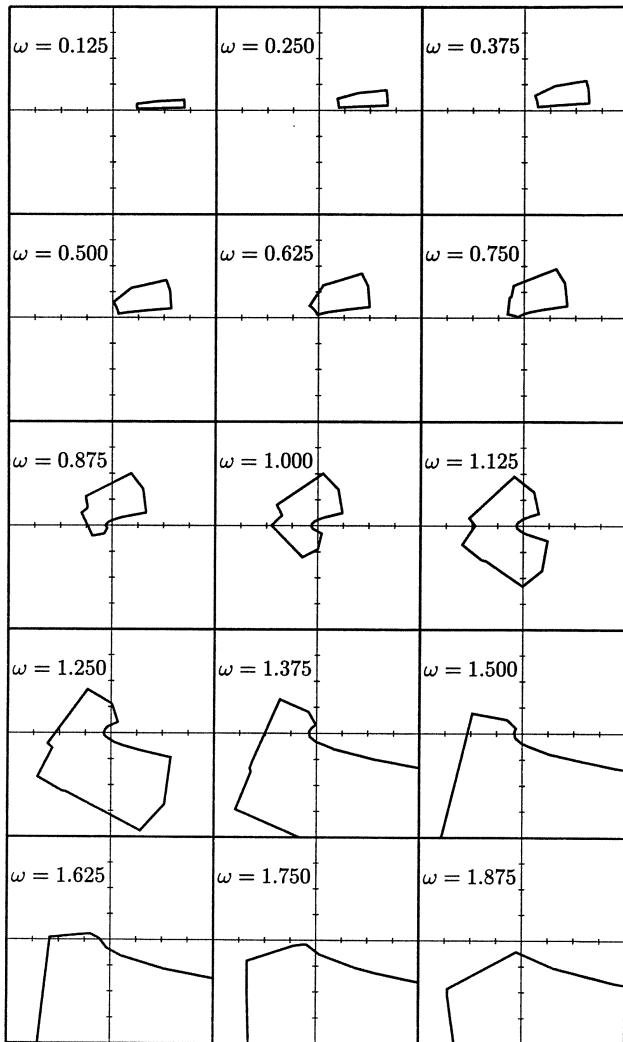


Fig. 6.18: Scenes from a value set animation

## 7 The Stability Radius

In Chapters 4 through 6 the stability of polynomial families

$$P(s, Q) = \{ p(s, \mathbf{q}) \mid \mathbf{q} \in Q \} \quad (7.0.1)$$

with  $p(s, \mathbf{q}) = a_0(\mathbf{q}) + a_1(\mathbf{q})s + \dots + a_n(\mathbf{q})s^n$  and  $q_i \in [q_i^-; q_i^+]$ ,  $i = 1, 2, \dots, \ell$  was investigated. The primary interest were necessary and sufficient conditions for stability. These conditions give primarily a yes or no answer to the robust stability problem. If the answer is yes, then there is some kind of stability margin of the uncertain parameters, the domain  $Q$  of the parameters can be enlarged without loosing stability. The next question is: What is the smallest perturbation in the parameters  $\mathbf{q}$  that destabilizes the system?

Assume that  $p(s, \mathbf{q})$  is stable for the center point  $\mathbf{q}^0$  of the  $Q$ -box. Then the polynomial family is stable in a sufficiently small neighborhood of  $\mathbf{q}^0$  because of the continuity of the roots. By shifting the origin of the parameter space it is always possible to have  $\mathbf{q}^0 = \mathbf{0}$ . The box centered at  $\mathbf{q}^0 = \mathbf{0}$  may now be blown up by a dilation factor  $\gamma$ , i.e. the polynomial family

$$P(s, \gamma Q) = \{ p(s, \mathbf{q}) \mid \mathbf{q} \in \gamma Q \} \quad (7.0.2)$$

is considered. By increasing  $\gamma$  a value  $\rho$  must be reached, where a member of the polynomial family becomes unstable, i.e. the box  $\gamma Q$  hits a stability boundary, this value  $\rho$  is called the *stability radius* of the polynomial family, it is a measure for the smallest destabilizing perturbation. For all  $\mathbf{q}$  with  $\|\mathbf{q}\|_\infty = \max_{i=1,2,\dots,\ell} |q_i| < \rho$  the polynomial family is stable. Note that the value  $\rho$  may be  $\infty$ , for example, consider  $p(s, q) = s + 1 + q^2$ .

The sidelength of the  $Q$ -box provides a scaling for the individual uncertain parameters  $q_i$ . After that scaling, a unit box  $Q_\infty = \{ \mathbf{q} \mid q_i \in [-1; 1] \}$  is enlarged by the dilatation factor  $\gamma$ . For computational simplicity a unit ball  $Q_2 = \{ \mathbf{q} \mid \sum_{i=1}^\ell q_i^2 = 1 \}$  may be dilated. The stability radius  $\rho_2$  is then defined as the smallest  $\gamma$  for which  $\gamma Q_2$  touches a stability boundary. The difficulty of finding the stability radius  $\rho$  depends again on the type of coefficients functions  $a_i(\mathbf{q})$ ,  $i = 0, 1, \dots, n$ . For affine dependency two methods are demonstrated. The case of polynomial dependency (which includes multilinear dependency) will be treated in the third section. There the problem is solved in theory but in applications only a small number of parameters can be handled.

## 7.1 Tsypkin-Polyak Loci

In Chapter 5 on the interval and affine cases, algebraic tests have been used to check the stability of testing sets. An alternative approach is the use of graphical frequency domain stability criteria. This approach is particularly useful if the largest stable  $Q$ -box around its stable center with coordinates

$$q_i^0 = \frac{q_i^+ + q_i^-}{2}, \quad i = 1, 2, \dots, \ell \quad (7.1.1)$$

has to be determined. A given  $Q$ -box with  $q_i \in [q_i^-; q_i^+]$  may then be described by

$$|q_i - q_i^0| \leq \alpha_i = \frac{q_i^+ - q_i^-}{2}, \quad i = 1, 2, \dots, \ell \quad (7.1.2)$$

A variable size of the uncertainty box can now be introduced by a common real dilation factor  $\gamma \geq 0$  for all uncertainties, i.e.

$$|q_i - q_i^0| \leq \gamma \alpha_i, \quad i = 1, 2, \dots, \ell \quad (7.1.3)$$

The case  $\gamma = 1$  coincides with the given box in (7.1.2). For  $\gamma < 1$  the box size is reduced and by increasing  $\gamma$  beyond 1 the uncertainty box is blown up. The  $\gamma$ -value for which the box around a stable center point (7.1.1) first hits the stability boundary is called the *stability radius*.

### Case 1: Interval Coefficients

Consider the polynomial family

$$P(s) = a_0 + a_1 s + \dots + a_n s^n \quad (7.1.4)$$

with the coefficients subject to the following constraints

$$|a_i - a_i^0| \leq \gamma \alpha_i, \quad i = 0, 1, \dots, n, \quad a_0^0 > 0 \quad (7.1.5)$$

This means that for each coefficient  $a_i$  there is a nominal value  $a_i^0$  and a scaling factor  $\alpha_i \geq 0$  for the coefficient perturbation.

If  $p^0(s) = a_0^0 + a_1^0 s + \dots + a_n^0 s^n$ ,  $a_0^0 > 0$ , is the nominal polynomial then

$$p^0(j\omega) = \operatorname{Re} p^0(j\omega) + j \operatorname{Im} p^0(j\omega) =: U(\omega) + j\omega V(\omega), \quad 0 \leq \omega < \infty \quad (7.1.6)$$

represents the usual Mikhailov curve. For testing the polynomial family, introduce the two scaling polynomials

$$S(\omega) = \alpha_0 + \alpha_2 \omega^2 + \alpha_4 \omega^4 + \dots \quad (7.1.7)$$

$$T(\omega) = \alpha_1 + \alpha_3 \omega^2 + \alpha_5 \omega^4 + \dots \quad (7.1.8)$$

and construct the frequency plot

$$z(\omega) = x(\omega) + jy(\omega) \quad (7.1.9)$$

with

$$x(\omega) := \frac{U(\omega)}{S(\omega)} \quad (7.1.10)$$

$$y(\omega) := \frac{V(\omega)}{T(\omega)} \quad (7.1.11)$$

Observe that the coefficients of the numerator polynomials only depend on the coefficients of the nominal polynomial, whereas the denominator coefficients depend on the scaling factors  $\alpha_i$ . If the  $\alpha_i$  are positive then the denominator polynomials are positive so that  $z(\omega)$  is finite for all finite  $\omega > 0$ . The boundary points  $z(0)$  and  $z(\infty)$  are

$$x(0) = \frac{a_0^0}{\alpha_0}, \quad (7.1.12)$$

$$y(0) = \frac{a_1^0}{\alpha_1} \quad (7.1.13)$$

and

$$x(\infty) = \begin{cases} \frac{a_n^0}{\alpha_n}, & n \text{ even} \\ \frac{a_{n-1}^0}{\alpha_{n-1}}, & n \text{ odd} \end{cases} \quad (7.1.14)$$

$$y(\infty) = \begin{cases} \frac{a_n^0}{\alpha_n}, & n \text{ odd} \\ \frac{a_{n-1}^0}{\alpha_{n-1}}, & n \text{ even} \end{cases} \quad (7.1.15)$$

*Theorem 7.1. (Tsypkin and Polyak Theorem for interval polynomials)*

For the stability of the polynomial family (7.1.4) and (7.1.5) it is necessary and sufficient that the following conditions hold for the plot  $z(\omega)$ :

1. It goes through  $n$  quadrants in counterclockwise direction for  $\omega \in [0; \infty)$ .
2. It does not intersect the square centered at the origin with sidelength  $2\gamma$ .
3. Its boundary points  $z(0), z(\infty)$  have coordinates with absolute values larger than  $\gamma$ .

□

Condition 2 requires that  $z(\omega)$  is outside the square. Condition 3 means that  $|x(0)| > \gamma$ ,  $|y(0)| > \gamma$ ,  $|x(\infty)| > \gamma$ ,  $|y(\infty)| > \gamma$ . Thus,  $z(0)$  and  $z(\infty)$  must lie in the regions I, II, III or IV of Fig. 7.1.

This theorem is a frequency domain version of Kharitonov's Theorem. The advantage compared to plotting four Mikhailov curves is that only one plot (even bounded if  $\alpha_0 \neq 0, \alpha_1 \neq 0, \alpha_{n-1} \neq 0, \alpha_n \neq 0$ ) is needed and the stability radius can be determined easily. When  $\gamma$  is enlarged until one of the conditions of the theorem is violated, then also the critical frequency is known, i.e. the frequency where the root set crosses the imaginary axis.

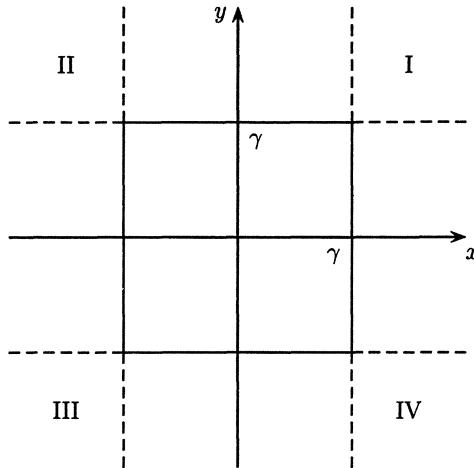


Fig. 7.1: The feasible domain for the Tsypkin-Polyak locus

*Remark 7.1.* Theorem 7.1 is a simplified version of the original theorem in [166]. Tsypkin and Polyak allow additionally other types of perturbations. For the considerable long proof the reader is referred to the original paper. A simpler proof was given by Mansour [122].  $\square$

*Example 7.1.* Recall the characteristic polynomial of Example 5.1. The polynomial family (5.1.36) has first to be scaled. With  $k_3^0 = -10000$ ,  $k_4^0 = 0$ ,  $\gamma = 1$  and

$$k_3 = -10000 + 10000q_1 \quad (7.1.16)$$

$$k_4 = 10000q_2 \quad (7.1.17)$$

the transformed polynomial family

$$p(s, q_1, q_2) = 3 + 10s + (18 + 5q_1)s^2 + (10 + 5q_2)s^3 + 5s^4 \quad (7.1.18)$$

must be stable in the unit square. The polynomials (7.1.6-7.1.8) are  $U(\omega) = 3 - 18\omega^2 + 5\omega^4$ ,  $V(\omega) = 10 - 10\omega^2$  and  $S(\omega) = T(\omega) = 5\omega^2$ . The frequency plot

$$z(\omega) = \frac{3 - 18\omega^2 + 5\omega^4}{5\omega^2} + j \frac{10 - 10\omega^2}{5\omega^2} \quad (7.1.19)$$

must go through four quadrants and avoid the unit square. The boundary conditions are satisfied because  $z(0) = \infty + j\infty$  and  $z(\infty) = \infty - 2j$ . The plot starts in the first quadrant and goes through four quadrants. Thus, the nominal polynomial for  $q_1 = q_2 = 0$  is stable, see Fig. 7.2.

To determine the stability radius the box is enlarged until it hits the Tsypkin-Polyak locus. From Fig. 7.2 it follows that the box touches the Tsypkin-Polyak plot for  $x(\omega) = y(\omega)$  i.e.  $5\omega^4 - 18\omega^2 + 3 = -10\omega^2 + 10$  or  $5\omega^4 - 8\omega^2 - 7 = 0$ . The only

real positive root  $\omega_m = 1.49$  leads to  $z_m = -1.1 - 1.1j$  and to the stability radius  $\rho = \gamma_{max} = 1.1$ .  $\square$

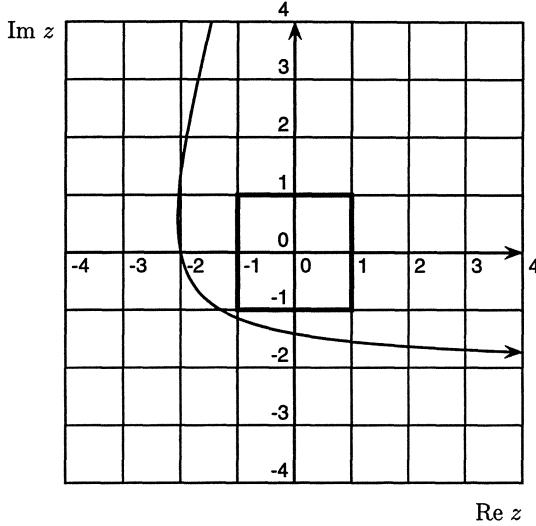


Fig. 7.2: The Tsypkin-Polyak locus

### Case 2: Affine coefficients

The next step is now the extension of the frequency domain approach to affine dependency of the polynomial coefficients. The polynomial family  $P(s, Q)$  can be rewritten in the form

$$\begin{aligned} p(s, q) &= p_0(s) + \sum_{i=1}^{\ell} q_i p_i(s) \\ p_0(s) &= a_0^0 + a_1^0 s + \dots + a_n^0 s^n, \quad a_n^0 > 0 \\ p_i(s) &= a_0^i + a_1^i s + \dots + a_n^i s^n, \quad i = 1, 2, \dots, \ell \end{aligned} \tag{7.1.20}$$

$q$  may vary in an  $\ell$ -dimensional box i.e.  $q_i \in [q_i^-; q_i^+]$ ,  $i = 1, 2, \dots, \ell$ . Shifting the origin and scaling the  $q_i$ -axes in the parameter space, the box  $Q$  is transformed to an  $\ell$ -dimensional cube with sidelength two and center at  $q = [0 \dots 0]^T$ , see (5.2.4). Let  $\omega_s$  be a common real zero of the rational functions

$$\text{Im}(p_k(j\omega)/p_0(j\omega)) = 0$$

for  $k = 1, 2, \dots, \ell$ . Now a real-valued function  $\tau(\omega)$  is defined by

$$\tau(\omega) = \max_{1 \leq k \leq \ell} \frac{|\text{Im}(p_0(j\omega)/p_k(j\omega))|}{\sum_{i=1}^{\ell} |\text{Im}(p_i(j\omega)/p_k(j\omega))|}, \quad 0 < \omega < \infty, \quad \omega \neq \omega_s \tag{7.1.21}$$

$$\tau(\omega) = \frac{|p_0(j\omega)|}{\sum_{i=1}^{\ell} |p_i(j\omega)|}, \quad \omega = \omega_s \quad (7.1.22)$$

$$\tau(0) = \frac{|a_0^0|}{\sum_{i=1}^{\ell} |a_i^0|}, \quad \tau(\infty) = \frac{|a_n^0|}{\sum_{i=1}^{\ell} |a_i^0|} \quad (7.1.23)$$

For  $\omega = 0$ ,  $\omega = \infty$  and  $\omega = \omega_s$  the function is in general discontinuous. The first two cases correspond again to roots at  $s = 0$  and  $s = \infty$ . The separate definition of  $\tau(\omega)$  for  $\omega_s$  is necessary because  $\tau(\omega)$  is discontinuous at the isolated frequencies  $\omega_s$ , a formal evaluation of  $\tau(\omega_s)$  with (7.1.21) would lead to 0/0.

*Theorem 7.2. (Tsypkin and Polyak Theorem for affine coefficients)*

The polynomial family  $P(s, Q)$  is stable, if and only if

1.  $p_0(s)$  is stable
2.  $\tau(\omega) > 1$ ,  $0 \leq \omega \leq \infty$ .

□

A slightly different form of this theorem, which resembles to the Mikhailov plot is

*Theorem 7.3. (Tsypkin and Polyak Theorem for affine coefficients)*

The polynomial family  $P(s, Q)$  is stable, if and only if

1.  $p_0(j\omega) \neq 0$
2.  $z(\omega) = \frac{p_0(j\omega)}{|p_0(j\omega)|} \tau(\omega)$  goes for  $0 \leq \omega < \infty$  through  $n$  quadrants and does not intersect the unit circle.

□

The difference between Theorem 7.2 and Theorem 7.3 is only the graphical representation. For a stability test by Theorem 7.2 the function  $\tau(\omega)$  is plotted. Its plot must be entirely above the line  $\tau = 1$ . In Theorem 7.3 a polar plot is generated, where  $\tau(\omega)$  is the distance from the origin and the phase angle is that of the nominal polynomial i.e.  $\frac{p_0(j\omega)}{|p_0(j\omega)|}$ .

*Remark 7.2.* The function  $\tau(\omega)$  can also be defined using trigonometric functions. But for computational purposes it is more convenient to have a nontrigonometric version. For the proof the reader again is referred to the original paper [167]. Using Theorem 7.2 or 7.3 beyond a yes or no answer, critical frequencies are recognized.  $\square$

What is the amount of calculation for this test? First, the frequency  $\omega$  has to be gridded and the polynomials  $p_0(s), p_i(s), i = 1, 2, \dots, \ell$  have to be evaluated at  $s = j\omega$ . Then  $\ell$  functions have to be calculated and their maximum has to be determined. The isolated frequencies  $\omega_s$  can be found by transforming  $\text{Im}(p_k(j\omega)/p_0(j\omega))$ . With  $p_k = R_k + jI_k$  and  $p_0 = R_0 + jI_0$  it is

$$\text{Im}(p_k(j\omega)/p_0(j\omega)) = \text{Im} \frac{R_k + jI_k}{R_0 + jI_0} = \frac{R_0 I_k - R_k I_0}{R_0^2 + I_0^2} = 0 \quad (7.1.24)$$

So  $\omega_s$  must be a common (real) root of the polynomials

$$\begin{aligned} R_0 I_1 - R_1 I_0 &= 0 \\ R_0 I_2 - R_2 I_0 &= 0 \\ &\vdots \\ R_0 I_\ell - R_\ell I_0 &= 0 \end{aligned} \quad (7.1.25)$$

*Example 7.2.* Recall the characteristic polynomial (4.4.2) of Example 4.8. The transformation

$$k_2 = 535 + 5q_1, \quad k_3 = 7662.5 + 87.5q_2$$

allows  $q_1$  and  $q_2$  to vary in the square of sidelength two. The polynomial family is

$$p(s, q_1, q_2) = p_0(s) + q_1 p_1(s) + q_2 p_2(s) \quad (7.1.26)$$

with

$$\begin{aligned} p_0(s) &= 129 + 166s + 237s^2 + 108s^3 + 80s^4 \\ p_1(s) &= -16 + 24s - 12s^2 + 4s^3 \\ p_2(s) &= -21 + 42s - 21s^2 \end{aligned}$$

At first a check is made for the existence of isolated frequencies. The polynomials (7.1.25) are

$$\begin{aligned} R_0 &= 129 - 237\omega^2 + 80\omega^4 \\ I_0 &= 166\omega - 108\omega^3 \\ R_1 &= -16 + 12\omega^2 \\ I_1 &= 24\omega - 4\omega^3 \\ R_2 &= -21 + 21\omega^2 \\ I_2 &= 42\omega \end{aligned}$$

and

$$\begin{aligned} R_0 I_1 - R_1 I_0 &= -4\omega(\omega^2 - 2)(80\omega^4 - 881\omega^2 + 719) \\ R_0 I_2 - R_2 I_0 &= 84\omega(\omega^2 - 2)(67\omega^2 - 53) \end{aligned}$$

Common real roots are  $\omega_s = 0$  (which is always the case) and  $\omega_s = \sqrt{2}$ .

$$\tau(\sqrt{2}) = \frac{|p_0(j\sqrt{2})|}{\sum_{i=1}^2 |p_i(j\sqrt{2})|} = \frac{R_0(\sqrt{2})}{R_1(\sqrt{2}) + R_2(\sqrt{2})} = \frac{I_0(\sqrt{2})}{I_1(\sqrt{2}) + I_2(\sqrt{2})} = 25/29$$

The test could be finished because this value is smaller than one and the polynomial family must be unstable. For completeness the other values of  $\tau(\omega)$  are calculated. They are

$$\begin{aligned} \tau(0) &= \frac{R_0(0)}{R_1(0) + R_2(0)} = 129/37 \\ \tau(\infty) &= \infty \end{aligned}$$

and otherwise

$$\begin{aligned} \tau_1(\omega) &= \frac{|\text{Im}(p_0(j\omega)/p_1(j\omega))|}{|\text{Im}(p_2(j\omega)/p_1(j\omega))|} \\ \tau_2(\omega) &= \frac{|\text{Im}(p_0(j\omega)/p_2(j\omega))|}{|\text{Im}(p_1(j\omega)/p_2(j\omega))|} \\ \tau(\omega) &= \max\{\tau_1(\omega), \tau_2(\omega)\} \end{aligned}$$

From Fig. 7.3 (the ordinate is logarithmically scaled) it can be seen that  $\tau(\omega) > 1$  for all  $\omega \neq \sqrt{2}$ . The minimum of the function  $\tau(\omega)$  is 25/29 for the frequency  $\omega = \sqrt{2}$ . The modulus of the components  $q_1$  and  $q_2$  must be smaller than 25/29. The corresponding intervals for the original feedback gains are then  $k_1 \in [530.7; 539.3]$  and  $k_2 \in [7587; 7738]$ . The parameter combination  $q_1 = q_2 = 25/29$  produces a polynomial with a root pair at  $s = j\sqrt{2}$ .

An alternative method is using Theorem 7.3. The magnitude of the complex-valued function  $z(\omega) = \frac{p_0(j\omega)}{|p_0(j\omega)|}\tau(\omega)$  is evaluated for all  $\omega$  and the infimum of this function determines the stability radius. This is equivalent to enlarging the unit circle in the complex plane until there is an intersection with the frequency plot  $z(\omega)$ . In Fig. 7.4 the alternative Tsyplkin-Polyak locus is plotted for determining the stability radius. The function  $z(\omega)$  is discontinuous at the isolated frequencies  $\omega_{s1} = 0$  and  $\omega_{s2} = \sqrt{2}$  (marked with  $\circ$ ). The continuous part avoids the unit circle but  $z(\sqrt{2})$  is inside the unit circle. Thus, the polynomial family is unstable and to get stability the operating domain of  $q_1$  and  $q_2$  must be reduced by a factor  $< z(\sqrt{2}) = 25/29$

□

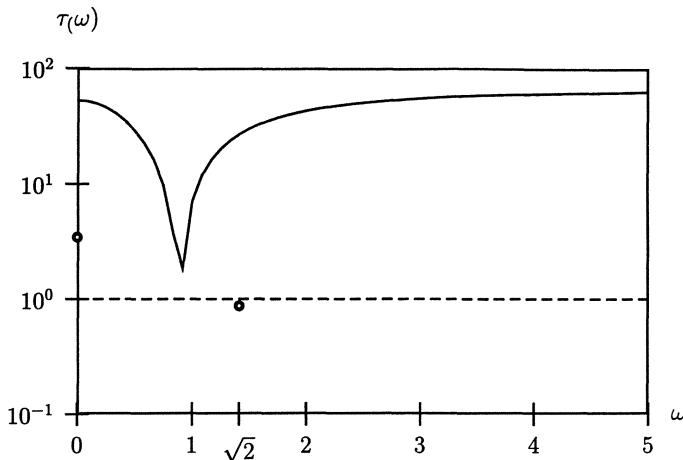


Fig. 7.3: The distance function  $\tau(\omega)$  for testing the stability.  $\tau(\omega)$  is  $> 1$  at the continuous part, but  $< 1$  at the isolated frequency  $\omega_s = \sqrt{2}$ , so the polynomial family is unstable

## 7.2 Affine Dependency: The Largest Hypersphere in Parameter Space

The evaluation of  $\tau(\omega)$  and the determination of the largest stable box is a rather expensive task. A more effective method is now presented for a closely related problem namely calculating the largest stable hypersphere in the space of parameters  $\mathbf{q}$ . This problem was first solved by Soh et al. [156]. A simpler solution was given by the authors [98], it is presented here.

Consider an uncertain polynomial  $p(s, \mathbf{q}) = [1 s \dots s^n] \mathbf{a}(\mathbf{q})$  with affine dependency of the coefficient vector  $\mathbf{a}$  on  $\mathbf{q}$ , i.e.

$$\mathbf{a} = \mathbf{a}^0 + \mathbf{F}\mathbf{q} \quad (7.2.1)$$

with  $\mathbf{a}^0 \in \mathbb{R}^{n+1}$  and  $\mathbf{F} \in \mathbb{R}^{(n+1) \times \ell}$ . Each element  $q_i$  of  $\mathbf{q}$  is scaled by a “high value”  $q_i^+$  and this scaling factor is included in the  $i$ -th column of  $\mathbf{F}$ . Then a hypersphere  $\sum_{i=1}^{\ell} q_i^2 = \text{const.}$  in  $\mathbf{q}$ -space is a reasonable measure for the neighborhood of  $\mathbf{q} = \mathbf{0}$ . The parameter vector  $\mathbf{q} = \mathbf{0}$  is the nominal point, i.e. the coefficient vector of the unperturbed polynomial is  $\mathbf{a}^0$ . Assume that  $p(s, \mathbf{0})$  is stable. For  $\ell = n + 1$  and  $\mathbf{F} = \mathbf{I}$  the polynomials are interval polynomials. The monic case ( $a_n = 1$ ) is also included if the last element of  $\mathbf{a}^0$  is one and the elements of the last row of  $\mathbf{F}$  are zero.

A geometric approach is used to determine the largest hypersphere. The knowledge of some fundamental definitions and formulas of higher-dimensional geometry is needed. In  $\mathbb{R}^3$  a plane is defined by

$$E = e_1 q_1 + e_2 q_2 + e_3 q_3 + e_0 = 0 \quad (7.2.2)$$

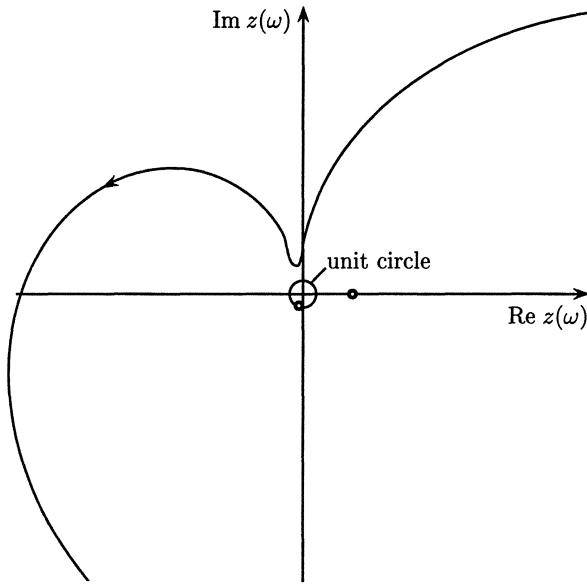


Fig. 7.4: The alternative Tsypkin-Polyak locus for testing the stability radius, it is discontinuous and does not avoid the unit circle at the isolated frequency  $\omega_{s2} = \sqrt{2}$ , so the polynomial family is unstable

The generalization to  $\mathbb{R}^\ell$  is the  $(\ell - 1)$ -dimensional hyperplane

$$E = e_1 q_1 + e_2 q_2 + \dots + e_\ell q_\ell + e_0 = 0 \quad (7.2.3)$$

The normal vector  $e (\neq 0)$  of a hyperplane is given by

$$e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_\ell \end{bmatrix} \quad (7.2.4)$$

Two hyperplanes are orthogonal if their normal vectors  $e_1, e_2$  are orthogonal that is  $e_1^T e_2 = 0$ . The squared distance  $d^2$  of the origin from the hyperplane is

$$d^2 = \frac{e_0^2}{e_1^2 + e_2^2 + \dots + e_\ell^2} = \frac{e_0^2}{e^T e} \quad (7.2.5)$$

Starting from the stable nominal point and varying  $q$  the Boundary Crossing Theorem states that there are three possibilities for the polynomial to become unstable:

- a) a real zero goes through the origin ( $s = 0$ )
- b) a real zero goes through infinity ( $s = \infty$ )

c) a pair of conjugate zeros crosses the imaginary axis ( $s = \pm j\omega$ )

Note that the condition  $a_n(\mathbf{q}) > 0$  has been dropped in this chapter, so it is necessary to add case b). In the parameter space each of the three cases corresponds to a hypersurface. Parts of these hypersurfaces are the stability boundaries. The minimal distance of the origin from these hypersurfaces determines the largest hypersphere.

The first hypersurface is given by  $p(0, \mathbf{q}) = [1 \ 0 \ \dots \ 0] \mathbf{a} = [1 \ 0 \ \dots \ 0] [\mathbf{a}^0 + \mathbf{F}\mathbf{q}] = 0$ . Only the first row of (7.2.1) is important. Because of the affine dependency the hypersurface is a hyperplane with the equation

$$E_0 = a_0^0 + f_{11}q_1 + f_{12}q_2 + \dots + f_{1\ell}q_\ell = 0 \quad (7.2.6)$$

and the squared distance to the origin is with (7.2.5)

$$r_0^2 = \frac{(a_0^0)^2}{f_{11}^2 + f_{12}^2 + \dots + f_{1\ell}^2} \quad (7.2.7)$$

If  $a_0^0 \neq 0$  and  $f_{11} = f_{12} = \dots = f_{1\ell} = 0$ , then case a) is not possible and the hyperplane does not exist. No combination of the parameters  $q_1, q_2, \dots, q_\ell$  will produce a zero at  $s = 0$ . A reasonable choice for the distance is  $r_0 = \infty$ .

The same conclusion leads to the hyperplane for case b). It is  $p(\infty, \mathbf{q}) = 0$  (the leading coefficient must vanish) and

$$E_\infty = a_n^0 + f_{n+1,1}q_1 + f_{n+1,2}q_2 + \dots + f_{n+1,\ell}q_\ell = 0 \quad (7.2.8)$$

with the squared distance

$$r_\infty^2 = \frac{(a_n^0)^2}{f_{n+1,1}^2 + f_{n+1,2}^2 + \dots + f_{n+1,\ell}^2} \quad (7.2.9)$$

For a constant leading coefficient  $a_n$  again  $r_\infty = \infty$ .

For the more complicated case c) the polynomial  $p(s, \mathbf{q}) = [1 \ s \ \dots \ s^n] \mathbf{a} = [1 \ s \ \dots \ s^n] [\mathbf{a}^0 + \mathbf{F}\mathbf{q}]$  must have a root on the imaginary axis. For  $s = j\omega$  with  $\omega \neq 0$  the complex equation is equivalent to the following two real equations

$$E_1 = \operatorname{Re} p(j\omega, \mathbf{q}) = [1 \ 0 \ -\omega^2 \ 0 \ \omega^4 \ \dots] [\mathbf{a}^0 + \mathbf{F}\mathbf{q}] = 0 \quad (7.2.10)$$

and

$$E_2 = \frac{1}{\omega} \operatorname{Im} p(j\omega, \mathbf{q}) = [0 \ 1 \ 0 \ -\omega^2 \ 0 \ \omega^4 \ \dots] [\mathbf{a}^0 + \mathbf{F}\mathbf{q}] = 0 \quad (7.2.11)$$

The complex root boundary is generated by the intersection of the two hyperplanes  $E_1 = 0$  and  $E_2 = 0$  that vary with  $\omega$ . The set of intersection points for fixed  $\omega$  is an  $(\ell - 2)$ -dimensional hyperplane. The distance of the origin to this  $(\ell - 2)$ -dimensional hyperplane is a function of  $\omega$ , call it  $r_C(\omega)$ , and the distance from the complex root boundary is the minimum of this function  $r_C(\omega)$ .

The calculation of  $r_C(\omega)$  would be easy if the two hyperplanes (7.2.10) and (7.2.11) were orthogonal. This is in general not true, however it is possible to replace  $E_2 = 0$  by a third hyperplane  $E_3 = 0$  which is orthogonal to  $E_1 = 0$  such that  $E_1 \cap E_2 = E_1 \cap E_3$ . The important point to note is that the equation

$$E_3 = (1 - \lambda)E_1 + \lambda E_2 = 0 \quad (7.2.12)$$

is satisfied for all points lying on both  $E_1 = 0$  and  $E_2 = 0$ . (7.2.12) represents another  $(\ell - 2)$ -dimensional hyperplane which contains the intersection points of  $E_1 = 0$  and  $E_2 = 0$ , whatever the value of  $\lambda$ . As  $\lambda$  is varied a set of  $(\ell - 2)$ -dimensional hyperplanes is formed, two of which are  $E_1 = 0$  (when  $\lambda = 0$ ) and  $E_2 = 0$  (when  $\lambda = 1$ ).  $E_1 \cap E_3$  will produce the same set as  $E_1 \cap E_2$  if  $\lambda \neq 0$  (see Fig. 7.5).

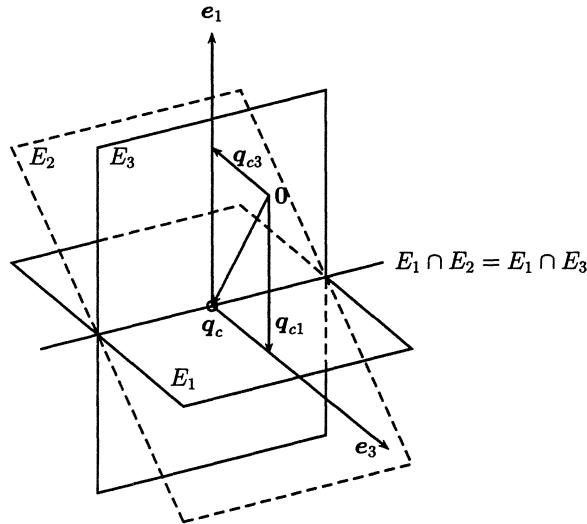


Fig. 7.5: For each frequency  $\omega$  the complex root boundary is generated by the intersection of the two hyperplanes  $E_1 = 0$  and  $E_2 = 0$ , which is equal to the intersection of  $E_1 = 0$  and  $E_3 = 0$

The value of  $\lambda$  can now be chosen in such a way that  $E_1 = 0$  and  $E_3 = 0$  are orthogonal. The normal vector  $e_3$  of  $E_3 = 0$  is

$$e_3 = (1 - \lambda)e_1 + \lambda e_2 \quad (7.2.13)$$

and  $e_1$  and  $e_3$  are orthogonal if

$$e_1^T e_3 = (1 - \lambda)e_1^T e_1 + \lambda e_1^T e_2 = 0 \quad (7.2.14)$$

or

$$\lambda = \frac{e_1^T e_1}{e_1^T e_1 - e_1^T e_2} \quad (7.2.15)$$

Note that  $\lambda$  is a function of  $\omega$ . This orthogonalization procedure is not possible if for special values of  $\omega = \omega_s$  the hyperplanes  $E_1 = 0$  and  $E_2 = 0$  are parallel. This means a) that for these values both hyperplanes have no common points and  $r_C(\omega_s) = \infty$ , or b)  $E_1 \equiv E_2$  and it is obvious how to determine the distance to the origin. Case b) produces a discontinuity in the distance function. It is another appearance of the isolated frequencies that were discussed in Section 4.4.

If the polynomial coefficients vary in the coefficient space between given bounds (interval polynomial) rather than in the parameter space, then the hyperplanes  $E_1 = 0$  and  $E_2 = 0$  are orthogonal and there cannot be any isolated frequencies ( $\mathbf{F} = \mathbf{I}$  and the normal vectors are orthogonal, i.e.  $\mathbf{d}_1^T \mathbf{d}_2 = 0$ ).

The reason for orthogonalizing is that afterwards the distance function  $r_C(\omega)$  can be determined. (7.2.10) and (7.2.11) can be written as

$$\mathbf{D}(\mathbf{a}^0 + \mathbf{F}\mathbf{q}) = \mathbf{D}\mathbf{a}^0 + \mathbf{G}\mathbf{q} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (7.2.16)$$

where

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & -\omega^2 & 0 & \omega^4 & \dots \\ 0 & 1 & 0 & -\omega^2 & 0 & \dots \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1^T \\ \mathbf{d}_2^T \end{bmatrix} \quad (7.2.17)$$

$$\mathbf{G} := \mathbf{D}\mathbf{F} = \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \end{bmatrix} \quad (7.2.18)$$

Thus, in the parameter space the equations of the two hyperplanes are

$$E_1 = \mathbf{d}_1^T \mathbf{a}^0 + \mathbf{e}_1^T \mathbf{q} = 0 \quad (7.2.19)$$

$$E_2 = \mathbf{d}_2^T \mathbf{a}^0 + \mathbf{e}_2^T \mathbf{q} = 0 \quad (7.2.20)$$

where  $\mathbf{e}_1$  and  $\mathbf{e}_2$  are the normal vectors of  $E_1 = 0$  and  $E_2 = 0$ . For each nonisolated frequency  $\omega$  it is possible to construct the hyperplane  $E_3 = 0$  of (7.2.12) with  $\lambda$  of (7.2.15) such that the two hyperplanes  $E_1 = 0$  and  $E_3 = 0$  are orthogonal:  $\mathbf{e}_1^T \mathbf{e}_3 = 0$ . This situation is illustrated by Fig. 7.5. The nominal point is  $\mathbf{q} = \mathbf{0}$ . Seen from there the closest point on the complex root boundary is  $\mathbf{q}_c = \mathbf{q}_{c1} + \mathbf{q}_{c3}$ . The unit vector parallel to  $\mathbf{e}_1$  is  $\mathbf{e}_1 / \sqrt{\mathbf{e}_1^T \mathbf{e}_1}$  and by (7.2.5)  $\mathbf{d}_1^T \mathbf{a}^0 / \sqrt{\mathbf{e}_1^T \mathbf{e}_1}$  is the distance between the origin and  $E_1 = 0$ , thus,

$$\mathbf{q}_{c1} = -\frac{\mathbf{d}_1^T \mathbf{a}^0}{\mathbf{e}_1^T \mathbf{e}_1} \mathbf{e}_1$$

Analog reasoning for  $\mathbf{q}_{c3}$  leads to

$$\mathbf{q}_c = -\frac{\mathbf{d}_1^T \mathbf{a}^0}{\mathbf{e}_1^T \mathbf{e}_1} \mathbf{e}_1 - \frac{\mathbf{d}_3^T \mathbf{a}^0}{\mathbf{e}_3^T \mathbf{e}_3} \mathbf{e}_3 \quad (7.2.21)$$

The distance  $r_C(\omega)$  is

$$r_C(\omega) = \|\mathbf{q}_c\|_2 = \sqrt{\frac{(\mathbf{d}_1^T \mathbf{a}^0)^2}{\mathbf{e}_1^T \mathbf{e}_1} + \frac{(\mathbf{d}_3^T \mathbf{a}^0)^2}{\mathbf{e}_3^T \mathbf{e}_3}} \quad (7.2.22)$$

The squared distance  $r_C^2(\omega)$  is a rational function of  $\omega$  and the necessary condition for the minimum at  $\omega^*$  is that the derivative vanishes. But keep in mind that (7.2.22) is only valid for nonisolated frequencies. At an isolated frequency  $\omega_s$  the normal vectors are parallel. It follows that either the hyperplanes  $E_1 = 0$  and  $E_2 = 0$  are parallel and not identical, then no  $(\ell - 2)$ -dimensional hyperplane is generated and  $r_C(\omega_s) = \infty$  can be set, or the hyperplanes are identical, i.e.  $E_1 = E_2 = 0$  and the formula used for  $E_0 = 0$  can be applied.

After the distances  $r_0, r_\infty, r_C(\omega^*)$  of the origin of the three boundaries are determined the stability radius  $\rho$  is found as

$$\rho = \min\{r_0, r_\infty, r_C(\omega^*)\} \quad (7.2.23)$$

In comparison to the method described in Section 7.1 for the affine case where  $\ell$  functions have to be evaluated, here only one function has to be evaluated.

*Example 7.3.* Recall Example 7.2. The matrices  $\mathbf{a}^0$  and  $\mathbf{F}$  are

$$\mathbf{a}^0 = \begin{bmatrix} 129 \\ 166 \\ 237 \\ 108 \\ 80 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -16 & -21 \\ 24 & 42 \\ -12 & -21 \\ 4 & 0 \\ 0 & 0 \end{bmatrix} \quad (7.2.24)$$

The real root boundary for  $s = 0$  is the straight line

$$E_0 = -16q_1 - 21q_2 + 129 = 0 \quad (7.2.25)$$

which gives  $r_0 = 129/\sqrt{16^2 + 21^2} \approx 4.89$ . The real root boundary for  $s = \infty$  does not exist, so  $r_\infty = \infty$ . The orthogonalization procedure leads for  $\omega \neq \sqrt{2}$  to

$$r_C^2(\omega) = \frac{10(640\omega^8 - 14096\omega^6 + 287085\omega^4 - 439886\omega^2 + 175573)}{441(\omega^2 + 1)^2} \quad (7.2.26)$$

and

$$\mathbf{q}_c = \begin{bmatrix} \frac{67\omega^2 - 53}{\omega^2 + 1} \\ \frac{80\omega^4 + 881\omega^2 + 719}{21(\omega^2 + 1)} \end{bmatrix} \quad (7.2.27)$$

There is a isolated frequency at  $\omega = \sqrt{2}$  and  $E_1(\sqrt{2}) = E_2(\sqrt{2}) = 8q_1 + 21q_2 - 25 = 0$ . Thus,

$$r_C(\sqrt{2}) = \frac{25}{\sqrt{8^2 + 21^2}} \approx 1.11 \quad (7.2.28)$$

A plot of  $r_C(\omega)$  (with logarithmically scaling on the ordinate) shows that this is the minimum (Fig 7.6). Thus, the stability radius is  $\rho_2 = r_C(\sqrt{2})$ .

Note that in Example 7.2 a smaller stability radius of  $\rho = 25/29$  was obtained for the maximum size of the box uncertainty. The hypersphere, i.e. circle for  $\ell = 2$  of Example 7.4 is contained in the box and can be dilated further before it hits the stability boundary.  $\square$

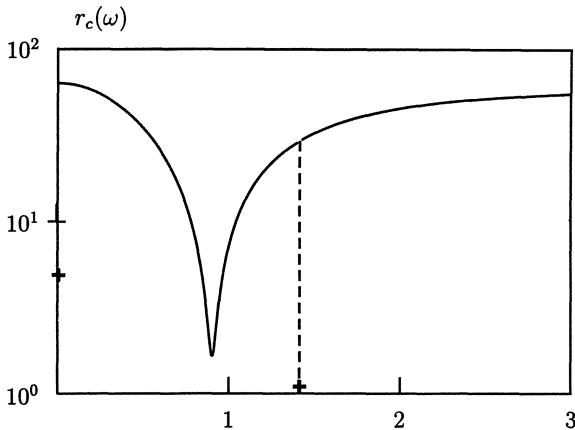


Fig. 7.6: The distance function  $r_C(\omega)$  with two discontinuities at  $\omega = 0$  and  $\omega = \sqrt{2}$

### 7.3 Polynomial Dependency

In this section polynomials are considered whose coefficients depend polynomially on the elements of an uncertain parameter vector. The size of perturbation is characterized by the weighted norm of the parameter vector. The smallest destabilizing perturbation defines the stability radius of the set of uncertain polynomials.

It will be shown that determining this radius is equivalent to solving a finite set of systems of algebraic equations and then selecting the real solution with the smallest norm. The number of systems of equations depends crucially on the dimension  $\ell$  of the parameter vector, whereas the complexity of systems of equations increases mainly with the kind of polynomial dependency  $a(\mathbf{q})$  and the degree  $n$  of the polynomial. This method also yields the smallest destabilizing parameter combination and the corresponding critical frequency.

Vicino et al. [171] transform the problem into an optimization problem and present a numerical algorithm to find the solution. Murdock et al. [132] solve the same problem using a genetic algorithm. It will be shown that the problem can also be solved in an analytical-numerical way, that is systems of algebraic equations [97] have to be solved. The derivation and solution of these systems of equations will be demonstrated in this section.

Given an uncertain polynomial  $p(s, \mathbf{q})$  with  $\mathbf{q} \in \mathbb{R}^\ell$  and real polynomials  $a_i(\mathbf{q})$ . The nominal polynomial  $p(s, \mathbf{0})$  is stable. Find the maximal  $\rho$  such that  $p(s, \mathbf{q})$  is stable for all  $\|\mathbf{q}\|_p < \rho$ .  $\rho$  is called the stability radius and the index  $p$  characterizes the type of norm.

Concerning the choice of the norm, i.e.  $p$ , there are three important possibilities. For  $p = \infty$  the set of admissible  $\mathbf{q}$  describes an  $\ell$ -dimensional hypercube. Dual to this norm

is  $p = 1$ , which corresponds to a diamond.  $p = 2$  yields an  $\ell$ -dimensional hypersphere in  $Q$ -space. From the practical point of view the case  $p = \infty$  is the most important one, because there the bounds for the uncertain parameters are independent. The case  $p = \infty$  will be handled in detail. For the other cases it is not difficult to derive the corresponding results.

It was shown in Section 7.2 that the stable set of  $\mathbf{q}$  is bounded by three hypersurfaces, namely  $a_0(\mathbf{q}) = 0$ ,  $a_n(\mathbf{q}) = 0$  and  $\Delta_{n-1}(\mathbf{q}) = 0$ , see Fig. 7.7.

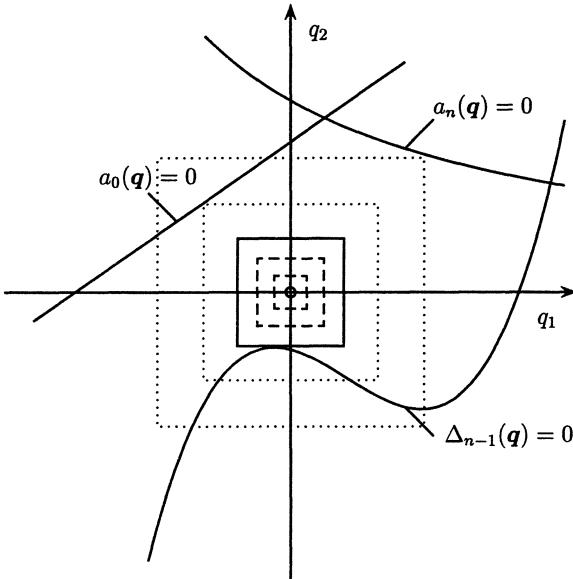


Fig. 7.7: The stable region is bounded by at most three hypersurfaces

The last equation is the last but one Hurwitz determinant, which is the critical one. It results from the elimination of  $\omega$  from the two equations  $\operatorname{Re} p(j\omega) = 0$  and  $\operatorname{Im} p(j\omega) = 0$  as was shown in the proof of Theorem 4.5. Here linear or nonlinear dependency make a big difference. For fixed  $\omega$  and linear dependency  $\operatorname{Re} p = 0$ ,  $\operatorname{Im} p = 0$  represent a linear manifold; this means for example that for  $\ell = 3$ ,  $\Delta_2(\mathbf{q}) = 0$  is generated by the continuous movement of a straight line. In the case of nonlinear dependency a set of curves in  $\mathbb{R}^3$  generate  $\Delta_2 = 0$ . Computing  $\Delta_{n-1}(\mathbf{q})$  must in general be done by a computer algebra program.

All three equations of the hypersurfaces will be treated in the same way. The only difference is that the third one will be the complicated one with respect to the number of terms and degree of the terms. In the sequel the notation  $F(\mathbf{q}) = 0$  is used for each of these equations.

Consider first the case of two parameters, where the basic idea is demonstrated. The polynomial family with  $\|\mathbf{q}\|_\infty \leq \gamma$  ( $\gamma$  sufficiently small) is stable and can be described

by a square of side length  $2\gamma$ . Enlarging this square continuously until there is an intersection point with the curve  $F(q_1, q_2) = 0$  as demonstrated in Fig. 7.7.

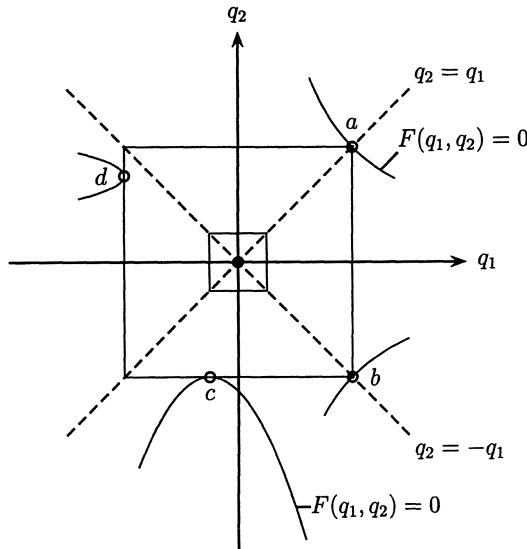


Fig. 7.8: The case  $p = \infty$

This point of first contact with a stability boundary may lie on a vertex or on an edge of the square (see Fig. 7.8). The first situation is characterized by the fact that  $q_1 = q_2$  (marked *a*) or  $q_1 = -q_2$  (marked *b*), which results in the two polynomials

$$\begin{aligned} F(q_1, q_1) &= 0 \\ F(q_1, -q_1) &= 0 \end{aligned} \tag{7.3.1}$$

The second case is an intersection point on an edge. This means that  $F(q_1, q_2) = 0$  has a horizontal (marked *c*) or a vertical tangent (marked *d*). This necessary condition leads to the two systems of equations in two unknowns

$$\begin{aligned} F(q_1, q_2) &= 0, \quad \frac{\partial F(q_1, q_2)}{\partial q_1} = 0 \\ F(q_1, q_2) &= 0, \quad \frac{\partial F(q_1, q_2)}{\partial q_2} = 0 \end{aligned} \tag{7.3.2}$$

It may be possible that in the intersection point the curve  $F(q_1, q_2) = 0$  can not be differentiated, i.e. the curve has for example a cusp or an isolated point. For these points both partial derivatives vanish and these solutions are already obtained by (7.3.2).

Finding the real roots of the two polynomials and the real solution vectors of the two systems of equations gives us a set of points  $(q_1, q_2)$ , which are candidates for the

first contact. For the first case (vertex contact)  $\|(\pm q, \pm q)\|_\infty = \|\mathbf{q}\|_\infty = |q|$  and for the second case (edge contact)  $\|(q_1, q_2)\|_\infty = \max\{|q_1|, |q_2|\}$ . The solution vector with the smallest norm  $\mathbf{q}^* = [q_1^*; q_2^*]^T$  yields the stability radius. This critical parameter combination  $\mathbf{q}^*$  also determines the critical frequency. The polynomial  $p(s, q_1^*, q_2^*) = 0$  has a root at  $s = 0$  or at  $s = \infty$  or a root pair at  $s = \pm j\omega$ . All other roots are not in the open right half plane.

Consider the case of three parameters. A surface  $F(q_1, q_2, q_3) = 0$  replaces the curve  $F(q_1, q_2) = 0$ . This surface bounds the stable polynomials and now a cube instead of a square is enlarged. The intersection points lie either on a vertex, on an edge or on a face of the cube. These subsets of the cube are called *subpolytopes*. Corresponding to the eight vertices of the cube the four polynomials

$$\begin{aligned} F(q, +q, +q) &= 0 \\ F(q, +q, -q) &= 0 \\ F(q, -q, +q) &= 0 \\ F(q, -q, -q) &= 0 \end{aligned}$$

describe this situation. In case of the twelve edges two intersection points must coincide, i.e. the partial derivatives must vanish and the six systems of equations

$$F(q_1, +q, +q) = 0, \quad \left. \frac{\partial F}{\partial q_1} \right|_{q_2=+q, q_3=+q} = 0$$

$$F(q_1, +q, -q) = 0, \quad \left. \frac{\partial F}{\partial q_1} \right|_{q_2=+q, q_3=-q} = 0$$

$$F(+q, q_2, +q) = 0, \quad \left. \frac{\partial F}{\partial q_2} \right|_{q_1=+q, q_3=+q} = 0$$

$$F(-q, q_2, +q) = 0, \quad \left. \frac{\partial F}{\partial q_2} \right|_{q_1=-q, q_3=+q} = 0$$

$$F(+q, +q, q_3) = 0, \quad \left. \frac{\partial F}{\partial q_3} \right|_{q_1=+q, q_2=+q} = 0$$

$$F(+q, -q, q_3) = 0, \quad \left. \frac{\partial F}{\partial q_3} \right|_{q_1=+q, q_2=-q} = 0$$

in two unknowns have to be solved.

If the intersection is on one of the six surfaces then the normal vector of the surface  $F(q_1, q_2, q_3) = 0$  is parallel to one of the coordinate axes which means that two of the three partial derivatives vanish simultaneously and the three systems

$$F(q_1, q_2, q_3) = 0, \quad \frac{\partial F}{\partial q_1} = 0, \quad \frac{\partial F}{\partial q_2} = 0$$

$$F(q_1, q_2, q_3) = 0, \quad \frac{\partial F}{\partial q_1} = 0, \quad \frac{\partial F}{\partial q_3} = 0$$

$$F(q_1, q_2, q_3) = 0, \quad \frac{\partial F}{\partial q_2} = 0, \quad \frac{\partial F}{\partial q_3} = 0$$

in three unknowns characterize this situation.

For  $\ell > 3$  parameters the number of polynomials and systems of equations obviously grows with the number of subpolytopes of an  $\ell$ -dimensional cube. The derivation of the polynomials and systems of equations for arbitrary  $\ell$  is straightforward. But now again occurs what is called the combinatorial explosion. The number of systems grows exponentially with the number of parameters.

Using other norms ( $p = 2$  or  $p = 1$ ) gives similar equations. In any case always equations which are linear combinations of  $F$  and their partial derivatives have to be solved.

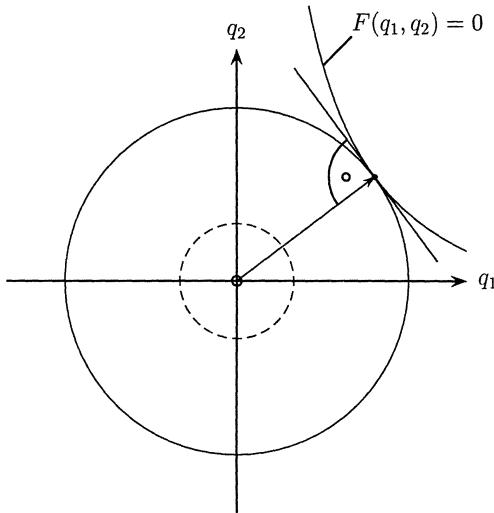


Fig. 7.9: The case  $p = 2$

For  $p = 2$  the number of systems is smaller, but the equations are more complicated. In that case at the intersection point the vector  $\mathbf{q}$  must be parallel to the gradient of  $F$  like indicated in Fig. 7.9.

Only one system of equations

$$F(\mathbf{q}) = 0, \quad \frac{\partial F}{\partial \mathbf{q}} = \lambda \mathbf{q}$$

is necessary for describing this situation.

A restriction for practical applications is the problem to find all solutions of the system of equations. The Theorem of Bezout (see Appendix B) says that for example a system of three equations with the degrees  $m_i$ ,  $i = 1, 2, 3$  has up to  $m_1 \cdot m_2 \cdot m_3$  solution vectors. So the resulting polynomial in one variable after eliminating the other two variables may have the degree  $m_1 \cdot m_2 \cdot m_3$ . Therefore, this method is recommended only up to three uncertain parameters. The restriction to few parameters seems to be severe, but a quotation of Lazard [114] in 1991 on software for the solution of algebraic equations gives hope for the future. "Five years ago, problems with four or five unknowns were outside of the capabilities of most available softwares. Recent progresses made or will make accessible problems with six or seven unknowns."

*Example 7.4.* Consider the track guided bus of Chapter 1. The transfer function of the uncontrolled bus depends on the virtual mass  $\tilde{m}$  and on the velocity  $v$ . For the data given in Exercise 1.4 it is

$$g(s, \tilde{m}, v) = \frac{6.079 \cdot 10^5 \tilde{m}v^2 s^2 + 3.886 \cdot 10^{11} vs + 4.803 \cdot 10^{10} v^2}{s^3(\tilde{m}^2 v^2 s^2 + 9.818 \cdot 10^5 \tilde{m}vs + 1.663 \cdot 10^4 \tilde{m}v^2 + 2.690 \cdot 10^{11})}$$

In [131] a controller was designed which led to the closed-loop polynomial

$$p(s, \tilde{m}, v) = \sum_{i=0}^8 a_i s^i$$

with

$$\begin{aligned} a_0 &= 4.503 \cdot 10^{14} v^2 \\ a_1 &= 5.253 \cdot 10^{14} v^2 + 3.625 \cdot 10^{15} v \\ a_2 &= 5.699 \cdot 10^9 \tilde{m}v^2 + 1.128 \cdot 10^{14} v^2 + 4.229 \cdot 10^{15} v \\ a_3 &= 6.908 \cdot 10^9 \tilde{m}v^2 + 9.062 \cdot 10^{14} v + 4.203 \cdot 10^{15} \\ a_5 &= 1.563 \cdot 10^4 \tilde{m}^2 v^2 + 8.315 \cdot 10^5 \tilde{m}v^2 + 1.344 \cdot 10^9 \tilde{m}v + 1.345 \cdot 10^{13} \\ a_6 &= 1.25 \cdot 10^3 \tilde{m}^2 v^2 + 1.663 \cdot 10^4 \tilde{m}v^2 + 5.376 \cdot 10^7 \tilde{m}v + 2.690 \cdot 10^{11} \\ a_7 &= 50 \tilde{m}^2 v^2 + 1.075 \cdot 10^6 \tilde{m}v \\ a_8 &= \tilde{m}^2 v^2 \end{aligned}$$

Let the nominal point be  $v = 20$  [ms<sup>-1</sup>] and  $\tilde{m} = 20$  [10<sup>3</sup> kg]. For determining the stability radius the parameter space method can be applied, i.e. plotting in the  $(v, \tilde{m})$ -plane the stability boundaries, see Fig. 7.10. But also the analytical method of this

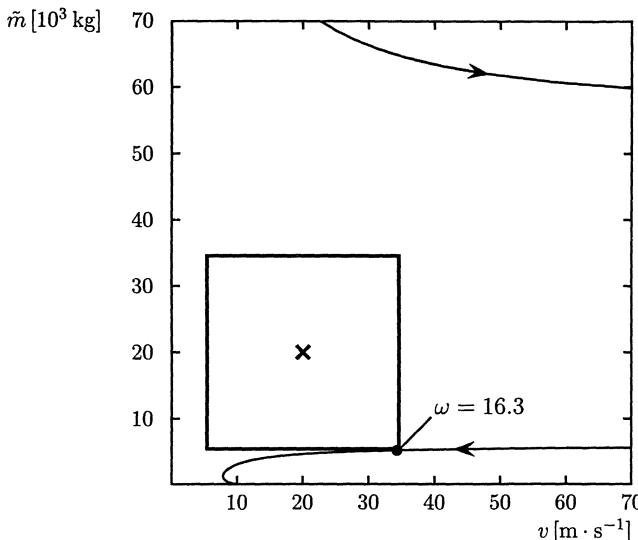


Fig. 7.10: The maximal square around the nominal point (20,20)

section is applicable. The distances to the real root boundaries  $\tilde{m} = 0$  and  $v = 0$  are trivial, but for the distance to the complex root boundary the Hurwitz determinant, which is of order seven, has to be calculated. This must be done by a computer algebra program. Solving (7.3.1) and (7.3.2) with  $q_1 = v - 20$  and  $q_2 = \tilde{m} - 20$  yields an intersection point of the growing box with the complex root boundary at  $q_1 = 14.8$ ,  $q_2 = -q_1$  for  $\omega = 16.3$ , see Fig. 7.10.  $\square$

## 7.4 Exercises

- 7.1. Find the stability radius (with respect to a sphere) of the uncertain polynomial

$$p(s, q_1, q_2, q_3) = (1 - q_1) + (-3 + q_2)s - q_3s^2 - s^3$$

starting with the nominal point  $q_1 = 2.5$ ,  $q_2 = 1.5$  and  $q_3 = 3.5$ . Which methods of this chapter can be used? Compare the possible methods.

- 7.2. Find the stability radius (with respect to a square) of the uncertain polynomial

$$p(s, q_1, q_2) = (1 + q_1) + (2 + q_1 + q_2)s + (2 + q_1)s^2 + s^3$$

starting with the nominal point  $q_1 = 2$  and  $q_2 = 0.5$ . Compare the method of Tsyplkin and Polyak with the general method for polynomial dependency.

- 7.3. Find the stability radius (with respect both to a circle and a square) of the uncertain polynomial

$$p(s, q_1, q_2) = (1 + q_1 q_2) + (2 + q_1 + q_2 - 3q_1 q_2)s + (4 + 2q_1 + 5q_2)s^2 + s^3$$

starting with the nominal point  $q_1 = 0.5$  and  $q_2 = 0.5$ .

- 7.4. Consider Example 2.7 of the crane with  $g = 10$ ,  $m_C = 1000$ ,  $\ell = 10$ ,  $k_1 = 500$ ,  $k_2 = 5000$ ,  $k_3 = 0$ . Find the stability radius (with respect to a square) with the uncertain parameters  $m_L$  and  $k_4$  if

- a)  $m_L^0 = 50$  and  $k_4^0 = 0$ ,
- b)  $m_L^0 = 3000$  and  $k_4^0 = 0$ .

- 7.5. Like Exercise 7.4 with  $\ell$  as third uncertain parameter and  $\ell^0 = 10$ . Because for  $\ell = 0$  the degree of the polynomial drops, the stability radius is always  $< 10$ . Therefore, introduce first a new variable  $\ell^* = 100\ell$ . Find the stability radius (with respect to a cube and a sphere) with the three uncertain parameters  $m_L$ ,  $k_4$  and  $\ell^*$ .

## **Part III**

# **Robustness Analysis of Feedback Systems**

# Single-Loop Feedback Structures

With Chapter 8 we are entering into Part III of this book. It deals with specific feedback structures and with additional robustness requirements on the closed-loop system, like sector uncertainty of a nonlinearity, Gamma-stability and discrete-time implementation of a robust controller.

In Chapter 8 robustness analysis results for single-loop feedback structures are introduced. These results are based on special structural subsystem properties, especially on properties of the open-loop system. We consider a simple unity feedback loop, see Fig. 8.1, where  $g(s, \mathbf{q})$  represents the uncertain open-loop transfer function

$$g(s, \mathbf{q}) = \frac{n(s, \mathbf{q})}{d(s, \mathbf{q})} \quad (8.0.1)$$

with the uncertain numerator polynomial

$$n(s, \mathbf{q}) = n_0(\mathbf{q}) + n_1(\mathbf{q})s + \dots + n_m(\mathbf{q})s^m \quad (8.0.2)$$

and the uncertain denominator polynomial

$$d(s, \mathbf{q}) = d_0(\mathbf{q}) + d_1(\mathbf{q})s + \dots + d_n(\mathbf{q})s^n, \quad d_n(\mathbf{q}) > 0 \quad (8.0.3)$$

As usual  $g(s, \mathbf{q})$  is assumed to be proper ( $m \leq n$ ).

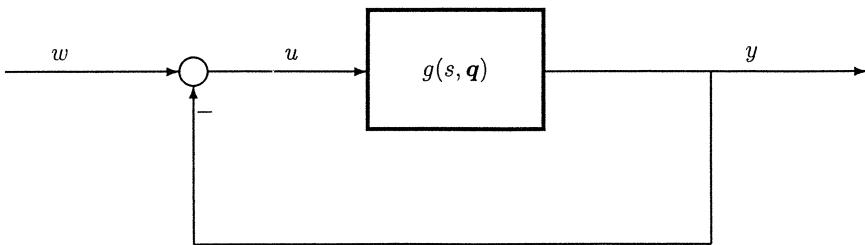


Fig. 8.1: The unity feedback loop of an uncertain SISO system

If the numerator  $n(s, \mathbf{q})$  and the denominator  $d(s, \mathbf{q})$  of the open-loop (plant) transfer function  $g(s, \mathbf{q})$  are interval polynomials

$$\begin{aligned} n(s, \mathbf{q}) &= n(s, \mathbf{n}) = n_0 + n_1 s + \dots + n_m s^m, & n_i &\in [n_i^-; n_i^+], & i &= 0, 1, \dots, m \\ d(s, \mathbf{q}) &= d(s, \mathbf{d}) = d_0 + d_1 s + \dots + d_n s^n, & d_i &\in [d_i^-; d_i^+], & i &= 0, 1, \dots, n \end{aligned} \quad (8.0.4)$$

then obviously the closed-loop characteristic polynomial

$$p(s, \mathbf{n}, \mathbf{d}) = n(s, \mathbf{n}) + d(s, \mathbf{d}) = a_0(\mathbf{n}, \mathbf{d}) + a_1(\mathbf{n}, \mathbf{d})s + \dots + a_n(\mathbf{n}, \mathbf{d})s^n$$

is also of the interval type since the coefficients  $a_i = n_i + d_i$  are mutually independent inside the interval boundaries

$$\begin{aligned} a_i^- &= n_i^- + d_i^- \\ a_i^+ &= n_i^+ + d_i^+, \quad i = 0, 1, \dots, n \end{aligned} \tag{8.1}$$

(Here and in the following undefined coefficients are zero, e.g. for  $m < n$ ,  $n_m = 0$ ). Robust stability of the closed-loop system can be proved by checking the four Kharitonov polynomials of (8.0.5). However, even for such an interval plant, only very special types of compensators in the feedback loop preserve the interval property of the closed-loop characteristic polynomial. Consider the feedback loop of Fig. 8.2 where the plant coefficients  $n_i, d_i, i = 0, 1$  are assumed to be uncertain within corresponding intervals and with fixed values of the compensator coefficients  $k_0 = k_0^*$  and  $k_1 = k_1^*$ . The characteristic polynomial of the closed-loop system,

$$\begin{aligned} p(s, n_0, n_1, d_0, d_1) &= k_0^*(d_0 + d_1 s + s^2) + (k_1^* + s)(n_0 + n_1 s) \\ &= k_0^* d_0 + k_1^* n_0 + (k_0^* d_1 + k_1^* n_1 + n_0)s + (k_0^* + n_1)s^2 \end{aligned} \tag{8.0.7}$$

is affine in the uncertain coefficients, only in the special case  $k_1^* = 0$  it is of the interval type. As a consequence, already for a general first order compensator there exists only a Kharitonov-like result and for a compensator with arbitrary order we can only get a special edge theorem. Both results are given in Section 8.1.

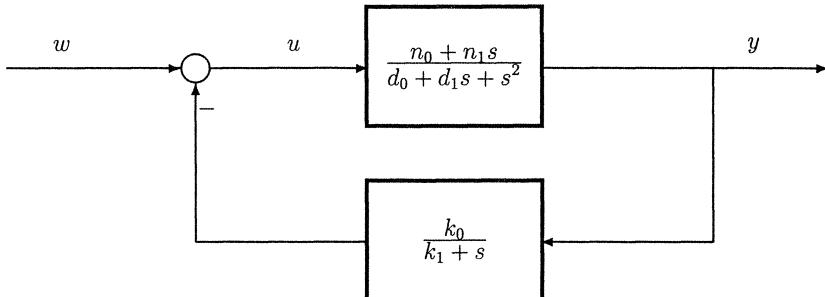


Fig. 8.2: The characteristic polynomial of the closed-loop system is not an interval polynomial

With an additional property of the interval plant transfer function, namely with the property of being strictly positive real and with a strictly positive real compensator in the feedback loop, again Kharitonov-like results exist for the robustness check, which we will present in Section 8.2. In Section 8.3 the assumption of an interval transfer function or of a positive real transfer function is given up. Instead, we investigate if an uncertain transfer function  $g(s, \mathbf{q})$  of a more general type can be brought into a form with some sort of a tree structure. It will be seen that the generalization of a tree

structure for polynomials (Section 6.5) to rational transfer functions can be useful for constructing the open-loop value set, in order to check robust stability by the classical Nyquist criterion. In Section 8.4 we go back to interval plants, but now in combination with a sector nonlinearity in the feedback loop and we will present a Kharitonov-like robustness result for that feedback structure.

## 8.1 Interval Plants with a Compensator

With the vector of uncertain parameters

$$\mathbf{q} := [q_1 \dots q_\ell]^T = [n_0 \ n_1 \dots n_m \ d_0 \ d_1 \dots d_n]^T = [\mathbf{n}^T \ \mathbf{d}^T]^T, \quad \ell = m + n + 2 \quad (8.1.1)$$

a family of transfer functions

$$G_I = G(s, Q) := \left\{ g(s, \mathbf{q}) = \frac{n(s, \mathbf{n})}{d(s, \mathbf{d})} \mid \mathbf{q} \in Q \right\} \quad (8.1.2)$$

for the  $Q$ -box

$$Q = \{ [\mathbf{n}^T \ \mathbf{d}^T]^T \mid n_i \in [n_i^- ; n_i^+], \ d_j \in [d_j^- ; d_j^+], \ i = 0, 1, \dots, m, \ j = 0, 1, \dots, n \} \quad (8.1.3)$$

is introduced. Here  $n(s, \mathbf{n})$  and  $d(s, \mathbf{d})$  are interval polynomials which generate the polynomial families  $N(s, Q_n)$  and  $D(s, Q_d)$ ,

$$\begin{aligned} N(s, Q_n) &:= \{n(s, \mathbf{n}) \mid \mathbf{n} \in Q_n\} \\ D(s, Q_d) &:= \{d(s, \mathbf{d}) \mid \mathbf{d} \in Q_d\} \end{aligned} \quad (8.1.4)$$

with  $Q_n$  and  $Q_d$  the uncertainty boxes of the numerator and the denominator coefficients,

$$Q_n := \{\mathbf{n} \mid n_i \in [n_i^- ; n_i^+], \ i = 0, 1, \dots, m\} \quad (8.1.5)$$

$$Q_d := \{\mathbf{d} \mid d_i \in [d_i^- ; d_i^+], \ i = 0, 1, \dots, n\} \quad (8.1.6)$$

The transfer function

$$g(s, \mathbf{n}, \mathbf{d}) = \frac{n(s, \mathbf{n})}{d(s, \mathbf{d})} \quad (8.1.7)$$

generating the family  $G_I$  of transfer functions in (8.1.2) is called an *interval transfer function*. For notational simplicity the four Kharitonov polynomials of  $N$  and  $D$  of (8.1.4) are characterized by indices,

$$\begin{aligned} n^{(1)}(s) &:= n^{+-}(s), \ n^{(2)}(s) := n^{++}(s), \ n^{(3)}(s) := n^{-+}(s), \ n^{(4)}(s) := n^{--}(s) \\ d^{(1)}(s) &:= d^{+-}(s), \ d^{(2)}(s) := d^{++}(s), \ d^{(3)}(s) := d^{-+}(s), \ d^{(4)}(s) := d^{--}(s) \end{aligned} \quad (8.1.8)$$

With these eight Kharitonov polynomials we denote the *Kharitonov plant family*  $G_I^K$  of the family of transfer functions  $G_I$  as the set

$$G_I^K := \left\{ \frac{n^{(i)}(s)}{d^{(j)}(s)} ; \quad i, j \in \{1, 2, 3, 4\} \right\} \quad (8.1.9)$$

of 16 transfer functions. As we already know, for checking stability of an interval transfer function it is necessary and sufficient to check if the four Kharitonov polynomials  $d^{(i)}(s)$ ,  $i = 1, 2, 3, 4$  of the denominator are stable. However, for the standard feedback loop of Fig. 8.3, where  $\hat{g}(s)$  is a fixed proper (controller) transfer function

$$\hat{g}(s) = \frac{\hat{n}(s)}{\hat{d}(s)} = \frac{\hat{n}_0 + \hat{n}_1 s + \dots + \hat{n}_{\hat{n}} s^{\hat{n}}}{\hat{d}_0 + \hat{d}_1 s + \dots + \hat{d}_{\hat{n}} s^{\hat{n}}} \quad (8.1.10)$$

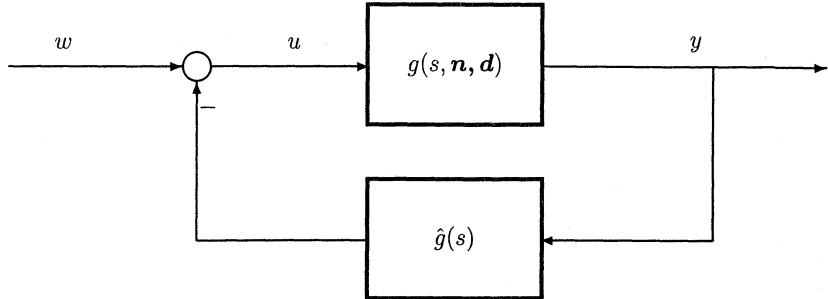


Fig. 8.3: Interval plant transfer function with fixed compensator transfer function in the feedback loop  
the closed-loop characteristic polynomial

$$p(s, \mathbf{n}, \mathbf{d}) = \hat{n}(s)n(s, \mathbf{n}) + \hat{d}(s)d(s, \mathbf{d}) \quad (8.1.11)$$

is no longer of the interval type. Substituting the respective polynomials in (8.1.11) we get

$$p(s, \mathbf{n}, \mathbf{d}) = \sum_{i=0}^{\hat{n}+n} \sum_{j=0}^i (\hat{n}_j n_{i-j} + \hat{d}_j d_{i-j}) s^i =: \sum_{i=0}^{\hat{n}+n} a_i s^i \quad (8.1.12)$$

The coefficients of the closed-loop characteristic polynomial

$$a_i = \sum_{j=0}^i (\hat{n}_j n_{i-j} + \hat{d}_j d_{i-j}), \quad i = 0, 1, \dots, \hat{n} + n \quad (8.1.13)$$

preserve a special affine dependency on the uncertain parameters  $n_i$  and  $d_j$ . For special classes of the compensator this leads to Kharitonov results or to Kharitonov-like results as will be shown later.

Since the parameter dependency of the closed-loop characteristic polynomial for the feedback loop of Fig. 8.3 is affine, we know from the edge theorem that it suffices to check  $\ell 2^{\ell-1}$  edges of the  $\ell$ -dimensional box  $Q$  for answering the question of robust stability. We ask now if the special nature of the affine dependency of the coefficients in (8.1.13) on  $n_i$  and  $d_j$  leads us to a simpler robustness test. For the closed-loop characteristic polynomial (8.1.11), the value set for a fixed  $s = j\omega^*$  is the set

$$\mathcal{P}(j\omega^*, \mathbf{n}, \mathbf{d}) = \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d}), \quad \mathbf{n} \in Q_n, \quad \mathbf{d} \in Q_d \quad (8.1.14)$$

Since the characteristic polynomial has a tree structure (see Section 6.5), the value sets  $\hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n})$  and  $\hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d})$  can be generated independently. To obtain the

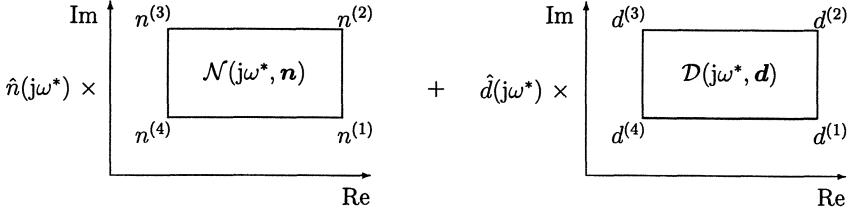


Fig. 8.4: Value set operations for interval plants

value set  $\mathcal{P}(j\omega^*, \mathbf{n}, \mathbf{d})$  of the characteristic polynomial both sets just have to be added. The sets  $\hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n})$  and  $\hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d})$  can easily be constructed. We know that  $\mathcal{N}(j\omega^*, \mathbf{n})$  is a rectangle whose vertices are formed by the complex values of the four Kharitonov polynomials of  $N(s, Q_n)$  for  $\omega = \omega^*$ , see Fig. 8.4. This rectangle has to be multiplied by the complex number  $\hat{n}(j\omega^*)$ , which leads to a rotation and a dilation of the rectangle  $\mathcal{N}(j\omega^*, \mathbf{n})$  corresponding to phase and magnitude of  $\hat{n}(j\omega^*)$ . The result is again a rectangle. The same procedure is performed to get the set  $\hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d})$ . Now both sets have to be added. This is done according to the rules for value set addition derived in Section 6.2. The operations

$$\begin{aligned} \hat{n}(j\omega^*)n^{(1)}(j\omega^*) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d}), & \quad \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)d^{(1)}(j\omega^*) \\ \hat{n}(j\omega^*)n^{(2)}(j\omega^*) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d}), & \quad \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)d^{(2)}(j\omega^*) \\ \hat{n}(j\omega^*)n^{(3)}(j\omega^*) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d}), & \quad \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)d^{(3)}(j\omega^*) \\ \hat{n}(j\omega^*)n^{(4)}(j\omega^*) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d}), & \quad \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)d^{(4)}(j\omega^*) \end{aligned} \quad (8.1.15)$$

have to be performed. The four rectangles on the left hand side of (8.1.15) have the same shape and orientation in the complex plane and the same is true also for the four rectangles on the right hand side of (8.1.15). Furthermore, it can easily be seen that each vertex of an arbitrary rectangle on the left hand side of (8.1.15) is a vertex of a rectangle on the right hand side and vice versa. It follows that the value set  $\mathcal{P}(j\omega^*, \mathbf{n}, \mathbf{d})$  which is the union of the eight rectangles of (8.1.15) must have the shape of Fig. 8.5. If the dashed lines are considered as the rectangles  $\hat{n}(j\omega^*)n^{(i)}(j\omega^*) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d})$ ,  $i = 1, \dots, 4$ , then the rectangles drawn with solid lines are  $\hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)d^{(i)}(j\omega^*)$ ,  $i = 1, \dots, 4$ .

Since the above arguments hold for an arbitrary  $\omega^* \geq 0$  we get from the zero exclusion principle the following theorem (first given in [44] where it is denoted as the Box Theorem.):

*Theorem 8.1. (Box Theorem of Chapellat and Bhattacharyya)*

The uncertain polynomial (8.1.11) is robustly stable for  $Q$  if and only if

- i) there exists a  $\mathbf{q}^* = [\mathbf{n}^{*T} \ \mathbf{d}^{*T}]^T \in Q$  such that  $p(s, \mathbf{n}^*, \mathbf{d}^*)$  is stable and
- ii) the edges of the value sets of (8.1.15) do not touch the origin for all  $\omega^* \geq 0$ .

□

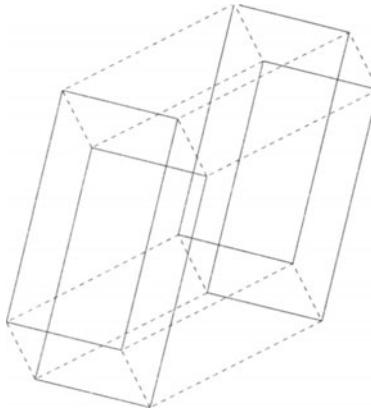


Fig. 8.5: The value set  $\mathcal{P}(j\omega^*, \mathbf{n}, \mathbf{d}) = \hat{n}(j\omega^*)\mathcal{N}(j\omega^*, \mathbf{n}) + \hat{d}(j\omega^*)\mathcal{D}(j\omega^*, \mathbf{d})$ ,  $\mathbf{n} \in Q_n, \mathbf{d} \in Q_d$

The first condition is easy to check and to show robust stability, it suffices to check the  $8 \cdot 4$  edges of the above rectangles for zero exclusion for all  $\omega \geq 0$ . Independent of the number of uncertain parameters of the interval plant, robust stability of the feedback loop of Fig. 8.3 can be proved by checking only the zero exclusion of no more than 32 edges in the complex plane for  $\omega \geq 0$ . Theorem 8.1 is equivalent to the following theorem:

*Theorem 8.2.*

The uncertain polynomial (8.1.11) is robustly stable for  $Q$  if and only if the edge polynomials

$$\begin{aligned} & \{\hat{n}(s)n^{(i)}(s) + \hat{d}(s)[d^{(j)}(s) + \lambda(d^{(k)}(s) - d^{(j)}(s))] \quad | \lambda \in [0; 1], \quad i, j \in \{1, \dots, 4\} \\ & \{\hat{n}(s)[n^{(i)}(s) + \lambda(n^{(k)}(s) - n^{(j)}(s))] + \hat{d}(s)d^{(i)}(s) \quad | \lambda \in [0, 1], \quad i, j \in \{1, \dots, 4\} \end{aligned} \quad (8.1.16)$$

are stable, where  $k = j + 1$  if  $j < 4$  and  $k = 1$  if  $j = 4$ .

□

The edge polynomials (8.1.16) are called *Kharitonov segments*. With Theorem 8.2 the robustness test can be performed by the edge test of Bialas (Theorem 4.6) for the 32 Kharitonov segments of (8.1.16).

For some very simple compensators it is possible to obtain a vertex result instead of the above edge result. If the compensator of Fig. 8.3 is restricted to be proportional feedback,

$$\hat{g}(s) = k_P, \quad k_P > 0 \quad (8.1.17)$$

then the closed-loop characteristic polynomial is

$$\begin{aligned} p(s, \mathbf{n}, \mathbf{d}) &= k_P(n_0 + n_1 s + \dots + n_m s^m) + (d_0 + d_1 s + \dots + d_n s^n) \\ &= a_0 + a_1 s + \dots + a_n s^n \end{aligned} \quad (8.1.18)$$

For  $n_n \in [n_n^- ; n_n^+]$ ,  $d_n \in [d_n^- ; d_n^+]$  assume

$$a_n = k_p n_n + d_n > 0 \quad (8.1.19)$$

The coefficients  $a_i$  of the characteristic polynomial (8.1.18),

$$a_i = k_P n_i + d_i, \quad i = 0, 1, \dots, n \quad (8.1.20)$$

obviously are independent of each other and each coefficient value is contained in the interval

$$a_i \in [k_P n_i^- + d_i^- ; k_P n_i^+ + d_i^+], \quad i = 0, 1, \dots, n \quad (8.1.21)$$

Thus  $p$  is again an interval polynomial, whose four Kharitonov polynomials have to be checked for determining robust stability [74].

The result can be generalized to the purely integrating controller of order  $k$ ,

$$\hat{g}(s) = \frac{k_I}{s^k}, \quad k \geq 0 \quad (8.1.22)$$

The closed-loop characteristic polynomial now is

$$\begin{aligned} p(s, \mathbf{n}, \mathbf{d}) &= k_I(n_0 + n_1 s + \dots + n_m s^m) + s^k(d_0 + d_1 s + \dots + d_n s^n) \\ &= a_0 + a_1 s + \dots + a_{n+k-1} s^{k+n-1} + a_{n+k} s^{k+n} \end{aligned} \quad (8.1.23)$$

and again the coefficients

$$a_i = k_I n_i + d_{i-k}, \quad i = 0, 1, \dots, k+n \quad (8.1.24)$$

are independent of each other and have the interval property

$$a_i \in [k_I n_i^- + d_{i-k}^- ; k_I n_i^+ + d_{i-k}^+], \quad i = 0, 1, \dots, k+n \quad (8.1.25)$$

The results for a proportional controller (8.1.17) or an integral controller (8.1.22) follow immediately from Kharitonov's theorem.

Recently, also a new *Kharitonov-like* result has been found, i.e. a result that yields a testing set with a fixed number of elements, independent of the number of uncertain parameters. Consider the general first order compensator

$$\hat{g}(s) = \frac{\hat{n}(s)}{\hat{d}(s)} = \frac{\hat{n}_0 + \hat{n}_1 s}{\hat{d}_0 + s} \quad (8.1.26)$$

where a PI-controller is a special case ( $\hat{d}_0 = 0$ ). Hollot et al. [84] succeeded in reducing the testing set to sixteen vertices.

*Theorem 8.3.* (Hollot, Kraus, Tempo, Barmish)

The closed-loop system of Fig. 8.3, where  $g(s, \mathbf{n}, \mathbf{d})$  is an interval (plant) transfer function is robustly stable with the fixed compensator (8.1.26) if and only if the 16 characteristic polynomials of the set

$$P^K := \{\hat{n}(s)n^{(i)}(s) + \hat{d}(s)d^{(j)}(s) \mid i, j \in \{1, 2, 3, 4\}\} \quad (8.1.27)$$

are stable.

□

The proof is quite complicated, the reader is referred to [32]. The set  $P^K$  of 16 characteristic polynomials is obtained when the 16 transfer functions of the Kharitonov plants  $G_I^K$  (8.1.9), of the interval plant  $G_I$  are chosen.

*Example 8.1.* For the crane in Section 1.1 the transfer function (1.1.10) from the force  $u$  to the crab position  $y_c$  was derived,

$$g_c(s, g, \ell, m_C, m_L) = \frac{g + \ell s^2}{[(m_L + m_C)g + m_C \ell s^2]s^2} \quad (8.1.28)$$

This transfer function is not of the interval type since  $g, \ell$  and  $m_C$  appear in more than one coefficient function. But with fixed values  $g = 10$  [m/s<sup>2</sup>],  $\ell = 10$  [m] and  $m_C = 1000$  [kg] we get the transfer function

$$g_c(s, m_L) = \frac{1 + s^2}{(m_L + 1000)s^2 + 1000s^4} \quad (8.1.29)$$

uncertain in the load mass  $m_L \in [m_L^-; m_L^+]$ . Now  $g_c(s, m_L)$  obviously is an interval transfer function. Thus for a controller of the type (2.3.4) with  $k_3 = 0$ ,

$$\hat{g}(s) = k_1 + \frac{k_2 s}{1 + T s} = \frac{k_1/T + (k_1 + k_2/T)s}{1/T + s} =: \frac{\hat{n}_0 + \hat{n}_1 s}{\hat{d}_0 + s} \quad (8.1.30)$$

we can analyze robust stability of the closed-loop system for any chosen controller coefficients  $k_1, k_2$  and  $T$  by applying Theorem 8.3. Since we have a single uncertain coefficient, the set (8.1.27) has only two elements:

$$\begin{aligned} P^K &= \{(\hat{n}_0 + \hat{n}_1 s) [(m_L^- + 1000)s^2 + 1000s^4] + (\hat{d}_0 + s)(1 + s^2), \\ &\quad (\hat{n}_0 + \hat{n}_1 s) [(m_L^+ + 1000)s^2 + 1000s^4] + (\hat{d}_0 + s)(1 + s^2)\} \end{aligned} \quad (8.1.31)$$

Therefore the closed loop for the interval plant (8.1.29) with the controller of first order (8.1.30) is robustly stable if and only if it is stable for  $m_L^-$  and  $m_L^+$ . □

We will summarize the robustness results that we have given so far for an interval plant with a fixed linear compensator in the feedback loop, see Fig. 8.3:

Independent of the compensator and plant order, robust stability can be checked with the 32 Kharitonov segments (8.1.16). This can be done with the Bialas test or by construction of the value sets (8.1.14). (Theorem 8.1 and Theorem 8.2)

For a first order controller we have to check only the stability of at most 16 vertex polynomials  $P^K$  of (8.1.27). This result is independent of the number of uncertain parameters of the interval plant, i.e. we have a Kharitonov-like result. (Theorem 8.3)

If the compensator is further specialized to be of the proportional or purely integrating type we could easily recognize the interval property of the closed-loop characteristic polynomial and it is sufficient to check the respective Kharitonov polynomials.

## 8.2 Positive Interval Plants with a Positive Compensator

There is a class of systems characterized by the special property of being strictly positive real. This property has important consequences for (robust) stability in a feedback loop. We will concentrate on the special and simplest case of linear time-invariant SISO systems which will be considered in the form of a proper rational transfer function

$$g(s) = \frac{n(s)}{d(s)} \quad (8.2.1)$$

with the numerator polynomial

$$n(s) = n_0 + n_1 s + \dots + n_m s^m \quad (8.2.2)$$

and the denominator polynomial

$$d(s) = d_0 + d_1 s + \dots + d_n s^n \quad (8.2.3)$$

*Definition 8.1.* A proper transfer function  $g(s)$  is called *strictly positive real* if

- i)  $d(s)$  is Hurwitz (all  $n$  zeros of  $d(s)$  are in the open left half complex plane) and
- ii)  $\operatorname{Re} g(j\omega) > 0$ , for all  $\omega \in \mathbb{R}$ .

□

In other words, a transfer function is strictly positive real if it is stable and its Nyquist plot is completely contained in the right half complex plane. Hence, for the phase shift  $\varphi$  of  $g$ ,  $|\varphi| \leq \pi/2$  for all  $\omega \in \mathbb{R}$  is implied. Since the  $n$  stable poles of  $g(s)$  contribute to a phase shift of  $-n \cdot \pi/2$  for  $\omega \rightarrow \infty$ , to get an opposite phase shift sufficiently high for  $\varphi$  to be below  $|\varphi| = \pi/2$ , all  $m$  zeros of  $z(s)$  must lie in the open left half complex plane where  $m \geq n - 1$  is necessary. That means for a transfer function  $g(s)$  to be strictly positive real it is necessary (but not sufficient) that  $g(s)$  is a stable minimum-phase system with  $n - 1$  or  $n$  zeros.

*Remark 8.1.* Going back to the original definition of positive realness (as for instance can be found in [133]) a rational function  $g(s)$  is defined to be strictly positive real if and only if  $g(s - \epsilon)$  is *positive real* for all  $\epsilon > 0$ . It follows that for a proper rational transfer function  $g(s)$  to be strictly positive real it is necessary and sufficient that  $d(s)$  is Hurwitz,  $\operatorname{Re} g(j\omega) > 0$  for all  $\omega \in \mathbb{R}$  (as in Definition 8.1) and in addition that  $\lim_{\omega^2 \rightarrow \infty} \omega^2 \operatorname{Re} |g(j\omega)| > 0$ .  $\square$

First, consider the standard feedback loop of Fig. 8.6 for known transfer functions

$$\begin{aligned} y(s) &= g(s)u(s) \\ u(s) &= -\hat{g}(s)y(s) + w(s) \end{aligned} \quad (8.2.4)$$

where as usual  $g$  represents a plant and  $\hat{g}$  a controller. We assume  $g(s)$  to be strictly positive real. With the assumption of a stable transfer function  $\hat{g}(s)$  in the feedback loop of Fig. 8.6, the Nyquist stability criterion says that the closed-loop system is stable if and only if the Nyquist plot of the open-loop system

$$g_0(s) := \hat{g}(s)g(s) \quad (8.2.5)$$

does not encircle the critical point  $-1$  counterclockwise. Since the phase of  $g_0$  is given by adding the phases of  $g$  and  $\hat{g}$ , this is guaranteed by the assumption of the  $\hat{g}$ -phase to be less than  $\pi/2$  for all  $\omega \in \mathbb{R}$ . But as we already know this is true for a strictly positive real transfer function  $\hat{g}$ . With the above considerations we can state an important stability result for the feedback configuration of Fig. 8.6:

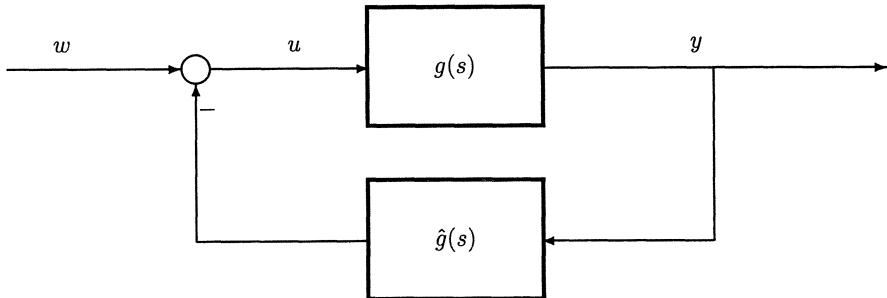


Fig. 8.6: Standard-feedback loop

*Theorem 8.4.*

The closed-loop system of Fig. 8.6 is stable if the two transfer functions  $g$  and  $\hat{g}$  are both strictly positive real.

$\square$

Now we come back to the problem of robust stability. For that, consider a family of uncertain transfer functions  $g(s, q)$  and/or  $\hat{g}(s, q)$  in the feedback loop of Fig. 8.6. From

Theorem 8.4 the closed-loop system is guaranteed to be stable for all members of the two families of uncertain systems if all of them are strictly positive real. Consider now an uncertain transfer function of the interval type which in addition is assumed to be robustly stable. For such a stable interval transfer function to prove strict positive realness is very simple and the result is given in the following theorem, [51, 45].

*Theorem 8.5.*

A family  $G_{Is}$  of transfer functions, generated by a robustly stable interval transfer function  $g(s, \mathbf{n}, \mathbf{d}) = n(s, \mathbf{n})/d(s, \mathbf{d})$ , is strictly positive real if and only if the following eight transfer functions of the Kharitonov plant family  $G_{Is}^K$  of  $G_{Is}$  are strictly positive real (for the notation see (8.1.8)):

$$\begin{aligned} g^{(1)}(s) &:= n^{(2)}(s)/d^{(1)}(s), & g^{(2)}(s) &:= n^{(3)}(s)/d^{(1)}(s), \\ g^{(3)}(s) &:= n^{(1)}(s)/d^{(2)}(s), & g^{(4)}(s) &:= n^{(4)}(s)/d^{(2)}(s), \\ g^{(5)}(s) &:= n^{(1)}(s)/d^{(3)}(s), & g^{(6)}(s) &:= n^{(4)}(s)/d^{(3)}(s), \\ g^{(7)}(s) &:= n^{(2)}(s)/d^{(4)}(s), & g^{(8)}(s) &:= n^{(3)}(s)/d^{(4)}(s). \end{aligned} \quad (8.2.6)$$

□

*Proof.*

The value sets of both the numerator  $n(s, \mathbf{n})$  and the denominator  $d(s, \mathbf{d})$  for a fixed  $s = j\omega^*$  are rectangles parallel to the coordinate axes in the complex plane. The phase condition of a stable transfer function

$$g(s) = \frac{n(s)}{d(s)} = \frac{|n(s)|}{|d(s)|} \cdot e^{j(\varphi_n - \varphi_d)} \quad (8.2.7)$$

to be strictly positive real is

$$|\varphi_c| := |\varphi_n - \varphi_d| < \frac{\pi}{2} \quad (8.2.8)$$

This means that for a stable interval transfer function to be strictly positive real both value sets  $\mathcal{N}(j\omega^*, \mathbf{n})$  and  $\mathcal{D}(j\omega^*, \mathbf{d})$  must be included in a  $\pi/2$ -sector, see Fig. 8.7. This certainly is guaranteed if the 16 stable transfer functions (8.1.9) of  $G_{Is}^K$  satisfy this phase condition. If the  $\pi/2$ -sector is contained in the right half complex plane, then the maximal phase difference is determined by  $n^{(3)}$  and  $d^{(4)}$  as in Fig. 8.7, or by  $d^{(3)}$  and  $n^{(4)}$ . For the  $\pi/2$ -sector in the upper, in the left and in the lower complex plane we get the remaining polynomial combinations of (8.2.6).

□

With Theorem 8.4 and Theorem 8.5 the closed-loop system of Fig. 8.6 is robustly stable for the two families  $G_{Is}$  and  $\hat{G}_{Is}$  generated by stable interval transfer functions if the eight transfer functions (8.2.6) of  $G_{Is}^K$  and the respective eight transfer functions of  $\hat{G}_{Is}^K$  are strictly positive real.

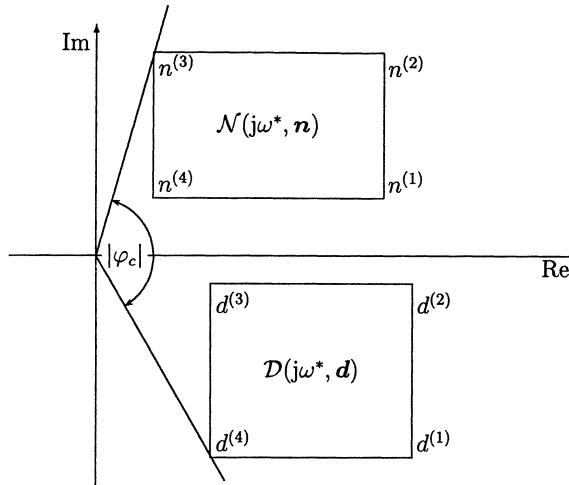


Fig. 8.7: Value set of the numerator,  $\mathcal{N}(j\omega^*, \mathbf{n})$  and the denominator,  $\mathcal{D}(j\omega^*, \mathbf{d})$

*Remark 8.2.* For a system with an equal number of multiple inputs and multiple outputs the notation of strict positive realness has also been defined, [133]: A quadratic transfer matrix  $\mathbf{G}(s)$  with proper rational elements is called strictly positive real if all elements of it have no poles in the closed right half complex plane and if the matrix  $\mathbf{G}(j\omega) + \mathbf{G}^T(-j\omega)$  is positive definite Hermite for all  $\omega \in \mathbb{R}$ . Furthermore, equivalent conditions for strict positive realness for a system in state space form can be given, [133]. For a completely controllable and observable time invariant system

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{Du}\end{aligned}\tag{8.2.9}$$

with an equal number of inputs and outputs, the transfer matrix

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}\tag{8.2.10}$$

is strictly positive real if and only if there exist matrices  $\mathbf{L}$ ,  $\mathbf{V}$  and  $\mathbf{P}$  with  $\mathbf{L}$  nonsingular and  $\mathbf{P}$  positive definite such that the following equations hold:

$$\begin{aligned}\mathbf{PA} + \mathbf{A}^T\mathbf{P} &= -\mathbf{LL}^T \\ \mathbf{C}^T &= \mathbf{PB} + \mathbf{LV} \\ \mathbf{D}^T + \mathbf{D} &= \mathbf{V}^T\mathbf{V}\end{aligned}\tag{8.2.11}$$

□

*Remark 8.3.* Modelling mechanical systems as for instance large flexible structures by finite element methods leads to linear dynamical equations of the form

$$\mathbf{M}\ddot{\mathbf{p}} + \tilde{\mathbf{D}}\dot{\mathbf{p}} + \mathbf{K}\mathbf{p} = \tilde{\mathbf{B}}\mathbf{u}\tag{8.2.12}$$

with a positive definite mass matrix  $\mathbf{M}$ , a damping matrix  $\tilde{\mathbf{D}}$ , and a symmetric stiffness matrix  $\mathbf{K}$ . The components of  $\mathbf{p}$  represent generalized displacements and those of  $\mathbf{u}$  generalized forces coming into the system through the input matrix  $\tilde{\mathbf{B}}$ . By introducing the state vector  $\mathbf{x} = (\mathbf{p}, \mathbf{v})$  with  $\mathbf{v} = \dot{\mathbf{p}}$ , (8.2.12) can be brought to the standard form of first order differential equations

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\tilde{\mathbf{D}} \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ \tilde{\mathbf{B}} \end{pmatrix} \mathbf{u} =: \mathbf{Ax} + \mathbf{Bu} \quad (8.2.13)$$

With the above assumptions on  $\mathbf{M}$  and  $\mathbf{K}$  it can be shown [127] that a matrix  $\Phi$  exists such that

$$\Phi^T \mathbf{M} \Phi = \mathbf{I}, \quad \Phi^T \mathbf{K} \Phi = \Omega^2 = \text{diag}\{\omega_i^2\} \quad (8.2.14)$$

If  $\Phi$  can be chosen such that in addition

$$\Phi^T \mathbf{D} \Phi = \Delta = \text{diag}\{\zeta_i\} \quad (8.2.15)$$

and if collocated sensors are used, with the system theoretical meaning that there exists a positive definite matrix  $\mathbf{P}$  such that

$$\mathbf{C}^T = \mathbf{PB} \quad (8.2.16)$$

the overall transfer matrix

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \quad (8.2.17)$$

can be shown to be (strictly) positive real [26]. Furthermore, this property is robust against changes in the uncertain positive coefficients of  $\mathbf{M}$ ,  $\tilde{\mathbf{D}}$ , and  $\mathbf{K}$ . That means for such a class of uncertain systems the robust positive realness is already guaranteed by the underlying structure of the system equations. For further details we refer to [26].

□ By Definition 8.1 a family  $G_s$  of stable transfer functions is strictly positive real if

$$\inf_{g \in G_s} \inf_{\omega} \text{Re } g(j\omega, \mathbf{q}) \geq 0 \quad (8.2.18)$$

and

$$\text{Re } g(j\omega, \mathbf{q}) \neq 0, \quad \forall \omega \in \mathbb{R}, \quad \forall g \in G_s \quad (8.2.19)$$

The proof of Theorem 8.5 easily leads to the following theorem given in [45]:

*Theorem 8.6.*

For a family  $G_{Is}$  of transfer functions, generated by a stable interval transfer function the infimum of  $\text{Re } g(j\omega, \mathbf{q})$  is determined over all  $\omega$  and all  $g \in G_{Is}$  by one of the 16 transfer functions of the Kharitonov plant family  $G_{Is}^K$  of  $G_{Is}$

$$\inf_{g \in G_{Is}} \inf_{\omega} \text{Re } g(j\omega, \mathbf{q}) = \inf_{g \in G_{Is}^K} \inf_{\omega} \text{Re } g(j\omega, \mathbf{q}) \quad (8.2.20)$$

□

Note that for the right hand side of (8.2.20) the finite Kharitonov system  $G_{Is}^K$  of 16 transfer functions of  $G_{Is}$  is used to determine the infimum, whereas for the left hand side the transfer functions of  $G_{Is}$  (with an infinite number of members) is considered. The transfer function which determines the infimum in (8.2.20) does not necessarily belong to one of the above eight transfer functions of (8.2.6). Theorem 8.6 will be used in Section 8.4.

*Example 8.2.* For the bus model introduced in Section 1.2 consider the transfer function from the steering angle  $\delta_f$  to the yaw rate  $r$ ,

$$\frac{r}{\delta_f} = \frac{b_0(\tilde{m}, v) + b_1(\tilde{m})s}{a_0(\tilde{m}, v) + a_1(\tilde{m}, v)s + s^2}$$

with

$$\begin{aligned} b_0 &= \frac{48031}{\tilde{m}^2 v}, & b_1 &= \frac{66.9733}{\tilde{m}} \\ a_0 &= \frac{16.6304}{\tilde{m}} + \frac{268973}{\tilde{m}^2 v^2}, & a_1 &= \frac{1075.15}{\tilde{m}v} \end{aligned}$$

The analytical expression for  $\operatorname{Re} g(j\omega, \tilde{m}, v)$  is

$$\operatorname{Re} g(j\omega, \tilde{m}, v) = \frac{[a_0(\tilde{m}, v) - \omega^2] b_0(\tilde{m}, v) + a_1(\tilde{m}, v) b_1(\tilde{m}) \omega^2}{[a_0(\tilde{m}, v) - \omega^2]^2 + a_1^2(\tilde{m}, v) \omega^2}$$

and it can easily be checked that

$$\operatorname{Re} g(j\omega, \tilde{m}, v) > 0$$

for all positive values  $\tilde{m}, v$ . Thus, the above transfer function is strictly positive real independent on the virtual mass  $\tilde{m} > 0$  and on the velocity  $v > 0$ .  $\square$

### 8.3 Tree Structured Transfer Functions

In this section we give up the assumption of a transfer function to be of the interval type or to be (strictly) positive real. Instead, we ask if the value set of an uncertain transfer function can easily be constructed by extending the idea of a tree structured uncertain polynomial which was introduced in Section 6.5. The stability test of a control system with many uncertain parameters can be performed extremely fast if the characteristic polynomial has a tree structure as was shown in Chapter 6. Tree structures arise naturally if several subsystems with independent parameters are connected to an overall feedback system.

*Example 8.3.* In this example we assume that the transfer functions have disjoint parameter sets in the numerator and denominator polynomials and that different transfer functions have no uncertain parameters in common.

In the feedback loop of Fig. 8.8 the closed-loop characteristic polynomial ( $\mathbf{n} := [\mathbf{n}_1^T \mathbf{n}_2^T \mathbf{n}_c^T]^T$ ,  $\mathbf{d} := [\mathbf{d}_1^T \mathbf{d}_2^T \mathbf{d}_c^T]^T$ )

$$\begin{aligned} p(s, \mathbf{n}, \mathbf{d}) = & n_1(s, \mathbf{n}_1) n_2(s, \mathbf{n}_2) n_c(s, \mathbf{n}_c) + \\ & + d_1(s, \mathbf{d}_1) d_2(s, \mathbf{d}_2) d_c(s, \mathbf{d}_c) \end{aligned} \quad (8.3.1)$$

has a tree structure. However, the characteristic polynomial of the feedback loop in

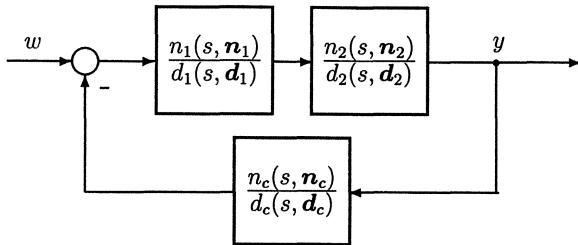


Fig. 8.8: Serial connection with feedback

Fig. 8.9 does not have a tree structure:

$$\begin{aligned} p(s, \mathbf{n}, \mathbf{d}) = & [n_1(s, \mathbf{n}_1)d_2(s, \mathbf{d}_2) + n_2(s, \mathbf{n}_2)d_1(s, \mathbf{d}_1)]n_c(s, \mathbf{n}_c) + \\ & + d_1(s, \mathbf{d}_1)d_2(s, \mathbf{d}_2)d_c(s, \mathbf{d}_c) \end{aligned} \quad (8.3.2)$$

The parameters  $\mathbf{d}_1$  and  $\mathbf{d}_2$  appear in two terms of  $p$ . However, in the characteristic

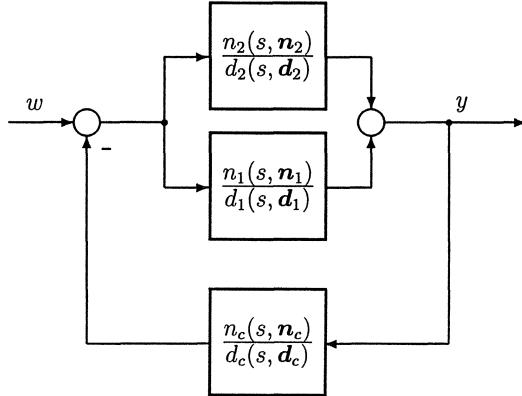


Fig. 8.9: Parallel connected transfer functions with feedback

equation of this feedback loop

$$\begin{aligned} r(s, \mathbf{n}, \mathbf{d}) &= 1 + g(s, \mathbf{n}, \mathbf{d}) \\ &= 1 + \left[ \frac{n_1(s, \mathbf{n}_1)}{d_1(s, \mathbf{d}_1)} + \frac{n_2(s, \mathbf{n}_2)}{d_2(s, \mathbf{d}_2)} \right] \frac{n_c(s, \mathbf{n}_c)}{d_c(s, \mathbf{d}_c)} \end{aligned} \quad (8.3.3)$$

where  $g(s, \mathbf{n}, \mathbf{d})$  is the open-loop transfer function, each uncertainty vector appears just once, either in a numerator or in a denominator polynomial. Thus, the value set of the open-loop transfer function  $g(s, \mathbf{n}, \mathbf{d})$  can be constructed sequentially, if we introduce

the operation of complex set inversion. Also the closed-loop transfer function

$$g_c(s, \mathbf{n}, \mathbf{d}) := \frac{1}{\frac{n_c(s, \mathbf{n}_c)}{d_c(s, \mathbf{d}_c)} + \frac{1}{\frac{n_1(s, \mathbf{n}_1)}{d_1(s, \mathbf{d}_1)} + \frac{n_2(s, \mathbf{n}_2)}{d_2(s, \mathbf{d}_2)}}} \quad (8.3.4)$$

can be constructed sequentially from the value sets of the three subsystems.

□

*Example 8.4.* The transfer function of the crane with full state feedback from input  $u$  to the position  $x_1$  is

$$\frac{x_1(s)}{u(s)} = \frac{g + \ell s^2}{gk_1 + gk_2s + (k_1\ell - k_3 + g(m_C + m_L))s^2 + (k_2\ell - k_4)s^3 + \ell m_C s^4}$$

The numerator is unchanged from the open-loop transfer function (1.1.10). The denominator has been changed by state feedback to the form (2.2.18). The numerator and denominator have the uncertain parameter rope length  $\ell$  in common. Dividing, both the numerator and the denominator polynomial by the numerator we get a continued fraction with each uncertain parameter appearing just once.

$$\frac{x_1(s)}{u(s)} = \frac{1}{k_1 + k_2s + m_C s^2 + \frac{gm_L s^2 - (k_3 s^2 + k_4 s)s^2}{g + \ell s^2}}$$

Thus, the value set of the closed-loop transfer function can be constructed sequentially.

□

For a stability test the zero exclusion principle is now applied to the denominator of the closed-loop transfer function. This denominator must not be in a polynomial form. The two examples above show that a rational form may be more advantageous for the sequential construction of the value set. A standard form of the characteristic equation is  $1 + g(s, \mathbf{q})$ , where  $g(s, \mathbf{q})$  is the open-loop transfer function. Remember that  $g(s, \mathbf{q})$  is not necessarily given in form of a ratio of two polynomials. The two examples have led to a continued fraction form. Any other nested rational form would be feasible too. The important point for the construction of the value set of  $g(s, \mathbf{q})$  is only that it is in a form such that each uncertain parameter appears only once.

### Nyquist Value Sets

It is obvious that the classical Nyquist stability test for a fixed transfer function can easily be extended to an uncertain transfer function: The closed-loop system is stable, i.e. the roots of  $1 + g(s, \mathbf{q})$  lie in the left half  $s$ -plane, if

- there exists a  $\mathbf{q}^* \in Q$  such that  $1 + g(s, \mathbf{q}^*)$  is stable and
- the critical point  $(-1)$  is not included in the value set of  $g(j\omega, \mathbf{q})$ :

$$-1 \notin \mathcal{G}(j\omega, Q) = \{g(j\omega, \mathbf{q}) \mid \omega \in [0; \infty), \mathbf{q} \in Q\} \quad (8.3.5)$$

The Nyquist set for the open-loop transfer function gives information about the stability margin of a system. The minimal distance of the Nyquist set from the critical point is a direct measure for the stability margin.

For constructing a value set of the uncertain transfer function  $n(s, \mathbf{n})/d(s, \mathbf{d})$ , the set  $\mathcal{N}(j\omega, Q_n)$  has to be divided by  $\mathcal{D}(j\omega, Q_d)$ . This is done by inverting the set  $\mathcal{D}(j\omega, Q_d)$  and then multiplying the resulting set  $1/\mathcal{D}(j\omega, Q_d)$  with  $\mathcal{N}(j\omega, Q_n)$ . Value set multiplication was already defined in Section 6.2. Thus, we have still to show the inversion of a complex set. The inversion of a complex set  $\mathcal{A} \subset \mathbb{C}, 0 \notin \mathcal{A}$  is defined as

$$\mathcal{A}^{-1} = \{1/a \mid a \in \mathcal{A}\} \quad (8.3.6)$$

Each point of the set  $\mathcal{A}$  has to be inverted. If zero is excluded from  $\mathcal{A}$ , then the inversion  $w = 1/z$  is a bijective and continuous function. An interior point of  $\mathcal{A}$  is mapped to an interior point of  $\mathcal{A}^{-1}$ . A boundary point of  $\mathcal{A}$  is mapped to a boundary point of  $\mathcal{A}^{-1}$ . Thus, only the boundary of  $\mathcal{A}$  has to be inverted to get the boundary of  $\mathcal{A}^{-1}$ ,

$$\partial(\mathcal{A}^{-1}) = (\partial\mathcal{A})^{-1}, \quad (0 \notin \mathcal{A}) \quad (8.3.7)$$

In a polar coordinate representation of the inversion map applied to a complex point  $z = |z|e^{j\varphi}$ , we obtain the inverse

$$w = \frac{1}{|z|}e^{-j\varphi} \quad (8.3.8)$$

It can easily be seen that the interior (exterior) of the unit circle in the  $z$ -plane is mapped onto the exterior (interior) of the unit circle in the  $w$ -plane. The unit circle itself is mapped onto the unit circle in the  $w$ -plane. Written in real and imaginary parts of  $z = x + jy$  and  $w = u + jv$  the mapping equation is

$$w = u + jv = \frac{1}{x + jy} = \frac{x}{x^2 + y^2} + j \frac{-y}{x^2 + y^2} \quad (8.3.9)$$

and thus

$$u = \frac{x}{x^2 + y^2}, \quad v = \frac{-y}{x^2 + y^2} \quad (8.3.10)$$

and conversely

$$x = \frac{u}{u^2 + v^2}, \quad y = \frac{-v}{u^2 + v^2} \quad (8.3.11)$$

For a computer-aided value set construction, set boundaries are approximated by polygons. Therefore, we are mainly interested how a line segment is transformed by the inversion map.

We consider all points  $z = x + jy$  on the straight line determined by the equation

$$2ax + 2by = 1 \quad (8.3.12)$$

Substituting  $x$  and  $y$  from (8.3.11) yields

$$2a \frac{u}{u^2 + v^2} + 2b \frac{-v}{u^2 + v^2} = 1$$

and after some elementary transformations

$$(u - a)^2 + (v + b)^2 = a^2 + b^2 \quad (8.3.13)$$

This is the equation of a circle with center  $(a, -b)$  and radius  $\sqrt{a^2 + b^2}$ . The circle always passes through the origin  $(0, 0)$  of the coordinate system. The point  $w = 0$  is the image of the point at  $z = \infty$  on the line. The resulting circle may also be determined easily with two additional points on the line to be mapped. Since we want to map a line segment with endpoints  $z_1$  and  $z_2$ , these points are the natural choice. The three points determine the circle, see Fig. 8.10. The image of the line segment  $\overline{z_1 z_2}$  is the arc not containing the origin.

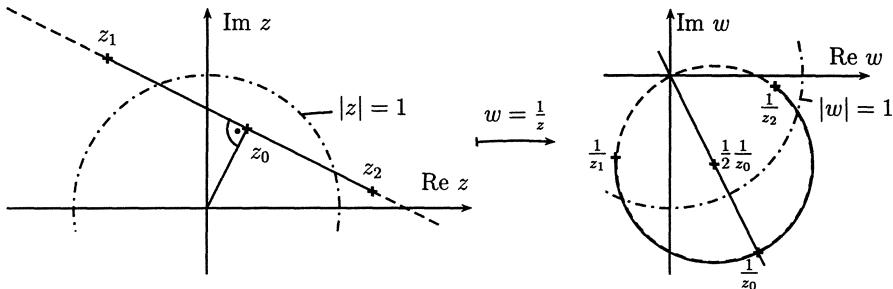


Fig. 8.10: Inversion of a line segment

*Example 8.5.* We consider the mechanical system given in Fig. 6.15. In Example 6.10 the value set of the characteristic polynomial of the system was constructed. In the example it was mentioned that the system is passive, which implies Hurwitz stability for arbitrary positive parameter values. However, if for instance the damping of the system is not sufficiently high, feedback control has to be introduced. For this example we consider the transfer function from input  $u$  to the position  $x_1$  of the mass  $m_1$

$$\frac{x_1(s)}{u(s)} = \frac{c_{12}}{p_1(s, \mathbf{q}^{I_1}, c_{12}) \cdot p_2(s, \mathbf{q}^{I_2}, c_{12}) - c_{12}^2} =: \frac{c_{12}}{p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12})}$$

with  $p_1(s, \mathbf{q}^{I_1}, c_{12})$  and  $p_2(s, \mathbf{q}^{I_2}, c_{12})$  given in (6.5.18). The parameter intervals were given as follows:  $m_1 \in [1; 3]$ ,  $d_1 \in [0.5; 2]$ ,  $c_1 \in [1; 2]$ ,  $m_2 \in [2; 5]$ ,  $d_2 \in [0.5; 2]$ ,  $c_2 \in [2; 4]$ . In [18] the controller

$$\frac{n_c(s)}{d_c(s)} = 471250 \frac{0.5 + 1.9s + 1.7s^2 + s^3}{19000 + 1450s + 62s^2 + s^3}$$

was proposed for position control of the mass  $m_1$ . For that controller we get the closed loop transfer function

$$F_c(s) = \frac{\frac{n_c(s)}{d_c(s)} \frac{c_{12}}{p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12})}}{1 + \frac{n_c(s)}{d_c(s)} \frac{c_{12}}{p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12})}} = \frac{n_c(s)c_{12}}{d_c(s)p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12}) + d_c(s)c_{12}} \quad (8.3.14)$$

The controller was determined by pole placement for the center of the  $Q$ -box. A stability analysis of the entire operating domain has to be performed. The characteristic polynomial  $p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12})$  has a tree structure if the parameter  $c_{12}$  is considered to be fixed,  $c_{12} = c_{12}^*$ . (If  $c_{12}$  is uncertain then this one parameter must be gridded.) Also the closed-loop characteristic polynomial has a tree structure for fixed  $c_{12}$  even if the compensator coefficients are uncertain. The characteristic equation

$$1 + \frac{n_c(s)}{d_c(s)} \frac{c_{12}^*}{p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12}^*)}$$

has a rational tree structure. In the following the value set of the open-loop transfer function is constructed and checked for exclusion of the critical point  $-1$ .

For a fixed frequency  $\omega = \omega^*$  the value set of the characteristic polynomial  $p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12}^*)$  is constructed as demonstrated in Example 6.10. The resulting set is inverted and multiplied by  $c_{12}^*$ . The value set for  $\omega^* = 1$  and  $c_{12}^* = 1$  is displayed in Fig. 8.11. To obtain the value set of the open-loop transfer function this set has to

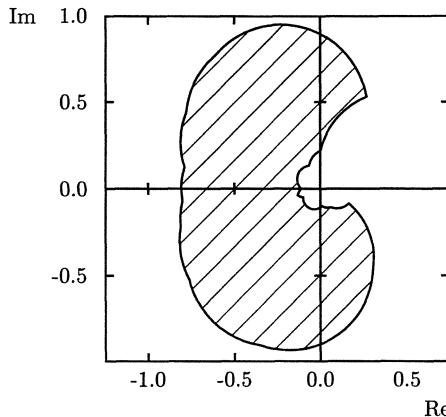


Fig. 8.11: Plant transfer function set for  $\omega = 1$

be multiplied by the transfer function of the controller. For a fixed frequency this is a complex number. In Fig. 8.12 the union of the Nyquist sets of the open-loop transfer function

$$\frac{n_c(s)}{d_c(s)} \frac{c_{12}^*}{p_p(s, \mathbf{q}^{I_1}, \mathbf{q}^{I_2}, c_{12}^*)}$$

are displayed. Fig. 8.13 shows a detailed view of the neighborhood of the critical point  $-1$ . The critical point  $-1$  is excluded from the union of the sets. Hence, the closed-loop system is robustly stable. The minimal distance of the Nyquist sets from the critical point is a measure for the stability margin of the closed-loop system. It can be determined graphically by enlarging a circle around the point  $-1$  until it touches the boundary of the union of the Nyquist sets. In the given example the radius of this circle is approximately 0.24.

□

With the use of transfer functions for value set operations tree structures can be exploited in complex control system structures. For more complicated networks Mason's formula [181] can be used to compute the characteristic equation. Similar as in the polynomial context, a basic rule for modelling of structured systems is:

*Never execute additions or multiplications of transfer functions with disjoint uncertainties.*

The tree structured decomposition of polynomials and transfer functions is a powerful tool for robustness analysis. It can handle an extremely large number of uncertain parameters as demonstrated in the following example.

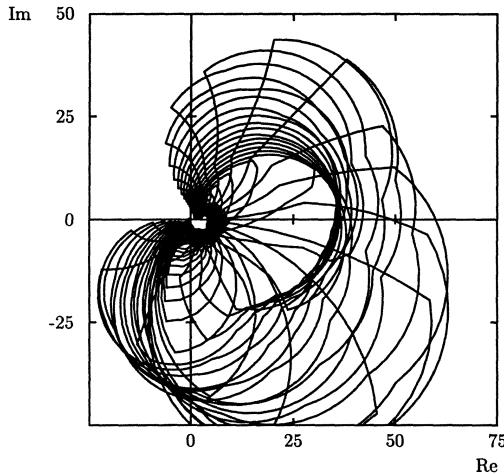


Fig. 8.12: Union of Nyquist sets

*Example 8.6.* Given is a chain of masses connected with springs and dampers as displayed in Fig. 8.14, [19]. The following problems have to be solved:

1. Does there exist a tree structured decomposition of the mass-spring-damper system?
2. If so, construct the value set for  $N = 17$ ,  $m_i \in [1; 2]$ ,  $a_i \in [2; 5]$ , and  $d_i \in [10; 12.5]$ .

The differential equations of the system are

$$\begin{aligned}
 m_1\ddot{x}_1 + d_1\dot{x}_1 + c_1x_1 - d_2s\dot{x}_2 - c_2x_2 &= 0 \\
 m_2(\ddot{x}_1 + \ddot{x}_2) + d_2\dot{x}_2 + c_2x_2 - d_3\dot{x}_3 - c_3x_3 &= 0 \\
 &\vdots && \vdots \\
 m_i \sum_{j=1}^i \ddot{x}_j + d_i\dot{x}_i + c_ix_i - d_{i+1}\dot{x}_{i+1} - c_{i+1}x_{i+1} &= 0 \\
 &\vdots && \vdots \\
 m_N \sum_{j=1}^N \ddot{x}_j + d_N\dot{x}_N + c_Nx_N &= u
 \end{aligned} \tag{8.3.15}$$

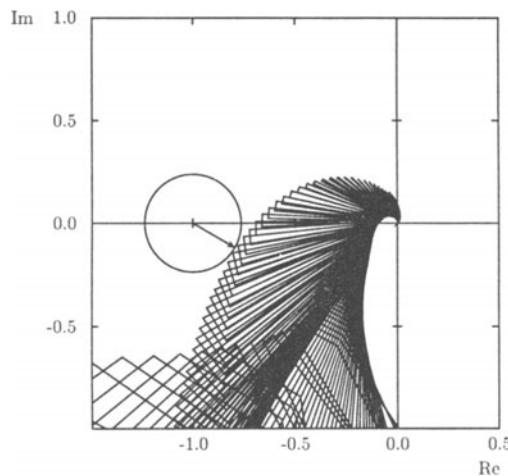


Fig. 8.13: Detailed view of the union of Nyquist sets

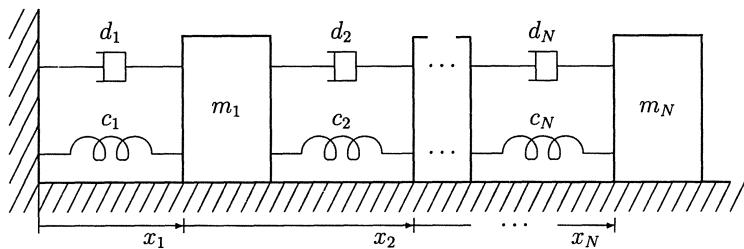


Fig. 8.14: Chain of masses connected by springs and dampers

and its Laplace transform is

$$\mathbf{M}(s, \mathbf{q}) \cdot \begin{bmatrix} x_1(s) \\ x_2(s) \\ \vdots \\ x_i(s) \\ \vdots \\ x_N(s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ u(s) \end{bmatrix} \quad (8.3.16)$$

with

$$\mathbf{M}(s, \mathbf{q}) = \begin{bmatrix} a_1 + 1/b_1 & -1/b_2 & 0 & \dots & 0 \\ a_2 & a_2 + 1/b_2 & -1/b_3 & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ a_i & \dots & a_i + 1/b_i & -1/b_{i+1} & 0 & \dots & 0 \\ \vdots & & & & & & \vdots \\ a_N & a_N & \dots & & a_N + 1/b_N & & \end{bmatrix} \quad (8.3.17)$$

and

$$\begin{aligned} a_i(s, m_i) &= m_i s^2 \\ 1/b_i(s, c_i, d_i) &= d_i s + c_i, \quad i = 1, \dots, N \end{aligned} \quad (8.3.18)$$

A rational form of the characteristic equation of the system (8.3.16) is the determinant of the matrix (8.3.17).

$$d(s, \mathbf{q}) = \det \mathbf{M}(s, \mathbf{q}) \quad (8.3.19)$$

After some manipulations the characteristic equation can be written as

$$d(s, \mathbf{q}) = \prod_{i=1}^N t_i(s, \mathbf{q}^{I_i}) \quad (8.3.20)$$

with

$$t_i(s, \mathbf{q}^{I_i}) = a_i + 1/b_i + \cfrac{1}{b_{i+1} + \cfrac{1}{a_{i+1} + \cfrac{1}{b_{i+2} + \dots + a_{N-1} + \cfrac{1}{1/a_N + b_N}}}} \quad (8.3.21)$$

and

$$\mathbf{q}^{I_i} = [q_i \ q_{i+1} \ \dots \ q_N]^T \quad (8.3.22)$$

The terms  $t_i(s, \mathbf{q}^{I_i})$  may be written as

$$t_i(s, \mathbf{q}^{I_i}) = w_i(s, \mathbf{q}^{I_i}) + 1/b_i \quad (8.3.23)$$

where  $w_i(s, \mathbf{q}^{I_i})$  is computed recursively by

$$\begin{aligned} w_N(s, \mathbf{q}^{I_i}) &= a_N \\ w_{i-1}(s, \mathbf{q}^{I_i}) &= a_{i-1} + \frac{1}{b_i + 1/w_i} \end{aligned} \quad (8.3.24)$$

This kind of representation of the characteristic equation in (8.3.20) is not a tree structured decomposition in its original sense. The index sets of the factors  $t_i$  do not have disjoint parameter sets. However, the zero exclusion from the value set can be checked: The value set  $\mathcal{D}(j\omega, \mathbf{q})$  does not contain the origin, if none of the value sets  $\mathcal{T}_i(s, Q^{I_i})$  contains the origin,

$$0 \notin \mathcal{T}_i(j\omega, Q^{I_i}), \quad i = 1, \dots, N \quad (8.3.25)$$

Each factor  $t_i(s, \mathbf{q}^{I_i})$  has a tree structure, i.e. each uncertain parameter appears just once in the continued fraction (8.3.21). Therefore, the value sets of each of the  $t_i(s, \mathbf{q}^{I_i})$  can be constructed sequentially fast and then be checked for zero exclusion.

Physically, the test of  $\prod_{i=1}^N t_i(s, \mathbf{q}^{I_i})$  is a stability test for a subsystem with the masses  $j, j+1, \dots, N$  where the mass  $j-1$  is inertially fixed. For  $j=1$  the entire system of Fig. 8.14 is tested. Thus, the test of  $\mathcal{T}_1$  is the most comprehensive one.

In Fig. 8.15 the union of  $\mathcal{T}_1(j\omega^*, \mathbf{q})$  is displayed for a frequency grid with  $\omega = i \cdot 0.1$ ,  $i = 1, 2, \dots$ . The picture was generated for 17 masses, i.e. 51 uncertain parameters. The

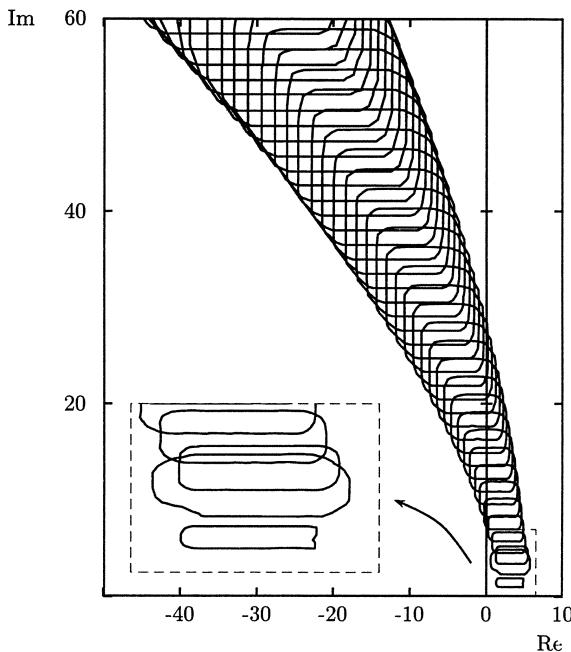


Fig. 8.15: Union of the value sets of  $\mathcal{T}_1(j\omega^*, \mathbf{q})$

sets for small frequencies are displayed magnified in the clipped window. They clearly show that some sets are not convex. The computation time per frequency grid point was eight seconds on a Hewlett Packard 9000/425t. Brute force gridding would have required  $10^{51} \cdot 1\text{ms} \approx 10^{35}$  years, if 10 gridding points per dimension and 1 millisecond per stability test are assumed. For stability analysis also the sets  $\mathcal{T}_{17}(j\omega^*, \mathbf{q})$  to  $\mathcal{T}_2(j\omega^*, \mathbf{q})$  have to be checked.

□

## 8.4 Robustness with Respect to Sector Nonlinearities

In this section we consider the feedback loop of the system

$$\begin{aligned} y(s) &= g(s)u(s) \\ u &= -f(y) \end{aligned} \quad (8.4.1)$$

which is shown in Fig. 8.16. In the feedforward path there is a rational transfer function

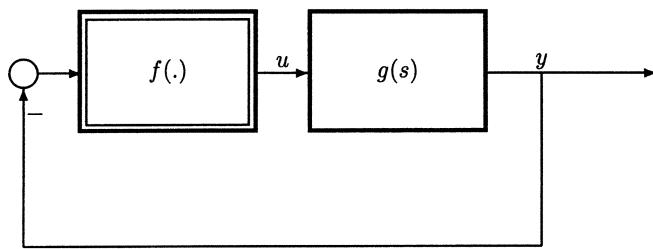


Fig. 8.16: Nonlinear function in the feedback loop

$$g(s) = \frac{n(s)}{d(s)} \quad (8.4.2)$$

with numerator

$$n(s) = n_0 + n_1 s + \dots + n_m s^m \quad (8.4.3)$$

and denominator

$$d(s) = d_0 + d_1 s + \dots + d_n s^n \quad (8.4.4)$$

Let us first assume that  $g(s)$  is known. However, in the feedback path the function  $f$  is considered to be uncertain and is only assumed to be continuous and to belong to the class of functions

$$\mathcal{F}_k := \{f(\cdot) \mid f(0) = 0, 0 \leq yf(y) \leq ky^2\}, \quad 0 < k < \infty \quad (8.4.5)$$

That means it is only known that the graph of  $f$  is within a sector  $[0 ; k]$ , limited below by the abscissa and limited above by the line

$$f(y) = k y \quad (8.4.6)$$

We call this sector the  $k$ -sector, with notation  $[0 ; k]$ . In the  $k$ -sector the graph of the continuous function  $f$  can be arbitrary, see Fig. 8.17. If for a given transfer function  $g$  the above closed-loop system is stable it is called *absolutely stable*. A classical result for absolute stability is the sufficient *Popov criterion* [140, 154, 172]:

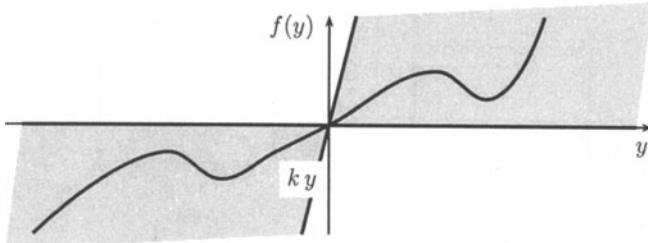


Fig. 8.17: Sector of nonlinearity

*Theorem 8.7. (Popov)*

For  $f$  a continuous function in the class  $\mathcal{F}_k$  of (8.4.5) and  $g$  a strictly proper ( $m < n$ ) rational transfer function with all poles in the open left half complex plane, the feedback loop of Fig. 8.16 is absolutely stable if there exists a  $\theta \in \mathbb{R}$  such that the inequality

$$1/k + \operatorname{Re} \{(1 + j\omega\theta)g(j\omega)\} > 0, \quad \forall \omega \geq 0 \quad (8.4.7)$$

holds. (The scalar factor  $k$  has to be considered as an arbitrary but fixed value,  $0 \leq k < \infty$ .)

□

The classical Popov criterion represents a strong robustness result, since  $f$  is assumed to be a totally uncertain (continuous) function within a sector  $[0; k]$ . No matter what  $f$  inside this sector is, the closed-loop system is stable if a  $\theta$  exists which satisfies (8.4.7).

If  $g(s)$  is unstable, then the Popov criterion cannot hold for any  $k > 0$  because the zero function ( $f(y) = 0, \forall y \in \mathbb{R}$ ) is in  $\mathcal{F}_k$ . We can now ask for a sector  $[k_1; k_2]$  not containing the zero function which guarantees absolute stability. For instance, in [154, 172] it is shown that this more general problem can be transformed to the canonical problem given above: If the open loop transfer function  $g(s)$  of Fig. 8.16 is unstable but can be stabilized with the constant feedback  $u = -\rho y$ ,  $\rho > 0$  then replace in the Popov criterion above the transfer function  $g(s)$  by  $\tilde{g}(s) = g(s)/[1 + \rho g(s)]$  and the nonlinear function  $f(\cdot)$  by  $\tilde{f}(\cdot) = (f - \rho)(\cdot)$ , see Fig. 8.18. If application of the Popov criterion to the transformed system yields a sector  $[0; k]$  for  $\tilde{f}$ , then the original system is absolutely stable for a sector  $[\rho; k + \rho]$ .

Values for  $k$  guaranteeing absolute stability can easily be determined graphically by means of the *Popov plot*  $g_p$ ,

$$g_p(j\omega) := \operatorname{Re} g(j\omega) + j\omega \operatorname{Im} g(j\omega), \quad \omega \geq 0 \quad (8.4.8)$$

of the stable transfer function  $g$ . Consider any straight line  $1/k + x + (-\theta)y = 0$  in the  $(x, y)$ -plane with  $k > 0$  cutting the negative  $x$ -axis at  $-1/k$  and with a slope of  $1/\theta$ . For all points on the right hand side of this line we have  $1/k + x + (-\theta)y > 0$ . If we set  $x = \operatorname{Re} g(j\omega)$ ,  $y = \omega \operatorname{Im} g(j\omega)$ , then

$$1/k + \operatorname{Re} g(j\omega) - \theta \omega \operatorname{Im} g(j\omega) > 0 \quad (8.4.9)$$

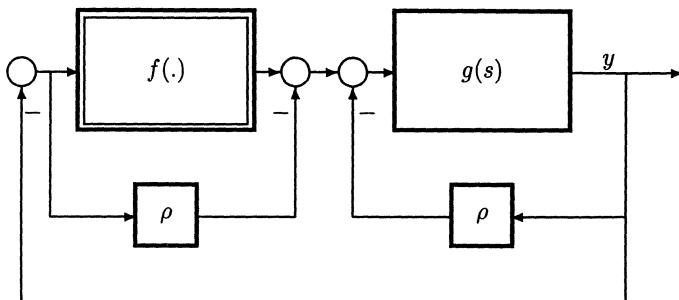


Fig. 8.18: Transformation to the standard form

for all  $\omega \geq 0$  if the Popov plot  $g_p(j\omega) := \operatorname{Re} g(j\omega) + j\omega \operatorname{Im} g(j\omega)$  lies on the right side of the above line. But (8.4.9) is identical with the expression (8.4.7). So we have the following result: Any straight line in the complex plane crossing the negative real axis at the point  $z < 0$  and with the property that the Popov plot lies totally on its right hand side yields a  $k = -1/z$  for a sector  $[0; k]$  of absolute stability. For such a  $k$  and with  $1/\theta$  as the slope of the corresponding line the Popov criterion (8.4.7) is satisfied. With the Popov plot we can easily find the maximal value for  $k$  with absolute stability. This sector is called the *Popov sector*. For instance,  $\tilde{k} = -1/\tilde{z}$  determines the Popov sector for a  $g$  with the Popov plot of Fig. 8.19 (with the corresponding slope  $1/\tilde{\theta} = \tan(\tilde{\alpha})$  of the line). But note that the Popov criterion is a sufficient condition: Hence we cannot exclude in general that for absolute stability a greater sector than the Popov sector exists.

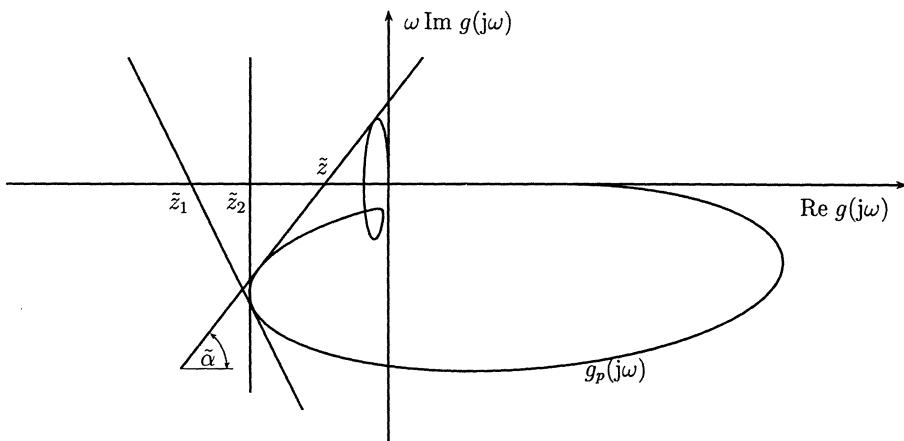


Fig. 8.19: Absolute stability and Popov sector

*Remark 8.4.* A good hypothesis for testing is the so called *Aizerman Conjecture*. It relates the Popov sector with the (Nyquist) sector of stable linear gains. The latter is obtained from the intersections of the Nyquist plot with the negative real axis. In view of (8.4.8) these intersections are identical with those of the Popov plot. A situation that frequently arises is shown in the Popov plot of Fig. 8.20. Here the tangent to the Popov plot in its most left real axis intersection is a feasible line for satisfaction of the Popov criterion, i.e. the Popov sector and the Nyquist sector have the same upper bound which in this case is obviously necessary and sufficient for absolute stability.

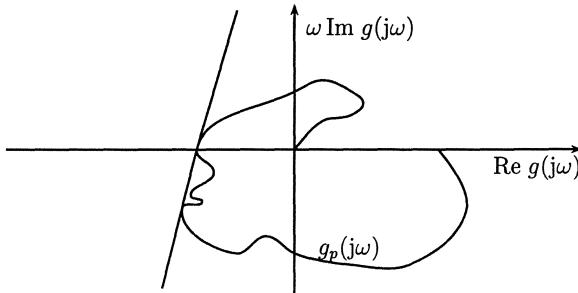


Fig. 8.20: An example for identical Nyquist and Popov sectors

□

If in the Popov criterion (8.4.7) the free parameter  $\theta$  is chosen to be zero (only vertical lines for the graphical Popov test are considered) we get the more conservative criterion

$$1/k + \operatorname{Re} g(j\omega) > 0, \quad \forall \omega \geq 0 \quad (8.4.10)$$

sufficient for absolute stability. However, this criterion is suitable for a robustness result where not only  $f$  but also the transfer function  $g$  can be considered as uncertain. The transfer function  $g$  is now assumed to be a stable interval transfer function, that means  $g$  is contained in the set  $G_{Is}$ ,

$$G_{Is} := \{g(s, \mathbf{q}) = g(s, \mathbf{n}, \mathbf{d}) = \frac{n(s, \mathbf{n})}{d(s, \mathbf{d})} \mid \mathbf{q} \in Q, d(s, \mathbf{d}) \text{ stable}\} \quad (8.4.11)$$

where  $Q$  is the box (8.1.3). Equation (8.2.20) of Theorem 8.6 and (8.4.10) lead to the following theorem of [45] for the determination of a  $k$ -sector for robust absolute stability:

*Theorem 8.8. (Chapellat, Dahleh, Bhattacharyya)*

The feedback loop of Fig. 8.16 with  $g \in G_{Is}$ , where the family of transfer functions  $G_{Is}$  is generated by a stable interval transfer function, is absolutely stable for

$$k < \infty \quad \text{if} \quad \inf_{G_{Is}^k} \inf_{\omega \in \mathbb{R}} \operatorname{Re} g(j\omega, \mathbf{q}) \geq 0 \quad (8.4.12)$$

and

$$k < -\frac{1}{\inf_{G_{I_s}^k} \inf_{\omega \in \mathbb{R}} \operatorname{Re} g(j\omega, \mathbf{q})} \quad (8.4.13)$$

otherwise.

□

*Remark 8.5.* For the standard feedback loop of Fig. 8.16, Tsyplkin and Polyak have investigated robust stability under the assumption of a frequency dependent uncertainty

$$|g(j\omega, \mathbf{q}) - g^0(j\omega)| \leq \gamma r(\omega) \quad (8.4.14)$$

where  $g^0(s) = g(s, \mathbf{q}^0)$  is a nominal plant transfer function,  $\gamma > 0$  is an uncertainty margin and  $r(\omega) \geq 0$  a given bounded scalar function. In [168] they have given a robust circle criterion, a Mikhailov-like criterion and a robustifying Popov criterion. □

In Theorem 8.8,  $G_{I_s}^k$  is the Kharitonov plant family (8.1.9) of  $G_{I_s}$  containing 16 transfer functions. For a family  $G_{I_s}$  generated by a stable interval transfer function where

$$\inf_{G_{I_s}^k} \inf_{\omega \in \mathbb{R}} \operatorname{Re} g(j\omega, \mathbf{q}) \geq 0 \quad (8.4.15)$$

is satisfied, (8.4.12) leads to the greatest possible  $k$ -sector for robust absolute stability (the complete first and third quadrant). For the case (8.4.13), we do not get in general the Popov sector for  $G_{I_s}$ , because we cannot exclude the existence of a  $\theta \neq 0$  which satisfies (8.4.10) for all  $g \in G_{I_s}$  with a  $k$ -value greater than the one given by (8.4.13). But for  $\theta \neq 0$ ,  $\tilde{g}(j\omega, \theta, \mathbf{q}) := (1 + j\omega\theta)g(j\omega, \mathbf{q})$  is no longer an interval transfer function to which Theorem 8.8 can be applied. For a more complicated dependency of the plant transfer function  $g(s, \mathbf{q})$  on the uncertain parameters  $\mathbf{q}$ , Popov's sufficient condition for absolute stability can be applied to the *Popov value set*

$$G_p(j\omega, Q) := \{g_p(j\omega, \mathbf{q}) \mid \mathbf{q} \in Q\} = \{\operatorname{Re} g(j\omega, \mathbf{q}) + j\omega \operatorname{Im} g(j\omega, \mathbf{q}) \mid \mathbf{q} \in Q\} \quad (8.4.16)$$

For a fixed frequency  $\omega = \omega^*$  the Popov value set  $G_p(j\omega^*, Q)$  can be easily constructed from the Nyquist value set  $G(j\omega^*, Q)$  by multiplication of the imaginary part with  $\omega^*$ . Thus, if we have found a tree-structured decomposition for an uncertain plant transfer function  $g(s, \mathbf{q})$ , then both the Nyquist and Popov value sets can be constructed.

*Example 8.7.* Consider the crane (1.1.6) uncertain in the rope length  $\ell \in [8 ; 16]$  [m], the load mass  $m_L \in [50 ; 2000]$  [kg], and the crab mass  $m_C \in [800 ; 1200]$  [kg]. Using full state feedback  $\mathbf{u} = \mathbf{k}^T \mathbf{x} = k_1 x_1 + k_2 x_2 + k_3 x_3 + k_4 x_4$  it can be shown that the closed loop system is robustly stable for a controller coefficient vector  $\mathbf{k}$  with values arbitrarily chosen in the intervals  $k_1 \in [500 ; 700]$ ,  $k_2 \in [3000 ; 4000]$ ,  $k_3 \in [-30000 ; -25000]$ ,  $k_4 \in [-2800 ; -2400]$ . Now we assume that the unmodelled actuator has a nonlinear but unknown characteristic of the class (8.4.5). We want to determine the greatest sector for that nonlinear function for which the closed-loop system is absolutely stable. The linear part of the system is described by the transfer function ( $\mathbf{q} = [\ell \ m_L \ m_C \ k_1 \ k_2 \ k_3 \ k_4]^T$ )

$$g(s, \mathbf{q}) = \frac{(k_1 + k_2 s)(\ell s^2 + g) - (k_3 + k_4 s)s^2}{s^2[m_C(s^2\ell + g) + m_L g]} \quad (8.4.17)$$

Since  $g$  is unstable, Theorem 8.7 cannot directly be applied. Thus, the system first has to be transformed to the standard form of Fig. 8.18. From the above we know that the closed-loop linear part of the system is robustly stable for  $\rho = 1$ . That value is used for constructing the Popov value set of  $\tilde{g} = g / (1 + g) = 1/(1 + 1/g)$ . In the transfer function (8.4.17) all uncertain parameters but the rope length appear just once, i.e. the transfer function has a tree structure if the rope length  $\ell$  is gridded. For the following  $\ell$  is assumed fixed with  $\ell = 12$  [m]. The Popov value set can be constructed sequentially, the union of the sets is shown in Fig. 8.21. From that plot we get the Popov sector

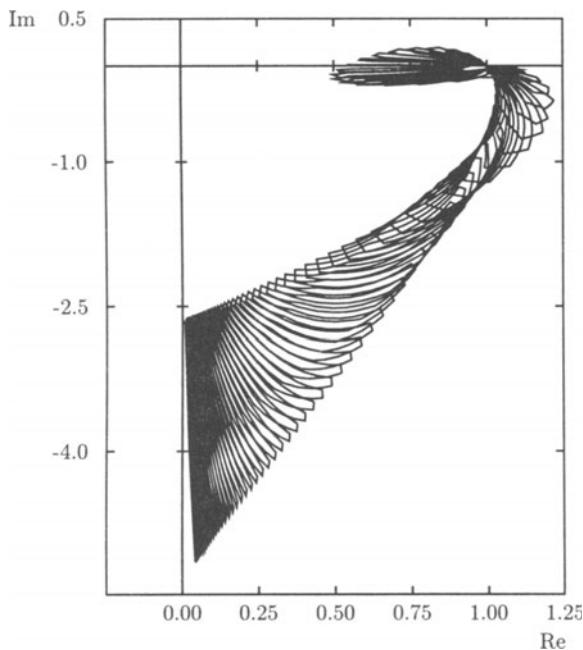


Fig. 8.21: Union of Popov value sets

$[0; \infty)$  for the nonlinear function  $\tilde{f}(.) = (f - \rho)(.).$  Thus, for the transfer function  $g$ , that means for the crane with the above state feedback, absolute stability is guaranteed in a  $k$ -sector  $[1; \infty).$  The problem of determining the Popov sector is formulated in the exercises of this chapter.  $\square$

## 8.5 Exercises

- 8.1. Consider the transfer function (8.1.29) for the crane

$$y_c/u = g_c(s, m_L) = (1 + s^2)/[(m_L + 1000)s^2 + 1000s^4]$$

Using the feedback control (8.1.30),

$$u = -\hat{g}(s) y_c = -[(\hat{n}_0 + \hat{n}_1 s)/(10 + s)] y_c$$

check if all controller coefficients in the rectangle

$$Q_{\hat{n}} := \{\hat{n}_0 \in [0.1; 0.1], \hat{n}_1 \in [1; 2]\} \quad (8.5.1)$$

robustly stabilize the crane for a load mass uncertain in the following interval

$$m_L \in [50; 3000] [\text{kg}] \quad (8.5.2)$$

Make the same investigation for

$$Q_{\hat{n}} := \{\hat{n}_0 \in [0.2; 0.2], \hat{n}_1 \in [1; 2]\}$$

Take any controller coefficients  $\hat{n}_0, \hat{n}_1$  that robustly stabilizes the crane for the load mass  $m_L$  in the above interval. Find the minimal load mass  $m_{Lmin}$  and the maximal load mass  $m_{Lmax}$  without loosing stability for  $m_L \in [m_{Lmin}; m_{Lmax}]$ .

8.2. Consider the robot arm control system of Fig. 2.13, let  $q_3 := 1/q_1$ .

a) Check the stability of the loop for  $q_2 \in [0.05; 0.2]$ ,  $q_3 \in [0.1; 1.0]$ .

b) Analyze absolute robust stability for an additional actuator nonlinearity in a gain sector  $[0.8; 1.2]$ . Does Aizerman's conjecture hold?

8.3. Check if the following transfer functions are strictly positive real:

$$\frac{1}{s + (q_1^2 + q_2^2)(q_3 + 2)}, \quad q_i \in [-1; 1], \quad i = 1, 2, 3$$

$$\frac{1}{s + (q_1^2 + q_2^2 - 0.01)(q_3 + 2)}, \quad q_i \in [-1; 1], \quad i = 1, 2, 3$$

$$\frac{1}{(s + q_1)(s + q_2)}, \quad q_i \in [1; 2], \quad i = 1, 2$$

$$\frac{s}{(s + q_1)(s + q_2)}, \quad q_i \in [1; 2], \quad i = 1, 2$$

$$\frac{s - 1}{(s + q_1)(s + q_2)}, \quad q_i \in [1; 2], \quad i = 1, 2$$

8.4. For the bus model introduced in Section 1.2 consider the transfer function from the steering angle  $\delta_f$  to the yaw rate  $r$ ,

$$\frac{r}{\delta_f} = \frac{b_0 + b_1 s}{a_0 + a_1 s + s^2}$$

with

$$b_0 = \frac{48031}{\tilde{m}^2 v}, \quad b_1 = \frac{66.9733}{\tilde{m}}$$

$$a_0 = \frac{16.6304}{\tilde{m}} + \frac{268973}{\tilde{m}^2 v^2}, \quad a_1 = \frac{1075.15}{\tilde{m} v}$$

For the uncertainty intervals

$$\tilde{m} \in [9950; 32000] [\text{kg}], \quad v \in [1; 20] [\text{m/s}]$$

evaluate  $x_i^+ = \max_{\tilde{m}, v}(x_i)$ ,  $x_i^- = \min_{\tilde{m}, v}(x_i)$ ,  $x = a, b$ ,  $i = 0, 1$  and apply Theorem 8.5 for the respective interval transfer function. Compare the result with Example 8.2.

- 8.5. For Example 8.7 produce the Popov plot for the fixed values

$$\ell = 12 [\text{m}], m_L = 1000 [\text{kg}], m_C = 1000 [\text{kg}]$$

$$k_1 = 600, k_2 = 3500, k_3 = -27500, k_4 = -2600$$

Determine the Popov sector. Repeat Example 8.7 with  $\rho = 0.01$ .

## 9 Gamma-Stability

Chapters 4 through 8 were exclusively concerned with robust Hurwitz stability, i.e. with the requirement that all roots of a polynomial family are located in the open left half plane. In Chapter 3 it was explained how unsatisfactory responses of stable systems can be improved by additional requirements on the root locations as specified by a region  $\Gamma$  in the complex  $s$ -plane. There also proposals were made for choosing  $\Gamma$ . For discrete-time systems  $\Gamma$  is the unit disk or a subset of it. In this chapter it will be shown which stability results can be generalized, must be modified or are no longer valid. The only approach that does not need a modification is the root set calculation. Other approaches are based on the Boundary Crossing Theorem 4.3. Obviously it remains valid for Gamma-stability. However, a representation of the boundary  $\partial\Gamma$  of  $\Gamma$  is needed.

### 9.1 Boundary Representation

First, the mathematical description of  $\Gamma$  has to be discussed. It is reasonable to describe  $\Gamma$  by the equation of the boundary  $\partial\Gamma$ . In the case of the left half plane the boundary is described by

$$\partial\Gamma := \{s \mid s = j\omega, 0 \leq \omega < \infty\} \quad (9.1.1)$$

The restriction to nonnegative  $\omega$  is made because polynomials with real coefficients are assumed. This assumption is further valid so the chosen root regions should always be symmetric with respect to the real axis. The boundary  $\partial\Gamma$  of the desired root region  $\Gamma$  can be described by

$$\partial\Gamma := \{s \mid s = \sigma(\alpha) + j\omega(\alpha), \alpha \in [\alpha^-; \alpha^+]\} \quad (9.1.2)$$

The lower and upper bounds  $\alpha^-$  and  $\alpha^+$  may also be  $+\infty$  or  $-\infty$ . The scalar parameter  $\alpha$  is called *generalized frequency* or shown  $\partial\Gamma$ -*parameter*. For generalizing the results of Chapters 4 to 8 to  $\Gamma$ -stability,  $s = j\omega$  has to be replaced by  $s = \sigma(\alpha) + j\omega(\alpha)$  and  $s = 0$  by  $s = a$  where  $a$  is an intersection point of  $\partial\Gamma$  with the real axis.

Assume that the (real) functions  $\sigma(\alpha)$  and  $\omega(\alpha)$  are polynomials or simple trigonometric functions. Choosing polynomials of degree less or equal two, root regions can

be described which are bounded by conic curves, i.e. straight lines, circles, ellipses, hyperbolas and parabolas. For most applications it suffices to compose the boundary  $\partial\Gamma$  from segments of conic curves. Examples have been given in Figs. 3.6, 3.7 and 3.9. The parametrization by the parameter  $\alpha$  is illustrated by the following examples.

*Example 9.1.* Lines of constant damping  $D$ , see Fig. 3.2. With  $\omega^2 = \omega_0^2(1 - D^2) = \sigma^2(1 - D^2)/D^2$ , a feasible parametrization is  $\alpha = \sigma$ , i.e.

$$s = \sigma + j\sigma\sqrt{1 - D^2}/D, \quad \sigma \leq 0 \quad (9.1.3)$$

e.g. for  $D = 1/\sqrt{2}$ ,  $s = \sigma + j\sigma$ . A real root boundary occurs at  $s = 0$ .  $\square$

*Example 9.2.* The roots must be located on the negative real axis in the interval  $[-b; -a]$ ,  $a, b > 0$ . Then

$$s = \sigma, \quad \sigma \in [-b; -a] \quad (9.1.4)$$

The segment generates two real root boundaries for  $\sigma = -b$  and  $\sigma = -a$  and a complex root boundary for a double root in the interval  $[-b; -a]$ . If the polynomial has only a single root in the above interval, then this root can leave the segment only at  $\sigma = -b$  or  $\sigma = -a$  and there exists no complex root boundary. If there are more than one root in the interval then a pair of roots can leave the segment also at an intermediate value of  $\sigma$ , see also (9.3.5).  $\square$

*Example 9.3.* Left branch of the hyperbola  $(\frac{\sigma}{a})^2 - (\frac{\omega}{b})^2 = 1$

The hyperbola is shown in Fig. 9.1.

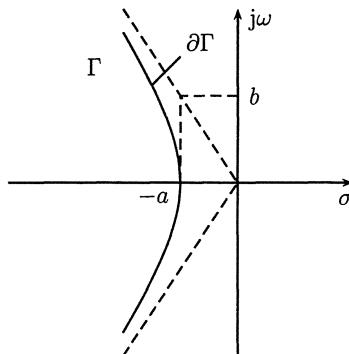


Fig. 9.1: A hyperbola combines constraints on damping and real part of the eigenvalues

This hyperbola ensures a minimum damping of  $D = \frac{a}{\sqrt{a^2+b^2}}$  corresponding to the asymptotes and a maximal real part of the eigenvalues of  $\sigma = -a$ . Let  $\alpha = \sigma$ , then  $\omega^2 = b^2(\sigma^2/a^2 - 1)$  and

$$s = \sigma + jb\sqrt{\sigma^2/a^2 - 1}, \quad \sigma \leq -a \quad (9.1.5)$$

A real root boundary occurs at  $\sigma = -a$ .  $\square$

*Example 9.4.* Circle with center  $s = 0$  and radius  $R$ :  $\sigma^2 + \omega^2 = R^2$ . Let  $\alpha = \sigma$ ,  $\omega^2 = R^2 - \sigma^2$ , then

$$s = \sigma + j\sqrt{R^2 - \sigma^2}, \quad \sigma \in [-R; R] \quad (9.1.6)$$

□

*Example 9.5.*  $\Gamma$ -stability region of Fig. 3.6. It can be composed from the two previous examples. The circle and the hyperbola intersect for a real value  $\sigma_1 = -\sqrt{\frac{\sigma^2(R^2+b^2)}{a^2+b^2}}$ . Therefore,

$$s = \begin{cases} \sigma + j\sqrt{R^2 - \sigma^2} & \text{for } \sigma \in [-R; \sigma_1] \\ \sigma + jb\sqrt{\sigma^2/a^2 - 1} & \text{for } \sigma \in [\sigma_1; -a] \end{cases}$$

There are two real root boundaries at  $\sigma = -R$  and  $\sigma = -a$ . □

*Example 9.6.* The boundary segments can be described also by different parameters  $\alpha, \beta, \gamma, \dots$ . Take for example the “pineapple segment” of Fig. 9.2.

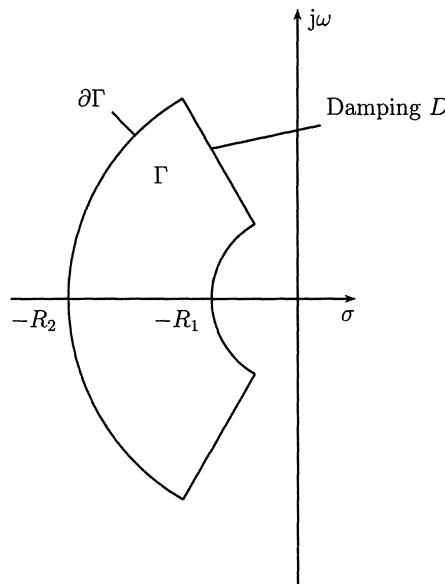


Fig. 9.2: Pineapple segment. This type of  $\Gamma$ -stability is required in flight control systems

The boundary is composed from those of Example 9.1 and Example 9.4 as

$$s = \begin{cases} \alpha + j\sqrt{R_2^2 - \alpha^2} & \text{for } \alpha \in [-R_2; -R_2D] \\ \beta + j\beta\sqrt{1 - D^2}/D & \text{for } \beta \in [-R_2D; -R_1D] \\ \gamma + j\sqrt{R_1^2 - \gamma^2} & \text{for } \gamma \in [-R_1D; -R_1] \end{cases}$$

The  $\partial\Gamma$ -parameters  $\alpha$ ,  $\beta$  and  $\gamma$  are all identified with the real part  $\sigma$ . However for a fixed  $\sigma$ , two or three boundary points with  $\omega > 0$  are obtained. The real root boundaries are  $\sigma = -R_2$  and  $\sigma = -R_1$ .  $\square$

In the last two examples the  $\Gamma$ -region was defined as an intersection of several regions. Also a union of regions can be useful. If the union consists of disjoint sets, then it must also be specified, how many roots are located in each disjoint set. A practical use of a union of sets is the following. Assume you have designed a controller for a nominal plant model, i.e. you know the nominal location of all closed-loop eigenvalues. If the uncertain parameters vary, you want to keep the eigenvalues in some well-defined neighborhood of their nominal locations.

*Example 9.7.* Assume that the nominal polynomial in factorized form is

$$p(s) = (4 + 2s + s^2)(2 + s)(5 + s)(8 + s)(72 + 12s + s^2)$$

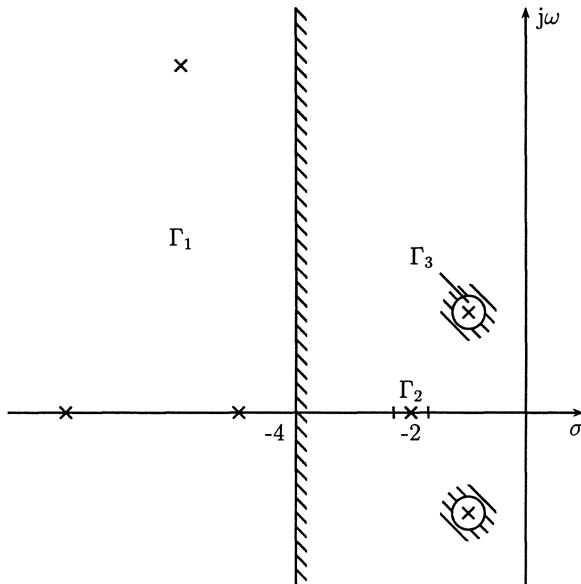


Fig. 9.3: Several components may constitute a  $\Gamma$ -stability region

$\square$

The roots are shown in Fig. 9.3, they represent a dominant Butterworth pole configuration in a distance two from the origin and some eigenvalues that are located further to

the left.  $\Gamma$ -stability is now defined by the admissible migration of individual eigenvalues or group of eigenvalues. Take for example a circle of radius 0.3 around the eigenvalues at  $s_{1,2} = -1 + j\sqrt{3}$ . Then it is important to keep also the real eigenvalue at  $s = -2$  in the interval  $\sigma \in [-2.3; -1.7]$ , otherwise the low damping  $D = 0.5$  of the two eigenvalues  $s_{1,2}$  yields too much overshoot of the step response, see Fig. 3.5. It is not necessary to care about the other eigenvalues as long as they do not interfere with the dominant behavior. Therefore, the requirement is that their real part must remain smaller than  $\sigma_1 = -4$ . The disjoint parts  $\partial\Gamma_1$ ,  $\partial\Gamma_2$  and  $\partial\Gamma_3$  of the boundary are indicated as in Fig. 9.3. Note that the real axis interval contains only one real root, therefore,  $\partial\Gamma_2$  consists only of the real root boundaries at  $\sigma_2 = -2.3$  and  $\sigma_3 = -1.7$ . The left region  $\Gamma_1$  has a real root boundary at  $\sigma_1 = -4$  and its complex root boundary is  $s = -4 + j\omega$ ,  $\omega \geq 0$ . The circle around the dominant complex poles is described by

$$\partial\Gamma_3 = \{s \mid s = -1 + 0.3 \cos \alpha + j(\sqrt{3} + 0.3 \sin \alpha), \alpha \in [0; 2\pi]\} \quad (9.1.7)$$

## 9.2 Boundary Crossing

The Boundary Crossing Theorem of Frazer and Duncan (Theorem 4.3) can be immediately generalized to other pole regions. Only minor modifications have to be made.

*Theorem 9.1. (Boundary Crossing Theorem for pole regions  $\Gamma$ )*

Given a set of polynomials  $P(s, Q)$  as introduced in (4.0.1.-4.0.7). The set  $P(s, Q)$  is robustly  $\Gamma$ -stable if and only if

- 1) there exists a  $\Gamma$ -stable polynomial  $p(s) \in P(s, Q)$ ,
- 2)  $\sigma(\alpha) + j\omega(\alpha) \notin \text{Roots}[P(s, Q)]$  for all  $\alpha \in [\alpha^-; \alpha^+]$ .

□

If  $\Gamma$  consists of an intersection or union of regions with boundaries  $\partial\Gamma_i$ , then the second conditions refers to all active branches of  $\partial\Gamma_i$ , i.e. all boundary segments that shall not be crossed by a root.

## 9.3 Algebraic Problem Formulation

Corresponding to the necessary and sufficient conditions of Hurwitz it is possible to derive sets of inequalities see [158]. Obviously the number of inequalities depends on

the complexity of  $\Gamma$  resp.  $\partial\Gamma$  (for example on the order of the bounding curves). A restriction to the critical  $\Gamma$ -stability constraints is sufficient because a  $\Gamma$ -stable  $\mathbf{q}^0$  is known and the boundary crossing occurs if these constraints are no longer satisfied.

For Hurwitz stability the three hypersurfaces

$$\operatorname{Re} p(j\omega, \mathbf{q}) = 0, \quad \operatorname{Im} p(j\omega, \mathbf{q}) = 0 \quad (9.3.1)$$

$$a_0(\mathbf{q}) = 0 \quad (9.3.2)$$

$$a_n(\mathbf{q}) = 0 \quad (9.3.3)$$

represent the boundary surfaces around a nominal point  $\mathbf{q}^0$  which corresponds to a stable polynomial. Eliminating  $\omega$  from the equation (9.3.1) gives the last but one Hurwitz determinant  $\Delta_{n-1}$ , the complex-root boundary, see Theorem 4.5 and (4.3.3).

The extension to other pole regions  $\Gamma$  is straightforward.  $s = j\omega$  in (9.3.1) is replaced by  $s = \sigma(\alpha) + j\omega(\alpha)$ . Compared to Hurwitz stability there are only minor differences. The resulting equations are more complicated depending on the complexity of the functions  $\sigma = \sigma(\alpha)$ ,  $\omega = \omega(\alpha)$  which describe the boundary of  $\Gamma$ . Eliminating  $\alpha$  by the resultant method gives the analogon to the last but one Hurwitz determinant that is

$$\operatorname{Res}_\alpha (\operatorname{Re} p[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}], \operatorname{Im} p[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}]) = 0 \quad (9.3.4)$$

This elimination has to be done for each component of the boundary  $\partial\Gamma$ . A special case is the aperiodicity constraint that admits only roots on the negative real axis, see Example 9.2. A polynomial  $p(s)$  has a double root at  $s = s^*$  if and only if  $p(s^*) = 0$  and  $p'(s^*) = 0$ , where  $p'(s) = dp(s)/ds$ . Therefore, (9.3.1) has to be replaced by

$$p(\alpha, \mathbf{q}) = 0, \quad p'(\alpha, \mathbf{q}) = 0 \quad (9.3.5)$$

The resultant with respect to  $p(s)$  and  $p'(s)$  is called the *discriminant* of  $p(s)$ , see Appendix B.

## 9.4 Gamma-Stability Boundaries in Parameter Space

An alternative to the algebraic formulation is the parameter space approach. Here the boundary  $\partial\Gamma$  has to be mapped into the  $\mathbf{q}$ -space to obtain  $Q_{\Gamma\text{-stable}}$ . In robustness analysis of a given operating domain  $Q$  it must be checked if  $Q \subset Q_{\Gamma\text{-stable}}$ . A very useful application of the parameter space approach is also in design, when the elements of the  $\mathbf{q}$ -vector are free controller parameters  $k_i$ . Then the set of all  $\Gamma$ -stabilizing controllers of the assumed structure can be described. This point will be explored in Chapter 11. The following theorem [15] gives a systematic approach to the problem of mapping  $s = \sigma(\alpha) + j\omega(\alpha)$  to the  $\mathbf{q}$ -space.

*Theorem 9.2. (Boundary Representation Theorem, Ackermann, Kaesbauer)*

Consider a polynomial family  $p(s, \mathbf{q}) = [1 s \dots s^n] \mathbf{a}(\mathbf{q})$  and

$$Q_{Im}(\alpha) := \{ \mathbf{q} \mid p(\sigma(\alpha) + j\omega(\alpha), \mathbf{q}) = 0, \alpha \in [\alpha^-; \alpha^+] \} \quad (9.4.1)$$

(That is the set of parameter vectors  $\mathbf{q}$ , which yield a polynomial with a root pair at  $s = \sigma(\alpha) \pm j\omega(\alpha)$ ). Now  $\mathbf{q} \in Q_{Im}(\alpha)$  if and only if

$$\begin{bmatrix} d_0(\alpha) & d_1(\alpha) & \dots & d_n(\alpha) \\ 0 & d_0(\alpha) & \dots & d_{n-1}(\alpha) \end{bmatrix} \mathbf{a}(\mathbf{q}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (9.4.2)$$

for some  $\alpha \in [\alpha^-; \alpha^+]$ , where

$$\begin{aligned} d_0(\alpha) &= 1 \\ d_1(\alpha) &= 2\sigma(\alpha) \\ d_{i+1}(\alpha) &= 2\sigma(\alpha)d_i(\alpha) - [\sigma^2(\alpha) + \omega^2(\alpha)]d_{i-1}(\alpha), \quad i = 1, 2, \dots, n-1 \end{aligned} \quad (9.4.3)$$

□

*Proof.*

Consider the polynomial  $p(s, \mathbf{q})$  for fixed  $\mathbf{q} = \mathbf{q}^*$ . It has a complex conjugate pair of roots on  $\partial\Gamma$  at  $\sigma(\alpha) \pm j\omega(\alpha)$  if and only if

$$p(s, \mathbf{q}^*) = [\sigma^2(\alpha) + \omega^2(\alpha) - 2\sigma(\alpha)s + s^2] r(s, \mathbf{q}^*) \quad (9.4.4)$$

where

$$r(s, \mathbf{q}^*) = r_0 + r_1 s + r_{n-2} s^{n-2} = [1 s \dots s^{n-2}] \mathbf{r} \quad (9.4.5)$$

is an arbitrary polynomial of degree  $n-2$  with real coefficients. Equivalently (omitting the dependency of  $\sigma$  and  $\omega$  on  $\alpha$  for notational convenience)

$$\mathbf{a}(\mathbf{q}^*) = \begin{bmatrix} 1 & -2\sigma & \sigma^2 + \omega^2 & 0 & \dots & 0 \\ 0 & 1 & -2\sigma & . & & \\ 0 & 0 & 1 & . & . & 0 \\ & & & . & . & \sigma^2 + \omega^2 \\ & & & . & . & -2\sigma \\ 0 & & & & & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ r_0 \\ . \\ . \\ r_{n-2} \end{bmatrix} \quad (9.4.6)$$

The matrix in (9.4.6) is triangular with identical elements on the diagonals. Therefore, its inverse  $\mathbf{D}$  has the same structure. The entries  $d_i$  of  $\mathbf{D}$  are determined from

$$\begin{bmatrix} d_0 & d_1 & & d_n \\ 0 & d_0 & . & . \\ . & . & d_1 & . \\ . & . & d_0 & \end{bmatrix} \begin{bmatrix} 1 & -2\sigma & \sigma^2 + \omega^2 & 0 & \dots & 0 \\ 0 & 1 & -2\sigma & . & & \\ 0 & 0 & 1 & . & . & 0 \\ & & & . & . & \sigma^2 + \omega^2 \\ & & & 0 & & -2\sigma \\ & & & & & 1 \end{bmatrix} = \mathbf{I} \quad (9.4.7)$$

which forces

$$\begin{aligned} d_0 &= 1 \\ -2\sigma d_0 + d_1 &= 0 \\ (\sigma^2 + \omega^2)d_0 - 2\sigma d_1 + d_2 &= 0 \\ &\vdots \\ (\sigma^2 + \omega^2)d_{n-2} - 2\sigma d_{n-1} + d_n &= 0 \end{aligned}$$

and solving for the  $d_i$  yields

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 2\sigma \\ d_2 &= 2\sigma d_1 - (\sigma^2 + \omega^2)d_0 \\ &\vdots \\ d_n &= 2\sigma d_{n-1} - (\sigma^2 + \omega^2)d_{n-2} \end{aligned}$$

Premultiplying (9.4.6) by  $\mathbf{D}$ , it can be concluded that  $p(s(\alpha), \mathbf{q}^*) = 0$  if and only if

$$\left[ \begin{array}{cccc} d_0 & d_1 & \dots & d_n \\ 0 & d_0 & & \\ & \ddots & & \\ & & \ddots & d_1 \\ 0 & & 0 & d_0 \end{array} \right] \mathbf{a}(\mathbf{q}^*) = \left[ \begin{array}{c} 0 \\ 0 \\ \vdots \\ r_0 \\ \vdots \\ r_{n-2} \end{array} \right] \quad (9.4.8)$$

The last  $n - 1$  rows of (9.4.8) have an undetermined right hand side because the remainder polynomial  $r(s, \mathbf{q}^*)$  is arbitrary. In other words: whatever the left-hand side requires, the equation can be satisfied by the appropriate choice of the  $r_i$ . There only remain the first two rows. Hence, it can be concluded that  $\mathbf{q}^* \in Q_{Im}(\alpha)$  if and only if

$$\left[ \begin{array}{cccc} d_0(\alpha) & d_1(\alpha) & \dots & d_n(\alpha) \\ 0 & d_0(\alpha) & \dots & d_{n-1}(\alpha) \end{array} \right] \mathbf{a}(\mathbf{q}^*) = \left[ \begin{array}{c} 0 \\ 0 \end{array} \right] \quad (9.4.9)$$

The fixed value  $\mathbf{q}^*$  may be replaced by the general vector  $\mathbf{q}$  to obtain (9.4.2). Thus, the proof of the theorem is complete.

□

Clearly, the polynomial  $p(s, \mathbf{q})$  has a complex conjugate pair of roots on  $\partial\Gamma$  if and only if there exists an  $\alpha \in [\alpha^-; \alpha^+]$  such that (9.4.2) holds. In other words, in analogy to the  $\omega$ -sweep over the imaginary axis in the robust stability test of Chapter 4, an  $\alpha$ -sweep along all branches or segments of  $\partial\Gamma$  must be made. For each  $\alpha$  the set  $Q_{Im}(\alpha)$  can be calculated from (9.4.2) with the exception of generalized isolated frequencies  $\alpha_s$ . The values of  $\alpha_s$  are determined as in Definition 4.1 with  $j\omega$  replaced by  $\sigma(\alpha) + j\omega(\alpha)$ .

*Example 9.8.* Let  $\partial\Gamma$  be the imaginary axis, i.e.  $\sigma = 0$ ,  $\omega^2(\alpha) = \alpha$ ,  $\alpha \in [0; \infty)$

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 0 \\ d_{i+1} &= -\alpha d_{i-1} \\ \left[ \begin{array}{cccccc} 1 & 0 & -\alpha & 0 & \alpha^2 & \dots \\ 0 & 1 & 0 & -\alpha & 0 & \dots \end{array} \right] \mathbf{a}(\mathbf{q}) &= \left[ \begin{array}{c} 0 \\ 0 \end{array} \right] \end{aligned} \quad (9.4.10)$$

This representation was used in (4.5.1). The real root boundaries are  $a_0(\mathbf{q}) = 0$  and  $a_n(\mathbf{q}) = 0$ .  $\square$

*Example 9.9.* Consider the hyperbola of Fig. 9.1 with  $s = \sigma + jb\sqrt{\sigma^2/a^2 - 1}$ . The recursion formula for the  $d_i(\sigma)$  yields

$$\begin{aligned} d_0(\sigma) &= 1 \\ d_1(\sigma) &= 2\sigma \\ d_{i+1}(\sigma) &= 2\sigma d_i(\sigma) - [\sigma^2(1 + b^2/a^2) - b^2] d_{i-1}(\sigma) \end{aligned}$$

The real root boundary is  $p(-a, \mathbf{q}) = 0$ .  $\square$

*Example 9.10.* Consider the circle of Example 9.4 with  $R = 1$  and  $s = \sigma + j\sqrt{1 - \sigma^2}$ ,  $\sigma \in [-1; 1]$ . The  $d_i$  in (9.4.3) are

$$\begin{aligned} d_0(\sigma) &= 1 \\ d_1(\sigma) &= 2\sigma \\ d_{i+1}(\sigma) &= 2\sigma d_i(\sigma) - d_{i-1}(\sigma) \end{aligned}$$

The resulting equation can be further simplified as will be shown in Chapter 10. The real root boundaries are  $p(-1, \mathbf{q}) = 0$  and  $p(1, \mathbf{q}) = 0$ .  $\square$

*Example 9.11.* Let  $\partial\Gamma$  be the logarithmic spiral  $z = e^{c\alpha}e^{j\alpha}$ ,  $\alpha \in [-\pi; 0]$ . This curve is of interest for sampled-data-systems. Constant damping lines  $\sigma = \pm c\omega$  ( $c = -D/\sqrt{1 - D^2}$ ,  $D$  = damping) are mapped via  $z = e^{sT} = e^{\sigma T}e^{j\omega T}$  into the  $z$ -plane (see also Section 10.3). Let  $\alpha = \omega T$ ,  $\alpha \in [-\pi; 0]$ ,  $z = \tau + j\eta = e^{c\alpha} \cos \alpha + j e^{c\alpha} \sin \alpha$ . The real and imaginary parts  $\tau$  and  $\eta$  now play the role of  $\sigma$  and  $\omega$  in (9.1.2).

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 2\tau = 2e^{c\alpha} \cos \alpha \end{aligned} \quad (9.4.11)$$

$$\begin{aligned} d_{i+1} &= 2\tau d_i - (\tau^2 + \omega^2) d_{i-1} \\ &= 2e^{c\alpha} \cos \alpha d_i - e^{2c\alpha} d_{i-1} \end{aligned} \quad (9.4.12)$$

This is an example where the algebraic (Hurwitz-like) approach fails. The boundary mapping, however is not particularly difficult. The real root boundaries are  $p(1, \mathbf{q}) = 0$  and  $p(-e^{c\alpha}, \mathbf{q}) = 0$ .  $\square$

*Remark 9.1.* A related boundary representation was derived by Šiljak [154]. Instead of the starting point (9.4.3) he uses the real and imaginary part of  $p[\sigma(\alpha) + j\omega(\alpha)]$ . The  $k$ -th power of  $s$  is expressed by  $[s(\alpha)]^k = [\sigma(\alpha) + j\omega(\alpha)]^k = X_k(\alpha) + jY_k(\alpha)$ . Then

$$\begin{aligned} p[\sigma(\alpha) + j\omega(\alpha)] &= \operatorname{Re} p(X_1(\alpha) + jY_1(\alpha)) + j \operatorname{Im} p(X_1(\alpha) + jY_1(\alpha)) \\ &= \sum_{k=0}^n a_k X_k(\alpha) + j \sum_{k=0}^n b_k Y_k(\alpha) = 0 \end{aligned} \quad (9.4.13)$$

The two equations for real and imaginary parts are related to (9.4.2) by

$$\begin{bmatrix} \operatorname{Re} p[\sigma(\alpha) + j\omega(\alpha)] \\ \operatorname{Im} p[\sigma(\alpha) + j\omega(\alpha)] \end{bmatrix} = \begin{bmatrix} 1 & -\sigma(\alpha) \\ 0 & \omega(\alpha) \end{bmatrix} \begin{bmatrix} d_0(\alpha) & d_1(\alpha) & \dots & d_n(\alpha) \\ 0 & d_0(\alpha) & \dots & d_{n-1}(\alpha) \end{bmatrix} \mathbf{a}(\mathbf{q}) \quad (9.4.14)$$

The advantage of the form (9.4.2) is that only one recursion for the  $d_k$  is needed instead of two recursions for  $X_k$  and  $Y_k$ .  $\square$

#### Parameter space graphics

The boundary mapping equation (9.4.2) is well suited for generating computer graphics of  $\Gamma$ -stability regions in  $\mathbf{q}$ -space. 3D-graphics have been used by Putz and Wozny [142]. Beautiful renderings of  $Q_{\Gamma\text{-stable}}$  can be rotated and zoomed. Also light sources for illuminating the object can be moved and placed favorably. Additional information can be coded by colors for the surface of  $\Gamma$ -stability regions. Such computer graphics environment definitively helps to understand the geometry of  $\Gamma$ -stability regions in a space of three uncertain parameters.

A critic may ask: What do you do if you have four parameters? Certainly, it becomes tedious to repeat the 3D-visualization for several grid points of  $q_4$ . An animation with  $q_4$  as time may help. Ideal would be an online computation in which we can interactively move back and forth in the  $q_4$ -direction by a dial and see the 3D-object change its shape. Here we quickly come to the limits of computing power, and our critic will certainly ask: What do you do if you have more parameters?

Let us ask back to the critic: How would you do it? Most likely he would suggest a scalar measure for  $Q_{\Gamma\text{-stable}}$ , for example a  $\Gamma$ -stable nominal point and a norm defining the sidelength proportions of a box centered at the nominal point. This is clearly a reasonable approach if we are only interested to know if a given  $Q$ -box is contained in  $Q_{\Gamma\text{-stable}}$ . We will come back to this topic in the section on the  $\Gamma$ -stability radius.

In Chapter 11 we will design  $\Gamma$ -stable control systems by simultaneous pole region assignment as introduced with Fig. 3.14. There we map  $\Gamma$ -stability boundaries into a  $\mathbf{k}$ -space of controller gains and look for intersections of admissible controller sets for different operating conditions. Intersections are best recognized in 2D-graphics. These graphics can be calculated so quickly that the interactive dialing of additional parameters, as mentioned above for 3D-graphics, become feasible. Also in our present context of robustness analysis there are nontrivial problems which have only two parameters, for example velocity  $v$  and virtual mass  $\tilde{m}$  ( $= m/\mu$ ) of a car. In such cases a robustness analysis in a parameter plane is very informative.

For a given uncertain polynomial

$$p(s, \mathbf{q}) = [1 \ s \ s^2 \ \dots \ s^n] \ \mathbf{a}(\mathbf{q}) \quad (9.4.15)$$

the real root boundaries and complex root boundaries in the  $(q_1, q_2)$ -plane are now constructed. If there are further uncertain parameters  $q_3, q_4, \dots, q_\ell$  then they must be gridded. The real root boundary is determined by the intersection points  $s_i$  of  $\partial\Gamma$  with the real axis, that is

$$p(s_i, \mathbf{q}) = [1 \ s_i \ s_i^2 \ \dots \ s_i^n] \ \mathbf{a}(\mathbf{q}) = 0 \quad (9.4.16)$$

In case of affine dependency of  $\mathbf{a}(\mathbf{q})$  this equation represents a straight line in the parameter plane. In case of polynomial dependency, (9.4.16) represents a curve.

The complex root boundary is generated by sweeping the  $\partial\Gamma$ -parameter  $\alpha$  and solving (9.4.2), a system of two equations in  $q_1$  and  $q_2$ . If the coefficients  $a_i$  depend affinely on  $q_1$  and  $q_2$  then this system is linear and the equations can be solved explicitly for  $q_1$  and  $q_2$  and lead to a parametric representation of the complex root boundary with the parameter  $\alpha$ .

A more complicated case occurs if the coefficients  $\mathbf{a}(\mathbf{q})$  depend polynomially on the uncertain parameters. Then (9.4.2) may be written as

$$f = f_0(q_2) + f_1(q_2)q_1 + \dots + f_k(q_2)q_1^k = 0 \quad (9.4.17)$$

$$g = g_0(q_2) + g_1(q_2)q_1 + \dots + g_m(q_2)q_1^m = 0 \quad (9.4.18)$$

The  $f_i$  and  $g_i$  are polynomials in  $q_2$  and continuous functions of  $\alpha$ . (9.4.17) and (9.4.18) describes the intersection of two algebraic curves in the  $(q_1, q_2)$ -plane. These solution pairs  $(q_1, q_2)$  can be calculated either with the resultant method or by construction of a Gröbner basis, see Appendix B.

*Example 9.12.* Consider the controlled track guided bus with the characteristic polynomial as given in Example 7.5. The closed-loop poles should be located to the left of the left branch of the hyperbola

$$\left(\frac{\sigma}{0.35}\right)^2 - \left(\frac{\omega}{1.75}\right)^2 = 1 \quad (9.4.19)$$

The real root boundary for  $\sigma = -0.35$  is described by

$$\begin{aligned} p(-0.35, q_1, q_2) &= \\ -79.66(q_1^2 q_2^2 - 5339q_1^2 q_2 - 3077q_1 q_2 - 3540213q_1^2 + 9946676q_1 + 22088293) &= 0 \end{aligned} \quad (9.4.20)$$

This curve can be plotted by gridding say  $q_2$  and solving the remaining quadratic equation.

The complex root boundary  $\partial\Gamma$  is parameterized by  $\sigma(\alpha) = \alpha$ ,  $\omega^2(\alpha) = 25\alpha^2 - 1.75^2$ ,  $\alpha \in (-\infty; -0.35]$ . The coefficients  $d_i$  in the boundary representation (9.4.2) are given by

$$d_0(\alpha) = 1 \quad (9.4.21)$$

$$d_1(\alpha) = 2\alpha \quad (9.4.22)$$

$$d_{i+1}(\alpha) = 2\alpha d_i(\alpha) - (26\alpha^2 - 3.0625) d_{i-1}(\alpha) \quad (9.4.23)$$

Substituting the  $d_i$  and  $\mathbf{a}(\mathbf{q})$  into (9.4.2) and collecting terms with the same power of  $q_1$ , the following form is obtained:

$$f_0(\alpha) + [f_{10}(\alpha) + f_{11}(\alpha)q_2]q_1 + [f_{20} + f_{21}(\alpha)q_2 + f_{22}(\alpha)q_2^2]q_1^2 = 0 \quad (9.4.24)$$

$$g_0(\alpha) + [g_{10}(\alpha) + g_{11}(\alpha)q_2]q_1 + [g_{20} + g_{21}(\alpha)q_2 + g_{22}(\alpha)q_2^2]q_1^2 = 0 \quad (9.4.25)$$

The resultant has the form

$$\text{Res}_{q_1}(\alpha, q_2) = h_0(\alpha) + h_1(\alpha)q_2 + h_2(\alpha)q_2^2 + h_3(\alpha)q_2^3 + h_4(\alpha)q_2^4 = 0 \quad (9.4.26)$$

If all its roots are complex for a given  $\alpha = \alpha^*$ , then there exists no real pair  $(q_1, q_2)$  for which the closed loop has an eigenvalue at  $\sigma(\alpha^*) + j\omega(\alpha^*)$ . On the other hand if there are real solutions  $q_2^{(i)}(\alpha^*)$ , then the corresponding  $q_1^{(i)}(\alpha^*)$  is given by the roots of the greatest common divisor of (9.4.36) and (9.4.37). In this example the  $q_1^{(i)}(\alpha^*)$  corresponding to  $q_2^{(i)}(\alpha^*)$  can be expressed immediately by the coefficients  $f_i$  and  $g_i$  (see Appendix B):

$$q_1 = - \begin{vmatrix} f_0 & f_1 \\ g_0 & g_1 \end{vmatrix} : \begin{vmatrix} f_0 & f_2 \\ g_0 & g_2 \end{vmatrix} = - \begin{vmatrix} f_0 & f_{10} + f_{11}q_2 \\ g_0 & g_{10} + g_{11}q_2 \end{vmatrix} : \begin{vmatrix} f_0 & f_{20} + f_{21}q_2 + f_{22}q_2^2 \\ g_0 & g_{20} + g_{21}q_2 + g_{22}q_2^2 \end{vmatrix} \quad (9.4.27)$$

It turns out that the polynomial (9.4.26) has two real roots so that the complex root boundary consists of two branches. In Fig. 9.4 the root boundaries are plotted in the interesting domain of the parameters velocity ( $= q_1$ ) and virtual mass ( $= q_2$ ). The real root boundary is the dashed line, the complex root boundary is the solid line. Assuming the operating domain as given by  $v \in [3; 20][\text{ms}^{-1}]$  and  $\tilde{m} \in [9.9; 32][10^3 \text{ kg}]$  then the track guided bus is robustly  $\Gamma$ -stable.

For low speed  $v$ , the real root boundary at  $\sigma = -0.35$  is critical. At high speed there are two critical complex eigenvalue pairs, one at minimum virtual mass and  $\alpha = -2.97$  ( $s = -2.97 \pm j14.75$ ), the other one at maximum virtual mass and  $\alpha = -0.69$  ( $s = -0.69 \pm j2.97$ ).

□

## 9.5 Value Sets for Gamma-Stability

The zero exclusion approach to stability analysis for a polynomial family  $P(s, Q)$  is based on the conditions

1) there exists a stable polynomial  $p(s, \mathbf{q}) \in P(s, Q)$ ,

2)  $0 \notin \mathcal{P}(j\omega, Q)$  for all  $\omega \geq 0$

The extension to  $\Gamma$ -stability is obvious.

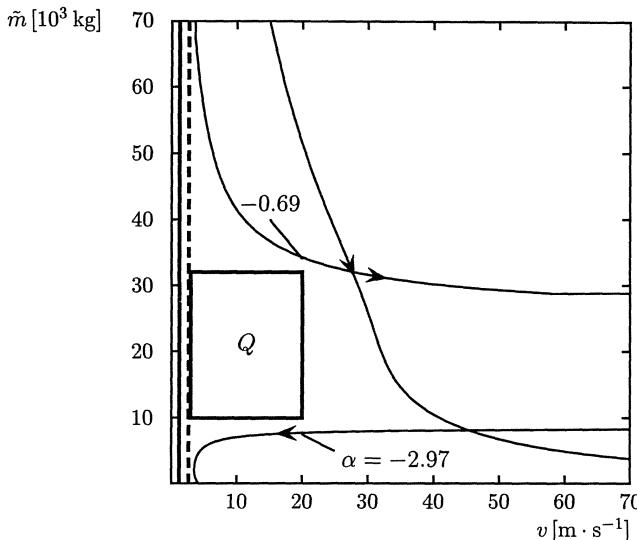


Fig. 9.4: The track guided bus is robustly  $\Gamma$ -stable in the operating domain  $Q$

*Theorem 9.3.*

Given a polynomial family  $P(s, Q)$  as in (4.0.1) - (4.0.7). The set  $P(s, Q)$  is robustly  $\Gamma$ -stable if and only if

1) there exists a  $\Gamma$ -stable polynomial  $p(s, \mathbf{q}) \in P(s, Q)$ ,

2)  $0 \notin \mathcal{P}(\sigma(\alpha) + j\omega(\alpha), Q)$  for all  $\alpha \in [\alpha^-; \alpha^+]$

If  $\partial\Gamma$  is composed of several boundary segments then all of them must be included in checking the second condition.

□

For  $\omega(\alpha) \neq 0$  the inverse of the relation (9.4.14) may be used. Instead of testing  $\operatorname{Re} p[\sigma(\alpha) + j\omega(\alpha)] + j\operatorname{Im} p[\sigma(\alpha) + j\omega(\alpha)] \neq 0$ , equivalently

$$c(\alpha, \mathbf{q}) = \operatorname{Re} p[\sigma(\alpha) + j\omega(\alpha)] + \frac{\sigma(\alpha)}{\omega(\alpha)} \operatorname{Im} p[\sigma(\alpha) + j\omega(\alpha)] + j \frac{1}{\omega(\alpha)} \operatorname{Im} p[\sigma(\alpha) + j\omega(\alpha)] \neq 0$$

may be used and with (9.4.14)

$$c(\alpha, \mathbf{q}) = [1 \ j] \begin{bmatrix} d_0(\alpha) & d_1(\alpha) & \dots & d_n(\alpha) \\ 0 & d_0(\alpha) & \dots & d_{n-1}(\alpha) \end{bmatrix} \mathbf{a}(\mathbf{q}) \neq 0$$

If however, a tree-structured decomposition of the polynomial

$$p(s, \mathbf{q}) = [1 \ s \ \dots \ s^n] \mathbf{a}(\mathbf{q}) = \operatorname{TSD}(p_1(s, \mathbf{q}^{I_1}), \dots, p_m(s, \mathbf{q}^{I_m}))$$

is possible, it should not be destroyed by forming  $\mathbf{a}(\mathbf{q})$ . The value set  $\mathcal{P}[\sigma(\alpha) + j\omega(\alpha), Q]$  can be directly generated from

$$\text{TSD } (p_1[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}^{I_1}], \dots, p_m[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}^{I_m}])$$

i.e. for a fixed  $\alpha^*$  the complex number  $s^* = \sigma(\alpha^*) + j\omega(\alpha^*)$  is substituted for  $s$ .

For a small number of uncertain parameters, i.e.  $\ell \leq 3$ , the value sets can be constructed using the Jacobian introduced in Chapter 5, even if the parameter dependency is nonlinear.

*Example 9.13.* Consider the track guided bus O 305 with the data given in Table 1.3 and the control structure shown in Fig. 2.4. In [131] the controller

$$f_C(s) = 25^3 \frac{0.6 + 0.7s + 0.15s^2}{(25 + s)(25^2 + 25s + s^2)}$$

was proposed. This controller guarantees  $\Gamma$ -stability for the entire operating domain  $v \in [3; 20] [\text{m} \cdot \text{s}^{-1}]$  and  $\tilde{m} \in [9950; 32000] [\text{kg}]$ , where  $\Gamma$  is the region to the left of the hyperbola

$$\partial\Gamma = \{\sigma + j\omega \mid \sigma = \alpha, \omega^2 = 25\alpha^2 - 49/16, \alpha \leq -0.25\}$$

A stability analysis can be performed by construction of the value set at all frequencies. The system has two uncertain parameters which enter polynomially into the characteristic polynomial. In Chapter 5 the Jacobian was introduced for the construction of the value set of an uncertain polynomial with multilinear parameter dependency. This result holds for arbitrary parameter dependency and arbitrary pole regions  $\Gamma$ . Besides the edges of the operating domain also points for which the Jacobian determinant vanishes have to be used to construct the value set at each frequency. In this example the value set has to be constructed for  $\alpha = -0.6$ , i.e.  $s = -0.6 + j\sqrt{5.9375}$ . Real and imaginary part of the characteristic polynomial are determined for this frequency and the Jacobian  $\mathbf{J}(\alpha, v, \tilde{m})$  is formed. Its determinant is

$$\det \mathbf{J}(-0.6, v, \tilde{m}) = 5.18 \cdot 10^9 v + (-7.26 \cdot 10^9 - 51057\tilde{m})v^2 + (7.8 \cdot 10^8 - 44694\tilde{m} + \tilde{m}^2)v^3$$

The parameter values for which this determinant vanishes are shown in Fig. 9.5. For the construction of the value set first the edges of the uncertainty domain are mapped via the characteristic polynomial into the complex plane. In Fig. 9.5 we recognize that the  $Q$ -box is intersected by the Jacobian curve; also these points have to be mapped. The resulting value set is shown in Fig. 9.6. The set is also bounded by the image of the Jacobian curve, i.e. also interior points of the operating domain contribute to the boundary of the value set. The constructed value set does not include the origin. For a stability analysis all  $\partial\Gamma$ -parameters  $\alpha$  with  $\alpha \leq -0.25$  have to be checked.  $\square$

The previous example showed that construction of value sets is possible, independent of the selected pole region  $\Gamma$ . Color coding is introduced now in order to visualize, which operating conditions are most critical for  $\Gamma$ -stability.

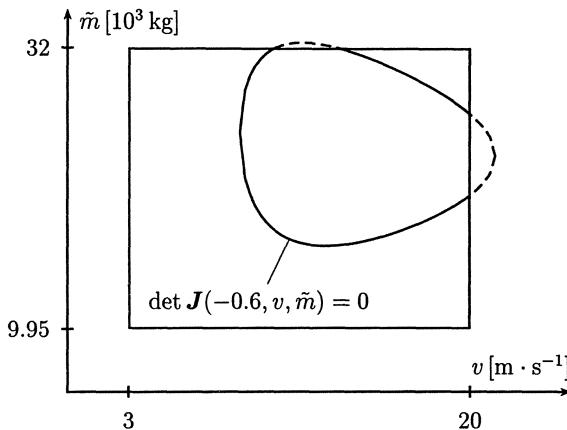


Fig. 9.5: Jacobian curve of the track guided bus for  $\alpha = -0.6$

*Example 9.14.* Consider the track guided bus as presented in the previous example. Again the value set has to be constructed but this time with color coding of the uncertainty domain. First, each vertex of the uncertainty domain is assigned a color, for instance the color red indicates minimal virtual mass  $\tilde{m}$  and maximal velocity  $v$ . A grid is laid over the  $Q$ -box and for each point on this grid a color can be interpolated from the vertex colors. The colored uncertainty domain for the bus is shown in Color plate 1 (see last pages of this book). This grid is mapped into the complex plane of the characteristic polynomial for a fixed frequency. In Color plate 3 value sets of the track guided bus for several frequencies are shown. For small frequencies the boundary of the set is formed by the boundary of the operating domain. For higher frequencies also interior points contribute to the boundary of the set. For small frequencies the operating conditions at small velocity are critical ( $\alpha = -0.36, -0.4, -0.5$ ), for  $\alpha = -0.7$  the vertex with the color blue, i.e. maximal virtual mass and maximal velocity comes closest to the origin.

□

## 9.6 Gamma-Stability Radius

Analogous to the stability radius introduced in Chapter 7, now a  $\Gamma$ -stable polynomial is assumed and the minimal perturbation which destroys the property of  $\Gamma$ -stability has to be found. Generalizations of the Tsyplkin-Polyak theorems for other regions than the half plane are not available with the exception of interval polynomials and assuming  $\Gamma$  as the unit circle [166]. But the result of Section 7.2 (affine dependency and largest hypersphere) can immediately be extended. Instead of  $s = 0$  the intersection points of  $\partial\Gamma$  with the real axis  $s = \sigma_i$  have to be used. In (7.2.10) and (7.2.11)  $s = j\omega$  is replaced

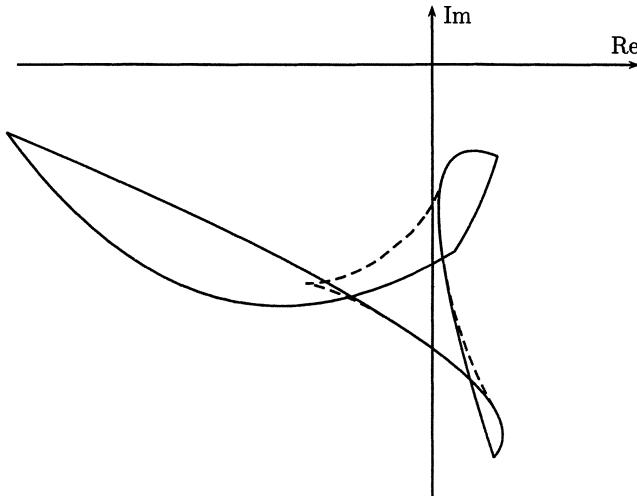


Fig. 9.6: Value set of the track guided bus for  $\alpha = -0.6$ . Its boundary is comprised of the image of the edges of the operating domain (solid) and the image of the Jacobian curves (dashed)

by  $s = \sigma(\alpha) + j\omega(\alpha)$ . Note that this has to be done for all branches of the boundary  $\partial\Gamma$ .

The relevant equations for the real root boundaries are defined in (9.3.2) and (9.3.3). The equation of the complex root boundary is the result of the elimination of  $\alpha$  from (9.3.1).

$$\text{Res}_\alpha(\text{Re } p[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}], \text{Im } p[\sigma(\alpha) + j\omega(\alpha), \mathbf{q}]) = 0 \quad (9.6.1)$$

Observe that in contrary to (9.4.26) now  $\alpha$  is eliminated. This elimination requires obviously computer algebraic programs because of the long and complicated terms.

*Example 9.15.* The stability radius of the track guided bus given in Example 7.5 with respect to the hyperbola defined in Example 9.3 is determined. The real root boundary is given in (9.4.20) and the defining equations for the complex root boundary are (9.4.24) and (9.4.25). First, the parameter  $\alpha$  has to be eliminated. The next step are the transformations  $q_1 = v - 20$  and  $q_2 = \tilde{m} - 20$ , as already executed in Example 7.5. Solving the four system of equations (7.3.1) and (7.3.2) and choosing the solution vector with the smallest norm yields the stability radius  $\rho = 8.91$  corresponding to the critical physical parameters  $v = 28.91 [\text{m} \cdot \text{s}^{-1}]$  and  $\tilde{m} = 28.91 [10^3 \text{kg}]$ . The characteristic polynomial with these parameters gives also the critical eigenvalue. There is a complex root pair on the hyperbola at  $s = -0.50 \pm 1.82j$  (Fig. 9.7).

□

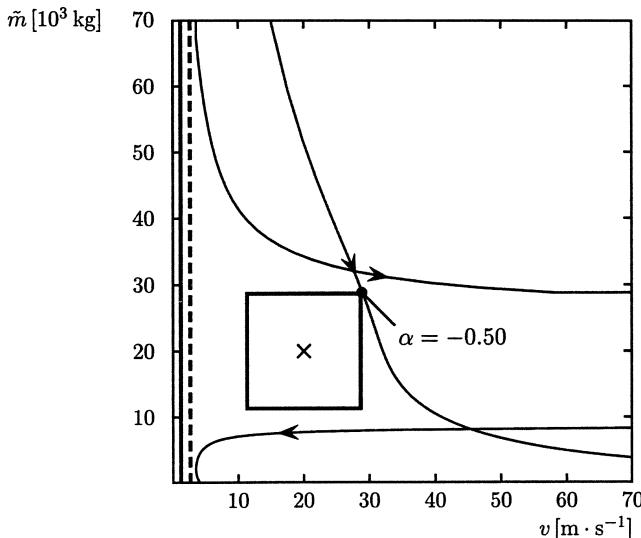


Fig. 9.7: Stability radius for a characteristic polynomial with polynomial dependency and a hyperbola as boundary of the eigenvalue region

## 9.7 Testing Sets

### *Extreme point results*

*Definition 9.1.* Given an interval polynomial. A region  $\Gamma$  is called a *weak Kharitonov region* if  $\Gamma$ -stability of all vertices implies  $\Gamma$ -stability of the entire polynomial family. If there exists a fixed number  $N$  of vertices whose  $\Gamma$ -stability guarantees the  $\Gamma$ -stability of the whole family independent of the degree, then  $\Gamma$  is called a *strong Kharitonov region*.  $\square$

The left half plane is a strong Kharitonov region with  $N = 4$  for real interval polynomials and  $N = 8$  for complex interval polynomials [105].

Observe that the unit circle is not a weak Kharitonov region, see the following counter-example of Bose and Zeheb [40]. The polynomial family

$$P(s, q_1) = \{ p(s) = s^4 + q_1 s^3 + 3/2 s^2 - 1/3 \mid q_1 \in [-17/8; 17/8] \}$$

has two vertex polynomials whose roots are inside the unit disk but has also the member  $p(s) = s^4 + 3/2 s^2 - 1/3$  whose roots are not all in the unit disk.

This at first discouraging fact for finding other regions with extreme point results did not prevent researchers to search for weak Kharitonov regions or to prove the nonexistence. A complete characterization of these regions for polynomials with complex coefficients was given by Rantzer [143]:

*Theorem 9.4. (Rantzer)*

For an open  $\Gamma$ -stability region the following two conditions are equivalent:

1. The complex family of interval polynomials

$$\begin{aligned} P(s, Q) = & \{ p(s, q) \\ = & (q_1 + jq_{n+1}) + (q_2 + jq_{n+2})s + \dots + (q_n + jq_{2n})s^{n-1} + s^n \\ | & q_i \in [q_i^- ; q_i^+], i = 1, 2, \dots, 2n \} \end{aligned}$$

is  $\Gamma$ -stable if and only if the  $2^{2n}$  extreme polynomials are  $\Gamma$ -stable.

2. The set  $\Gamma$  and the “inverse” set  $\Gamma^{-1} = \{ s \mid s^{-1} \in \Gamma \}$  are both convex.

□

For the proof of Rantzer’s theorem the reader is referred to the original paper [143].

*Remark 9.2.* Any region containing the origin is not a weak Kharitonov region. For example the unit disk is not a Kharitonov region. The disk with radius 1 and center  $-1$  is a Kharitonov region. Its inverse is the half plane to the left of a parallel to the imaginary axis at  $\sigma = -1/2$ . Further extreme point results for circles will be discussed in the context of sampled-data systems in Chapter 10. For construction of the set  $\Gamma^{-1}$  the reader is referred to Section 8.3. □

*Remark 9.3.* Note that the convexity of both  $\Gamma$  and  $\Gamma^{-1}$  is necessary for  $\Gamma$  to be a weak Kharitonov region, and this result was proven for complex coefficient polynomials. Rantzer’s theorem comprises a further result, which characterizes all open sets with property 2 of Theorem 9.4. The set  $\Gamma$  is an intersection of (not necessarily finite number of) open circles and half planes. This means that, for example, any circle not containing the origin is a weak Kharitonov region. The open set  $\Gamma$ , bounded by a branch of a hyperbola, has also the property that  $\Gamma^{-1}$  is convex. Thus, many of the desired eigenvalue locations discussed in Chapter 3 are weak Kharitonov regions. □

*Remark 9.4.* Obviously a weak Kharitonov region for complex polynomials is also a weak region for real polynomials, but not vice versa. Strong Kharitonov regions were discovered by Foo and Soh [157]. Let  $\Gamma$  be the sector described by the boundary

$$\partial\Gamma = \{ s \mid s = e^{j\Phi}\alpha, \alpha \in [0; \infty] \}$$

If  $\Phi = \frac{p}{q}\pi$  where  $p$  and  $q$  are relatively coprime positive integers and  $1 > \frac{p}{q} \geq \frac{1}{2}$  and  $\frac{\pi}{2} \leq \Phi \leq \pi$  then it is sufficient to check  $2q$  specially chosen vertex polynomials. □

### Edge results

The proof of the edge theorem in Section 5.2 was based on the fact that the value set is a parpolygon whose edges originate from edges of the  $Q$ -box, and the origin can enter the value set only through one of the edges of the parpolygon. In (5.2.64)  $s = j\omega^*$  was substituted. All arguments remain valid if  $s = \sigma(\alpha^*) + j\omega(\alpha^*)$  is substituted. Thus, the following theorem holds:

*Theorem 9.5. (Bartlett, Hollot, Huang)*

The polynomial family

$$P(s, Q) = \{ p(s, \mathbf{q}) = \sum_{i=0}^n a_i(\mathbf{q}) s^i \mid \mathbf{q} \in Q \}$$

with affine coefficient functions  $a_i(\mathbf{q})$  and  $Q = \{ \mathbf{q} \mid q_i \in [q_i^- ; q_i^+], i = 1, 2, \dots, \ell \}$  is  $\Gamma$ -stable if and only if the edges of  $Q$  are  $\Gamma$ -stable.

□

It remains to provide a  $\Gamma$ -stability test for an edge with endpoints  $\mathbf{q}_a$  and  $\mathbf{q}_b$  corresponding to the polynomials  $p_a(s) = p(s, \mathbf{q}_a)$  and  $p_b(s) = p(s, \mathbf{q}_b)$ . Since  $a(q)$  is affine, the set of polynomials corresponding to the edge of the  $Q$ -box is

$$P(s, Q) = \{ (1 - q)p_a(s) + qp_b(s) \mid q \in [0 ; 1] \}$$

A generalization of the Bialas test (Theorem 4.6 or 4.7) requires an algebraic formulation of the stability conditions, see Section 9.3. Such generalization will be given for the unit circle, i.e. Schur stability, in Chapter 10.

A brute force, but simple approach is a root locus plot for

$$(1 - q)p_a(s) + qp_b(s) = 0$$

which may be written as

$$1 + q \frac{p_b(s) - p_a(s)}{p_a(s)} = 0$$

The root locus for  $q \in [0 ; 1]$  must have  $\Gamma$ -stable end points and must not intersect any branch of  $\partial\Gamma$ . This is a necessary and sufficient condition for  $\Gamma$ -stability of the edge connecting the vertices  $\mathbf{q}_a$  and  $\mathbf{q}_b$ .

If  $\sigma_i$  are the intersection points of the real axis with  $\partial\Gamma$  and  $F(q)$  the resultant defined in (9.6.1), a generalization of Theorem 4.7, also valid for polynomial dependency, is

*Theorem 9.6.*

The one-dimensional polynomial family

$$P(s, Q) = \{ p(s) = (1 - q)p_a(s) + qp_b(s) \mid q \in [0 ; 1] \}$$

is  $\Gamma$ -stable if and only if

1.  $p_a(s)$  is  $\Gamma$ -stable
2.  $p(\sigma_i, q)$  does not vanish for  $q \in (0 ; 1]$
3.  $F(q)$  does not vanish for  $q \in (0 ; 1]$

□

Thus, a  $\Gamma$ -stability test is reduced to factorizing one polynomial and testing whether polynomials in one variable have a zero in the interval  $[0 ; 1]$ .

## 9.8 Exercises

9.1. The roots of the polynomials  $p(s) = p_0 + p_1 s + s^2$ ,  $s = \sigma + j\omega$ , should be located to the left of the left branch of the hyperbola  $\sigma^2 - \omega^2 = 1$ ,  $\sigma \leq -1$ , and within the bandwidth circle  $|s| \leq 3$ . Plot the stability-region in the  $(p_0, p_1)$ -plane.

9.2. Consider the uncertain polynomial

$$p(s, k_1, k_2) = (4 + k_1) + (2 + k_1 + k_2)s + (1 + 2k_2)s^2 + s^3$$

Which pairs  $(k_1, k_2)$  produce a damping  $> 1/\sqrt{2}$  for the roots of the polynomial?

9.3. Consider the system

$$\dot{\mathbf{x}} = \begin{bmatrix} 1 & 30 \\ -4 & -7 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

with the state feedback law  $u = -[k_1 \ k_2] \mathbf{x}$ .

The real part of the eigenvalues of the closed-loop system should be  $\leq -2$ .

- a) Determine from the feasible set of feedback vectors that one with the smallest norm  $\|\mathbf{k}\| = \sqrt{k_1^2 + k_2^2}$ .
- b) Multiply this feedback vector with a scalar factor  $\beta$  and sketch the root locus curve for  $\beta > 0$ .

9.4. Consider the system

$$\begin{aligned}\dot{\mathbf{x}} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \\ \mathbf{y} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 1 \end{bmatrix} \mathbf{x}\end{aligned}$$

with the output feedback law  $u = -[k_1 \ k_2] \mathbf{y}$ .

- a) Determine the set of all  $k_1, k_2$  such that the system is  $\Gamma$ -stable, where  $\Gamma$  is the region to the left of the left branch of the hyperbola  $\sigma^2 - \omega^2 = 1$ .
- b) Choose the solution with minimum norm  $\sqrt{k_1^2 + k_2^2}$  and sketch the eigenvalues both of the open loop and closed loop.
- c) Use this solution as nominal point and find the stability radius with respect both to a square and a circle.

9.5. Consider the system

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

with the state feedback law  $u = -[0.125 \ k_1 \ k_2] \mathbf{x}$ .

- a) Determine in the  $(k_1, k_2)$ -plane the region, such that the closed-loop system has a damping  $D$  of at least  $1/\sqrt{2}$ .
- b) Determine the feedback vector with minimum norm  $\sqrt{k_1^2 + k_2^2}$  which yields a damping  $D$  of at least  $1/\sqrt{2}$ . Where are the corresponding eigenvalues located?
- c) Use this feedback vector as nominal point and find the stability radius with respect both to a square and a circle.

9.6. Consider the crane example with  $g = 10 \text{ [m} \cdot \text{s}^{-2}]$  and fixed crab mass  $m_C = 1000 \text{ [kg]}$ . For  $m_L \in [50 ; 2395]$  the controller

$$u = -[500 \ 2769 \ -21556 \ 0] \mathbf{x}$$

$\Gamma$ -stabilizes the system if  $\Gamma$  is the hyperbola  $4\sigma^2 - \omega^2 = 0.25$ .

- a) What is the maximum load mass variation for  $\ell = 10$ .
- b) Is it possible to simultaneously  $\Gamma$ -stabilize the system for  $\ell = 10$  and  $\ell = 20$  with fixed  $m_L = 1000$ ?

# 10 Robustness of Sampled-Data Control Systems

Controllers are usually implemented in a digital computer. Fig. 10.1 shows a single-loop sampled-data control system with  $c_z(z)$  representing the  $z$ -transfer function of the digital controller, and  $(1 - e^{-Ts})/s$  as the transfer function of the hold element,  $T$  is the sampling interval.

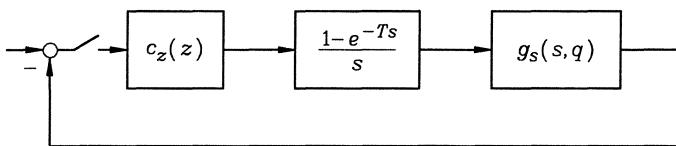


Fig. 10.1: Sampled-data control system with uncertain physical parameters  $q$  in the continuous-time plant

The problems and solution approaches treated in Chapters 1 through 9 have their counterpart for sampled-data control systems and the Sections 10.1 to 10.9 are organized accordingly. Before we go into details we should however discuss an obvious question:

Assume that a robust continuous controller was designed for the plant  $g_s(s, q)$ . Is it possible to find an approximately equivalent discrete-time controller  $c_z(z)$  that preserves robust stability?

Discretization of a continuous-time controller is a topic of current research interest. For a nominal plant  $g_s(s)$ , a controller discretization procedure was recently shown in [125] that preserves the closed-loop stability. The controller however depends on the nominal plant model and therefore does not meet our requirement of a fixed gain robust controller for an uncertain plant.

Since robust stability is not guaranteed after controller discretization, there remain two alternatives:

- try a controller discretization procedure that does not involve the plant transfer function, e.g. Tustin approximation with

$$s \approx \frac{2(z-1)}{T(z+1)} \quad (10.0.1)$$

and do a robust stability analysis for the resulting discrete-time system. For “sufficiently small” sampling intervals  $T$  the approximation is good. Essentially it replaces each continuous integrator  $1/s$  in the controller by a discrete approximation using the trapezoidal rule for integration with  $z$ -transfer function  $T(z+1)/2(z-1)$ .

- b) discretize the plant model and try any design tool to simultaneously stabilize the plant for some representative operating conditions. Then do a robust stability analysis for the continuum of parameter values. If we adopt this *multi-model approach* then design of the discrete-time controller is not more complicated than the continuous-time design, because numerator and denominator coefficients of  $c_z(z)$  in Fig. 10.1 enter linearly into the closed-loop characteristic polynomial like in the continuous-time case.

For both approaches a robust stability analysis of the resulting closed-loop discrete-time system is required, in case a) because stability cannot be guaranteed after approximate controller discretization, and in case b) because only representative operating conditions were considered in the design.

## 10.1 Plant Discretization

In Chapter 1 we have seen examples of parametric state-space and transfer-function models of continuous-time plants. Now the plant inputs  $u(t)$  are generated by a sampler and hold, i.e.

$$u(t) = u(iT), \quad t \in [iT; iT + T], \quad i = 0, 1, 2, \dots \quad (10.1.1)$$

and the plant model may be discretized to obtain a description of states and signals at the sampling instants  $t = iT$ . The discretization process is derived in many textbooks on sampled-data control systems, e.g. [23, 67, 5] and the results are only summarized here.

A state space model

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{q})\mathbf{x} + \mathbf{B}(\mathbf{q})\mathbf{u} \\ \mathbf{y} &= \mathbf{C}(\mathbf{q})\mathbf{x} \end{aligned} \quad (10.1.2)$$

with input (10.1.1) yields the discretized system

$$\begin{aligned} \mathbf{x}(iT + T) &= \mathbf{A}_d(\mathbf{q})\mathbf{x}(iT) + \mathbf{B}_d(\mathbf{q})\mathbf{u}(iT) \\ \mathbf{y}(iT) &= \mathbf{C}(\mathbf{q})\mathbf{x}(iT) \end{aligned} \quad (10.1.3)$$

Let

$$\mathbf{R}(\mathbf{q}) := \int_0^T e^{\mathbf{A}(\mathbf{q})v} dv \quad (10.1.4)$$

Then

$$\begin{aligned}\mathbf{A}_d(\mathbf{q}) &= e^{\mathbf{A}(\mathbf{q})T} = \mathbf{I} + \mathbf{R}(\mathbf{q})\mathbf{A}(\mathbf{q}) \\ \mathbf{B}_d(\mathbf{q}) &= \mathbf{R}(\mathbf{q})\mathbf{B}(\mathbf{q})\end{aligned}\quad (10.1.5)$$

Each eigenvalue of  $\mathbf{A}$  at  $s_i$  is mapped to an eigenvalue of  $\mathbf{A}_d$  at  $z_i = e^{s_i T}$  with the same multiplicity.

The  $z$ -transfer function matrix of the system (10.1.3) with  $z = e^{sT}$  is

$$H_z(z, \mathbf{q}) = \mathbf{C}(\mathbf{q})[z\mathbf{I} - \mathbf{A}_d(\mathbf{q})]^{-1}\mathbf{B}_d(\mathbf{q}) \quad (10.1.6)$$

and in the single-input, single-output case

$$h_z(z, \mathbf{q}) = \mathbf{c}^T(\mathbf{q})[z\mathbf{I} - \mathbf{A}_d(\mathbf{q})]^{-1}\mathbf{b}_d(\mathbf{q}) \quad (10.1.7)$$

An alternative approach to plant discretization starts from the plant transfer function  $g_s(s, \mathbf{q})$ . ( $g_s(s, \mathbf{q})$  may also be an element of a transfer function matrix.) First, the  $z$ -transfer function for sampler, hold, and plant is determined as

$$h_z(z, \mathbf{q}) = \frac{z-1}{z} \mathcal{Z} \left\{ \frac{g_s(s, \mathbf{q})}{s} \right\} \quad (10.1.8)$$

or equivalently by the Poisson sum [119, 118]

$$h_z(e^{sT}, \mathbf{q}) = (1 - e^{-sT}) \sum_{m=-\infty}^{\infty} \frac{g_s(s + jm2\pi/T, \mathbf{q})}{sT + jm2\pi} \quad (10.1.9)$$

*Example 10.1.*

$$g_s(s, q) = \frac{1}{s(1 + s/q)} \quad (10.1.10)$$

$$h_z(z, q) = \frac{(1 - qTe^{-qT} - e^{-qT}) - (1 - qT - e^{-qT})z}{q(z-1)(z - e^{-qT})} \quad (10.1.11)$$

The poles of  $g_s(s, \mathbf{q})$  at  $s_1 = 0$  and  $s_2 = -q$  are transformed to poles of  $h_z(z, \mathbf{q})$  at  $z_1 = e^{s_1 T} = 1$  and  $z_2 = e^{s_2 T} = e^{-qT}$ . The zero of  $h_z(z, \mathbf{q})$ , however, does not even have a continuous-time counterpart. In the numerator it is not possible to introduce only one new uncertain variable  $e^{-qT}$  instead of  $q$  because  $qT$  also enters in form of a sum and a product. Therefore, we have to deal with an exponential parameter dependency that also enters into the closed-loop characteristic polynomial.

The Poisson form of the  $z$ -transfer function is

$$h_z(e^{sT}, q) = \frac{1 - e^{-sT}}{T} \sum_{m=-\infty}^{\infty} \frac{1}{(s + jm2\pi/T)^2[1 + (s + jm2\pi/T)/q]} \quad (10.1.12)$$

For numerically given  $q$ , the form (10.1.11) of the  $z$ -transfer function is more convenient. For uncertain  $q$ , however, (10.1.12) is a useful form because  $h_z(e^{sT}, q)$  may be approximated by truncation of the series, see Section 10.8. This approximation is rational in  $q$ .  $\square$

## 10.2 Discrete-time Controllers

Controllability and observability of a controllable single-input single-output plant get lost in the discretization process, whenever two distinct eigenvalues  $s_1 \neq s_2$  of the continuous-time plant map into identical eigenvalues  $e^{s_1 T} = e^{s_2 T}$  of the discrete-time plant [100].

*Example 10.2.* Crane

By (1.1.7) the crane has eigenvalues on the imaginary axis at  $s_{3,4} = \pm j\sqrt{g(m_L + m_C)/\ell m_C}$ . Controllability gets lost for  $e^{s_3 T} = e^{s_4 T}$ , i.e.  $T_k = k\pi\sqrt{\ell m_C/g(m_L + m_C)}$ ,  $k = 1, 2, \dots$ . For  $T = T_1$  two distinct eigenvalues of the continuous system coincide after transformation at  $z = -1$  in the discrete-time model. A good choice for  $T$  is  $T \leq T_1/4$ .

In the parameter dependent case the sampling interval must be chosen for the worst case  $T_1 = T_1^-$ , i.e.

$$T \leq \frac{\pi}{4} \sqrt{\frac{\ell^-}{g(1 + m_L^+ / m_C^-)}} \quad (10.2.1)$$

If we let the minimum rope length  $\ell^-$  go to zero then also  $T$  goes to zero. Therefore, it is necessary to specify positive values for  $\ell^-$  and  $m_C^-$  in order to obtain a nonzero  $T$ .  $\square$

*Remark 10.1.* From a practical point of view one may want to fix the sampling interval  $T$  at an early stage of analysis and design. In the case of uncertain parameters the usual rules for choosing  $T$  apply to the operating condition  $\mathbf{q}^*$  with the fastest plant dynamics or largest bandwidth. Alternatively,  $T$  may be treated as an additional undetermined parameter in robustness analysis.  $T$  is not an additional parameter if the system is *scalable* [14]. Roughly speaking a system with  $\ell$  uncertain parameters  $\mathbf{q}$  is scalable if there is a rescaling of  $\mathbf{q}$  that changes only the time scale of all solutions to the differential equations describing the plant. It is then possible to introduce  $\ell$  new parameters in an  $\ell$ -vector  $\mathbf{r} = \mathbf{r}(\mathbf{q}, T)$ , i.e. the number of uncertain parameters is not increased by allowing an uncertain  $T$ . Conceptually, scalability is important if we want to compare continuous and sampled systems by their respective stability regions in the same scaled parameter space, where for  $T \rightarrow 0$  the same bounds are obtained.  $\square$

The general discussion on controller structures in Chapter 2 remains valid for discrete-time controllers. There are however some additional considerations: Discrete-time compensators for single-loop systems are assumed with the same numerator and denominator degree. A nonzero relative degree would introduce undesirable time delays into the loop. In continuous-time systems frequently a strictly proper controller transfer function is assumed in order to reduce high frequency disturbances (e.g. measurement noise, high frequency model uncertainty). In a sampled-data system the same effect can be achieved by an analog anti-aliasing filter before the sampler. This is a low pass filter with a bandwidth  $\omega_B$  of at most half the sampling frequency, i.e.  $\omega_B \leq 2\pi/T$ .

The problem classes of parametric polynomials listed in Section 2.7 (interval, affine, multilinear, polynomial) must be augmented by polynomials with an even more complicated *exponential* parameter dependency.

*Example 10.3.* Consider the sampled-data feedback loop of Fig. 10.1 with  $g_s(s, q)$  as in (10.1.10) and a proportional controller  $c_z(z) = k$ . The closed-loop characteristic polynomial is

$$\begin{aligned} p(z, q, k) &= a_0 + a_1 z + a_2 z^2 \\ a_0 &= qe^{-qT} + k(1 - qTe^{-qT} - e^{-qT}) \\ a_1 &= -q(1 + e^{-qT}) - k(1 - qT - e^{-qT}) \\ a_2 &= q \end{aligned} \tag{10.2.2}$$

The coefficient functions are exponential for the uncertain plant parameter  $q$ , they are affine for the controller parameter  $k$ .  $\square$

## 10.3 Eigenvalue Specifications

The open-loop eigenvalue locations of sampled-data systems depend on the sampling interval. An extreme case is obtained with  $T \rightarrow 0$  and all eigenvalues  $z_i = e^{s_i T}$  approaching  $z = 1$ . Practically,  $T$  is chosen such that the eigenvalues are in some neighborhood of  $z = 1$ . In [5] a rule of thumb is given:  $T$  should be chosen such that all open-loop eigenvalues are located in a unit circle centered at  $z = \sqrt{2}$ .

For closed-loop stability these eigenvalues must be shifted inside the unit circle. The eigenvalue locations  $s_i$  recommended in Section 3.1 may be directly mapped into the  $z$ -plane via  $z_i = e^{s_i T}$ . The type of discrete-time solutions is quite similar to the corresponding continuous-time solutions as long as  $\operatorname{Re} z_i > 0$ . The mapping of lines of constant damping and constant natural frequency from the  $s$ -plane to the  $z$ -plane is illustrated by Fig. 10.2.

A special feature of sampled-data systems is the existence of deadbeat solutions. They are achieved with all eigenvalues located at  $z = 0$ . Practically this requires relatively long sampling intervals  $T$  in order to avoid excessive magnitudes of the plant inputs  $u$ . Deadbeat control is an ideal case achievable with finite controller gains. We can come close to the ideal case by shifting the eigenvalues close to the origin.

Another special case in sampled-data systems occurs for an eigenvalue at  $z = -1$ . The corresponding solution term at the plant input is shown in Fig. 10.3.

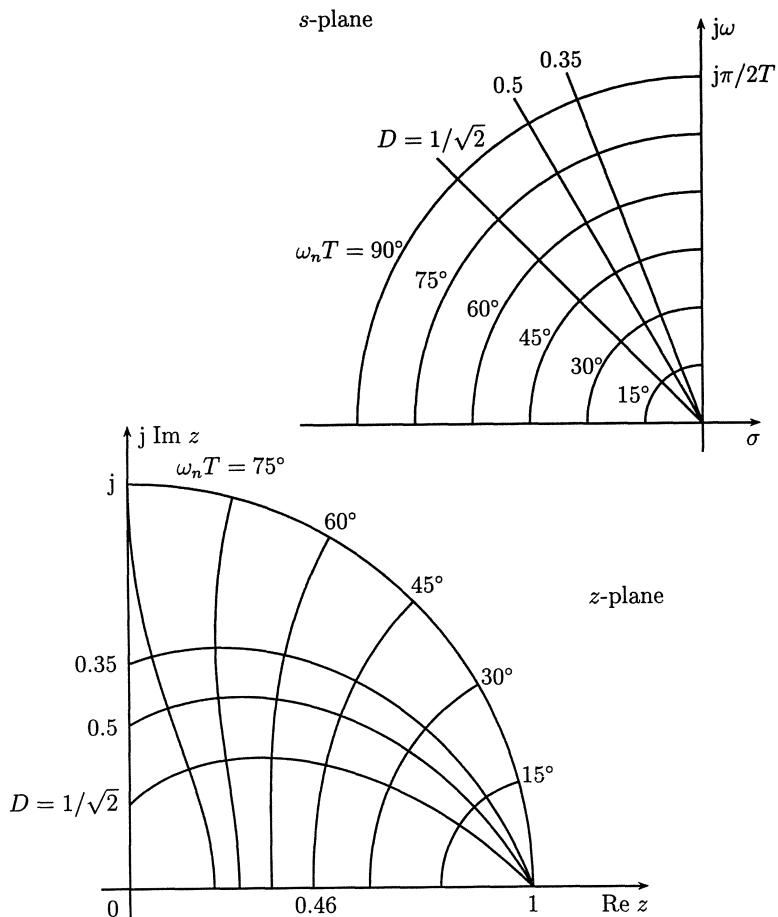
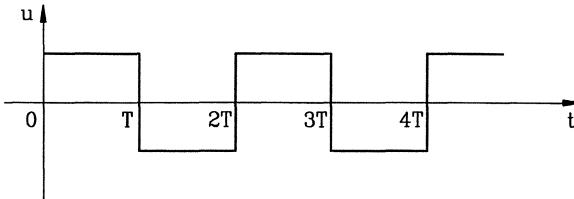


Fig. 10.2: Mapping of constant damping ( $D$ ) lines and constant natural frequency ( $\omega_n$ ) lines from the  $s$ -plane to the plane  $z = e^{sT}$

Fig. 10.3: Solution term corresponding to an eigenvalue at  $z = -1$ 

Consider the situation for a sampling interval  $T$  that is long compared to the settling time of the plant step response. Then the response has approximately reached its stationary value before the next sampling occurs and the plant response to the input signal shown in Fig. 10.3 consists of consecutive full length step responses. This type of solution and its neighborhood is obviously very undesirable. It is an indication either of a too long sampling interval compared to the plant dynamics or of a bad design that has placed closed-loop eigenvalues too close to  $z = -1$ . Closed-loop poles on the negative real axis at  $z_i \in (-1; 0)$  give rise to decaying solution terms with alternating sign of  $u(iT)$ . Such excessive actuator activity is inefficient and should be avoided.

In later sections we will formally treat the real root stability boundaries at  $z = 1$  and  $z = -1$ . We should keep in mind that a good practical solution cannot be close to the  $z = -1$  boundary. Closeness to the  $z = 1$  boundary may be tolerable if the sampling interval  $T$  is short. For example an eigenvalue at  $z = 0.98$  yields a solution term  $0.98^i$  that decays only by 2% from sampling instant to sampling instant. For small  $T$  this may be a rapid exponential decay in real time  $t$ .

We will now move towards a definition of a useful  $\Gamma$ -stability region in the  $z$ -plane that takes into account the above discussion and is also computationally tractable. Consider the  $\Gamma$ -stability regions shown in Fig. 10.4.

Constant damping lines in  $s$ -plane map into logarithmic spirals in  $z$ -plane, constant real part lines in  $s$ -plane map into circles centered at  $z = 0$ . Three examples with increasing damping (0.35, 0.5, 0.707) and decreasing real part boundary are mapped via  $z = e^{sT}$  into the corresponding boundaries in the  $z$ -plane.

In the  $s$ -plane we have smoothed the piecewise defined boundary by a hyperbola, see Fig. 3.6, because a hyperbola is reasonably well tractable as a  $\Gamma$ -stability boundary. In the  $z$ -plane we smoothen the image by an even simpler boundary, that is a circle. Let  $z = \tau + j\eta$  and consider a circle

$$(\tau - \tau_0)^2 + \eta^2 = r^2 \quad (10.3.1)$$

The circle has center  $\tau_0$  and radius  $r$ . Now let

$$\tau_0 = \begin{cases} r & \text{for } 0 \leq r \leq 0.5 \\ 1 - r & \text{for } 0.5 \leq r \leq 1 \\ 0 & \text{for } 1 \leq r \end{cases} \quad (10.3.2)$$

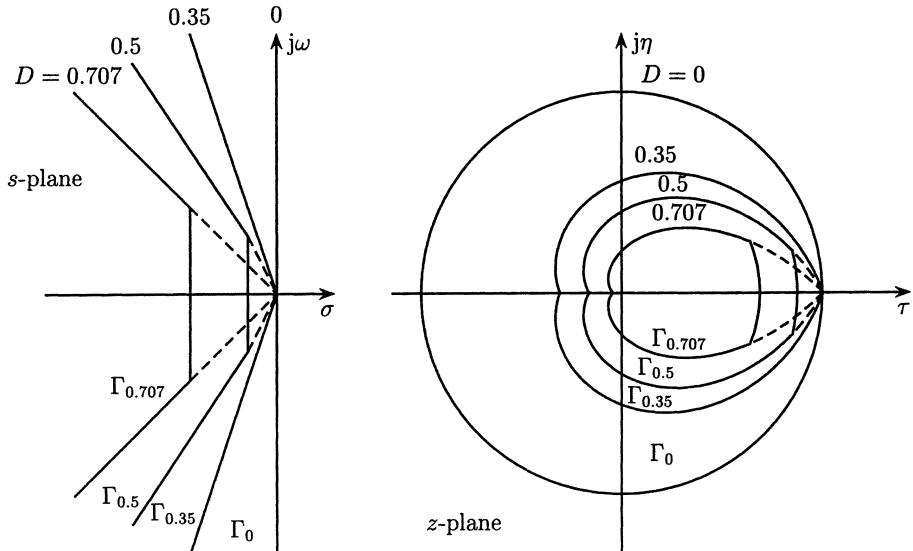


Fig. 10.4: Mapping of  $\Gamma$ -stability regions from  $s$ -plane to  $z$ -plane

Fig. 10.5 shows some circles for  $r = 1, 0.8, 0.6, 0.5, 0.44, 0.33$ .

Given that we are talking about a rule of thumb, the  $\Gamma$ -stability regions are reasonably well approximated with the following correspondence

$D$	$r$	$\tau_0$
0	1	0
0.35	0.5	0.5
0.5	0.44	0.44
0.707	0.33	0.33

Conceptually a design process may be viewed as follows. By choice of the sampling interval  $T$  the open-loop eigenvalues are located in a unit circle centered at  $\sqrt{2}$ . Thus, a circle with radius  $r = 1 + \sqrt{2} \approx 2.4$  includes these eigenvalues for all operating conditions  $q \in Q$ . The open loop is ( $r = 2.4$ )-stable. In the design process we move away from the open loop  $k = 0$  in  $k$ -space. The design direction should be such that  $r$  is reduced. For  $r = 1$  Schur stability is achieved, for  $r = 0.6$  a solution term  $(-0.2)^i$  is still admitted but no alternating terms that decay slower. For  $r = 0.5$  all alternating terms are ruled out but the deadbeat solution is still admitted. A further improvement in the degree of stability is now achieved by pushing eigenvalues in the right half  $z$ -plane

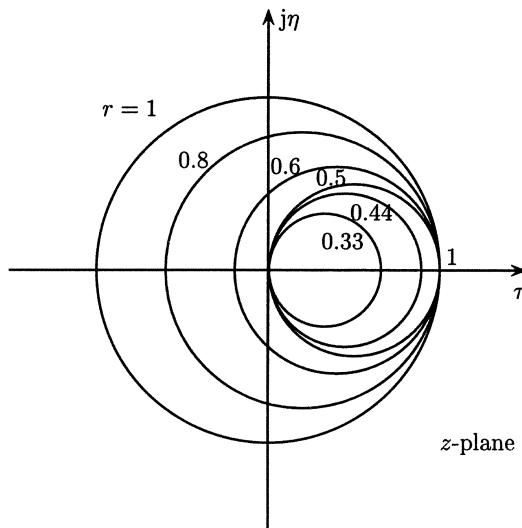


Fig. 10.5: Approximation of  $\Gamma$ -stability regions of Fig. 10.4 by circles

closer to the origin. The ideal deadbeat solution cannot be achieved simultaneously for a plant family, but it is a well-posed question to ask for the smallest radius  $r_{min}$  that can be achieved simultaneously by a controller of fixed structure.

*Remark 10.2.* If a strict contraction is desired, then the relationship between  $\tau_0$  and  $r$  may be changed to

$$\tau_0(1 - \tau_0) = \alpha r(1 - r) \quad (10.3.3)$$

with  $\alpha < 1$  instead of  $\alpha = 1$  in (10.3.2).  $\square$

## 10.4 Classical Stability Tests

The classical approaches of Chapter 4 can easily be modified for testing Schur stability of polynomial families

$$P(z, \mathbf{q}) = \{ p(z, \mathbf{q}) = \sum_{i=0}^n a_i(\mathbf{q}) z^i \mid \mathbf{q} \in Q \} \quad (10.4.1)$$

The root set calculation by gridding all parameter intervals is the same as for continuous time, only the interpretation is different because we are now interested in the root location relative to the unit circle. If we rely on a gridding approach for robustness

analysis, then also the discretization (10.1.3) may be performed numerically on a  $\mathbf{q}$ -grid.

#### *Boundary crossing*

The boundary crossing idea for discrete-time systems was formulated by Jury and Pavlidis [95]. There are three boundaries

- i) a real root boundary at  $z = 1$ , i.e.

$$p(1, \mathbf{q}) = 0 \quad (10.4.2)$$

- ii) a real root boundary at  $z = -1$ , i.e.

$$p(-1, \mathbf{q}) = 0 \quad (10.4.3)$$

- iii) a complex root boundary for a root on the upper half of the unit circle, i.e.

$$p(e^{j\omega T}, \mathbf{q}) = 0, \quad \omega T \in [0; \pi] \quad (10.4.4)$$

The polynomial with real coefficients then also has a symmetric root on the lower half of the unit circle.

#### *Algebraic problem formulation*

The algebraic problem formulation corresponding to Theorem 4.5 (Frazer, Duncan) is

#### *Theorem 10.1. (Jury, Pavlidis)*

The polynomial family  $P(z, Q) = \{ p(z, \mathbf{q}) \mid \mathbf{q} \in Q \}$  with  $p(z, \mathbf{q})$  continuous is Schur stable if and only if

1. there exists a  $\mathbf{q}^0 \in Q$  such that the polynomial  $p_0(z) = p(z, \mathbf{q}_0)$  is Schur stable
2.  $p(1, \mathbf{q}) \neq 0$  for all  $\mathbf{q} \in Q$
3.  $p(-1, \mathbf{q}) \neq 0$  for all  $\mathbf{q} \in Q$
4.  $\det \mathbf{S}(\mathbf{q}) \neq 0$  for all  $\mathbf{q} \in Q$ , where  $\mathbf{S}(\mathbf{q}) = \mathbf{X}(\mathbf{q}) - \mathbf{Y}(\mathbf{q})$  and (omitting the dependency on  $\mathbf{q}$ )

$$\mathbf{X} = \begin{bmatrix} a_n & a_{n-1} & a_{n-2} & \cdots & a_2 \\ 0 & a_n & a_{n-1} & \cdots & a_3 \\ 0 & 0 & a_n & \cdots & a_4 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & a_n \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} 0 & 0 & 0 & \cdots & a_0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_0 & \cdots & a_{n-4} \\ 0 & a_0 & a_1 & \cdots & a_{n-3} \\ a_0 & a_1 & a_2 & \cdots & a_{n-2} \end{bmatrix} \quad (10.4.5)$$

□

*Proof.*

In [95], the following counterpart to Orlando's formula was shown

$$\det \mathbf{S} = a_n^{n-1} \prod_{i=1}^{n-1} \prod_{k=i+1}^n (1 - z_i z_k) \quad (10.4.6)$$

where  $z_i, z_k$  are roots of  $p(z)$ . Consider now a continuous variation of  $\mathbf{q}$  starting with  $\mathbf{q}^0$ , i.e. a stable polynomial. If a conjugate pair of roots  $z_i, z_k$  crosses the unit circle, then  $1 - z_i z_k = 0$  and  $\det \mathbf{S} = 0$ . The only other possibilities for the polynomial to become unstable are that a real root crosses the unit circle at  $z = 1$  (condition 2) or  $z = -1$  (condition 3). Then Theorem 10.1 follows from the Boundary Crossing Theorem 9.1.

□

*Remark 10.3.* Complete tests for a polynomial to have all its roots inside the unit circle have been given by Schur [148, 149] and Cohn [47]. A simplified formulation for polynomials with real coefficients was given by Jury [94]. Since we always assume that we know a stable polynomial to start with and use the Boundary Crossing Theorem, we do not need the complete set of algebraic inequalities. Here we only give the inequalities corresponding to conditions 2, 3 and 4 of Theorem 10.1.

$$\begin{aligned} p(1) &> 0 \\ (-1)^n p(-1) &> 0 \\ \det \mathbf{S} &> 0 \end{aligned} \quad (10.4.7)$$

□

*Example 10.4.* Consider the characteristic polynomial (10.2.2). Let  $T = 1$ ,  $Q = \{q, k \mid q \in [0.1; 2], k \in [0.1; 2]\}$ . Is this polynomial family Schur stable?

$$\begin{aligned} p(z, q, k) &= a_0 + a_1 z + a_2 z^2 \\ a_0 &= qe^{-q} + k(1 - qe^{-q} - e^{-q}) \\ a_1 &= -q(1 + e^{-q}) - k(1 - q - e^{-q}) \\ a_2 &= q \end{aligned}$$

1. Let  $q = 1$ ,  $k = 1$ , then  $p_z(z, 1, 1) = (1 - e^{-1}) - z + z^2$  has roots inside the unit circle. The three stability boundaries are given by
2.  $p(1, q, k) = kq(1 - e^{-q}) \neq 0$
3.  $p(-1, q, k) = 2q(1 + e^{-q}) + k[2(1 - e^{-q}) - q(1 + e^{-q})] \neq 0$
4.  $X = a_2$ ,  $Y = a_0$ ,  $S = a_2 - a_0 = q(1 - e^{-q}) - k(1 - qe^{-q} - e^{-q}) \neq 0$

Condition 2 yields the boundaries  $q = 0$  and  $k = 0$ , which are outside  $Q$ . Conditions 3 and 4 may be checked graphically by plotting the stability boundaries in the  $(q, k)$ -plane.

From condition 3 follows the stability boundary at  $z = -1$  with

$$k_{-1} = \frac{2q(1 + e^{-q})}{q(1 + e^{-q}) - 2(1 - e^{-q})}$$

From condition 4 follows the complex root boundary at  $z = e^{j\omega T}$  with

$$k_c = \frac{q(1 - e^{-q})}{1 - qe^{-q} - e^{-q}}, \quad \lim_{q \rightarrow 0} k_c = 2$$

The stability boundaries  $k_{-1}$  and  $k_c$  are plotted in Fig. 10.6 together with the boundaries  $k = 0$  and  $q = 0$  resulting from condition 2. Fig. 10.6 not only provides a test for the given  $Q$ -box, but the entire stable region is exhibited. For comparison consider the continuous-time system (10.1.10) with feedback  $k$ . Its characteristic polynomial is  $qk + qs + s^2$ , it is stable for  $q > 0, k > 0$ , i.e. in the entire first quadrant in Fig. 10.6. The situation that sampling reduces stability regions is typical, however, it is possible to construct (somewhat exotic) examples for which sampling stabilizes [14].

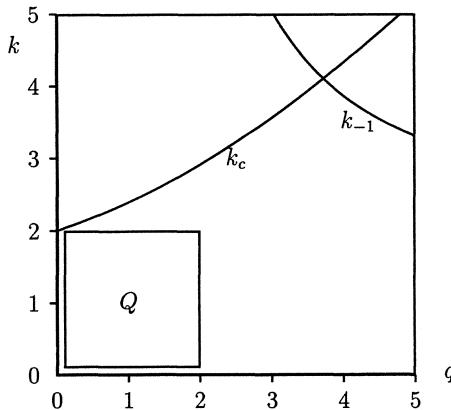


Fig. 10.6: The stability boundaries do not intersect  $Q$ , and  $Q$  contains a stable point at  $q = 1, k = 1$ . Therefore, the polynomial family is robustly Schur stable.

Note that in the example the loop gain  $k$  enters affinely into the characteristic polynomial. Therefore, it was easy to solve explicitly for the stability boundary  $k(q)$ . In general it is not possible to test conditions 3 and 4 of Theorem 10.1 without gridding  $\mathbf{q}$ .  $\square$

An alternative algebraic formulation of stability criteria uses a bilinear transformation of the polynomial  $p(z)$  by

$$w := \frac{z - 1}{z + 1}, \quad z = \frac{1 + w}{1 - w} \quad (10.4.8)$$

It maps the unit disk of the  $z$ -plane onto the left half  $w$ -plane. Substituting  $z = (1+w)/(1-w)$  into  $p(z)$  of (10.4.1) a new polynomial  $p_w(w)$  is obtained as

$$\begin{aligned} p_w(w) &= (1-w)^n p\left(\frac{1+w}{1-w}\right) \\ &= \sum_{i=0}^n a_i (1+w)^i (1-w)^{n-i} \\ &= \sum_{i=0}^n b_i w^i \end{aligned} \quad (10.4.9)$$

$p(z)$  is a Schur polynomial if and only if  $p_w(w)$  is a Hurwitz polynomial. Thus, results of Chapters 4 to 9 can be adapted to the discrete-time case. Of course an exponential dependency of  $a_i(\mathbf{q})$  is inherited by  $b_i(\mathbf{q})$  and the robustness analysis in terms of physical plant parameters  $\mathbf{q}$  remains a difficult problem.

*Example 10.5.* Consider again Example 10.4. The transformed polynomial is

$$p_w(w, q, k) = b_0 + b_1 w + b_2 w^2$$

$$b_0 = a_0 + a_1 + a_2 = kq(1 - e^{-q})$$

$$b_1 = 2(a_2 + a_0) = 2[q(1 - e^{-q}) - k(1 - qe^{-q} - e^{-q})]$$

$$b_2 = a_0 - a_1 + a_2 = 2q(1 + e^{-q}) + k[2(1 - e^{-q}) - q(1 + e^{-q})]$$

The boundaries for Hurwitz stability, i.e.  $b_0 = 0, b_1 = 0, b_2 = 0$ , are identical to the Schur stability boundaries of Example 10.4 with the result plotted in Fig. 10.6.  $\square$

The correspondence between the critical stability conditions of  $p(z, \mathbf{q})$  and  $p_w(w, \mathbf{q})$  is the following

$$\begin{aligned} p(1, \mathbf{q}) = 0 &\iff b_0(\mathbf{q}) = 0 \\ p(-1, \mathbf{q}) = 0 &\iff b_n(\mathbf{q}) = 0 \\ \det \mathbf{S}(\mathbf{q}) = 0 &\iff \det \mathbf{H}_{n-1}(\mathbf{q}) = 0 \end{aligned} \quad (10.4.10)$$

The bilinear transformation also provides a set of relatively simple necessary stability conditions

$$b_i(\mathbf{q}) > 0, \quad i = 0, 1, \dots, n \quad (10.4.11)$$

The necessary condition (10.4.11), interpreted in terms of the original coefficients  $a_i$ , is closely related to the following result on the convex hull of the stable region in coefficient space [64]

*Theorem 10.2. (Fam, Meditch )*

Consider the stability region of

$$p(z) = a_0 + a_1 z + \dots + a_{n-1} z^{n-1} + z^n \quad (10.4.12)$$

in the space of coefficients  $a_0, a_1, \dots, a_{n-1}$ . Its convex hull is a polytope with  $n+1$  vertices corresponding to the polynomials

$$p_i(z) = (z+1)^i(z-1)^{n-i}, \quad i = 0, 1, \dots, n \quad (10.4.13)$$

□

It was shown in [5] that the above convex hull is formed by hyperplanes representing the necessary stability conditions  $b_i(a_0, a_1, \dots, a_{n-1}) > 0$ ,  $i = 0, 1, \dots, n$ .

*Example 10.6.* Necessary conditions for Schur stability of a given polynomial

$$p(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + z^4$$

The corresponding vertex polynomials are

$$p_0 = (1-w)^4 = 1 - 4w + 6w^2 - 4w^3 + w^4$$

$$p_1 = (1-w)^3(1+w) = 1 - 2w + 2w^3 - w^4$$

$$p_2 = (1-w)^2(1+w)^2 = 1 - 2w^2 + w^4$$

$$p_3 = (1-w)(1+w)^3 = 1 + 2w - 2w^3 - w^4$$

$$p_4 = (1+w)^4 = 1 + 4w + 6w^2 + 4w^3 + w^4$$

The transformed polynomial is

$$p_w(w) = \sum_{i=0}^4 a_i p_i(w) = \sum_{i=0}^4 b_i w^i$$

Necessary conditions for Schur stability of  $p(z)$  are

$$b_0 = a_0 + a_1 + a_2 + a_3 + 1 > 0$$

$$b_1 = 2(-2a_0 - a_1 - a_3 + 2) > 0$$

$$b_2 = 2(3a_0 - a_2 + 3) > 0$$

$$b_3 = 2(-2a_0 + a_1 - a_3 + 2) > 0$$

$$b_4 = a_0 - a_1 + a_2 - a_3 + 1 > 0$$

These five linear inequalities correspond to five three-dimensional hyperplanes in the four-dimensional space with coordinates  $a_0, a_1, a_2, a_3$ . The hyperplanes form the convex hull of the stability region.

The five inequalities can be combined as

$$|a_1 + a_3| - (1 + a_0) < a_2 < 3(1 + a_0)$$

$$|a_1 - a_3| < 2(1 - a_0)$$

□

Necessary stability conditions have only limited value in robustness analysis, because we have to be pessimistic in analysis. For design, however, ( $\mathbf{q}$  replaced by the undetermined controller coefficients  $\mathbf{k}$ ), necessary stability conditions are useful because they yield a finite region in  $\mathbf{k}$ -space where possible solutions must be located.

#### *Frequency domain problem formulation*

In frequency domain problem formulations we have to substitute  $s = j\omega$ , i.e.  $z = e^{j\omega T}$ . Then an initially stable polynomial  $p(z, \mathbf{q})$  will become unstable by variation of  $\mathbf{q}$  whenever we have simultaneously

$$\begin{aligned} \operatorname{Re} p(e^{j\omega T}, \mathbf{q}) &= 0 \\ \operatorname{Im} p(e^{j\omega T}, \mathbf{q}) &= 0 \end{aligned} \quad (10.4.14)$$

for some  $\omega T \in [0; \pi]$ . For  $\omega T = 0$  or  $\omega T = \pi$  the second condition of (10.4.14) is always satisfied and there remains only the first condition  $p(1, \mathbf{q}) = 0$  or  $p(-1, \mathbf{q}) = 0$ .

#### *Parameter space approach*

In the parameter space approach for two uncertain parameters  $\mathbf{q} = [q_1 \ q_2]$  we try to eliminate  $q_1$  or  $q_2$  from the two equations (10.4.14). This was possible up to polynomial coefficient functions as shown in Chapter 9. This approach is no more feasible if  $q_1$  and  $q_2$  enter exponentially.

The parameter space approach will be used in Chapter 11 as a design tool in  $\mathbf{k}$ -space. For single-input plants the controller can always be assumed in a form such that the undetermined controller parameters in  $\mathbf{k}$  enter affinely into the coefficients of  $p(z, \mathbf{k})$ . In the multi-input case the controller structure can be assumed such that the coefficient functions are multilinear. For such design considerations it is useful to have some idea of the shape of stability regions in the space of coefficients  $a_i$ . We will treat the examples of polynomials with degree  $n = 2$  and  $n = 3$ .

*Example 10.7.*  $p(z) = a_0 + a_1 z + z^2$

The polynomial is Schur stable for  $a_0 = 0, a_1 = 0$ . Find the stable neighborhood of this point in the coefficient plane.

By (10.4.2) and (10.4.3) the real root boundaries result from

$$\begin{aligned} p(1) &= a_0 + a_1 + 1 = 0 \\ p(-1) &= a_0 - a_1 + 1 = 0 \end{aligned}$$

These two straight lines are plotted in the  $(a_1, a_2)$ -plane of Fig. 10.7. By (10.4.4) the complex root boundary is

$$p(e^{j\omega T}) = a_0 + a_1 e^{j\omega T} + e^{j2\omega T}, \quad \omega T \in [0; \pi]$$

It is more convenient however to parameterize the unit circle in the plane  $z = \tau + j\eta$  by the real part  $\tau$  instead of the phase angle  $\omega T$ . On the unit circle  $\tau$  and  $\eta$  are connected by  $\tau^2 + \eta^2 = 1$ ,  $\tau \in [-1; 1]$ , and a polynomial with a pair of conjugate roots on the unit circle has the form

$$(z - \tau - j\eta)(z - \tau + j\eta) = z^2 - 2\tau z + 1, \quad \tau \in [-1; 1]$$

By comparison with the second order polynomial, the complex root boundary is obtained as

$$a_0 = 1, \quad a_1 = -2\tau, \quad \tau \in [-1; 1]$$

This line segment  $c$  is also shown in Fig. 10.7.

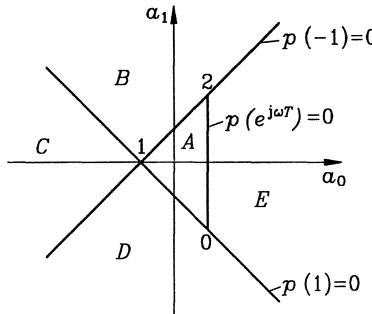


Fig. 10.7: Partition of the polynomial coefficient plane for  $n = 2$

Point 0 in Fig. 10.7 is common to  $p(e^{j\omega T}) = 0$  and  $p(1) = 0$ . A double root arises here for  $z = 1$ , i.e.

$$p_0(z) = (z - 1)^2 = z^2 - 2z + 1$$

and correspondingly

$$\begin{aligned} p_1(z) &= (z - 1)(z + 1) = z^2 - 1 \\ p_2(z) &= (z + 1)^2 = z^2 + 2z + 1 \end{aligned}$$

The two straight lines  $p(1)=0$  and  $p(-1)=0$  and the complex root boundary  $p(e^{i\omega T})=0$  partition the coefficient plane into the following regions (EV = eigenvalue, UC = unit circle).

- A: Both EVs in the UC
- B: One EV in the UC, one to the left
- C: One EV to the left, one to the right of the UC
- D: One EV in the UC, one to the right
- E: Both EVs outside the UC, either as a complex pair or both to the left, or both to the right. (Note that these cases can be transferred into each other by continuous motion of the eigenvalues across the branching points of a root locus without crossing the unit circle.)

We are mainly interested in the stability region  $A$  which is completely determined by the vertices of the triangle 012.

□

*Example 10.8.* Given is the polynomial

$$p(z) = a_0 + a_1 z + a_2 z^2 + z^3$$

The real root boundaries are

$$\begin{aligned} p(1) &= a_0 + a_1 + a_2 + 1 = 0 \\ p(-1) &= a_0 - a_1 + a_2 - 1 = 0 \end{aligned}$$

In the space of coefficients  $a_0, a_1, a_2$  the linear equation  $p(1) = 0$  describes a plane, in Fig. 10.8 it is the plane containing the points 0, 1 and 2. The plane  $p(-1) = 0$  contains the points 1, 2 and 3.

The polynomial  $p(z) = (z^2 - 2\tau z + 1)(z + r) = z^3 + (r - 2\tau)z^2 + (1 - 2r\tau)z + r$  has a conjugate pair of roots on the unit circle for an arbitrary real  $r$  and  $\tau \in [-1; 1]$ . Thus, the complex root boundary is

$$\begin{aligned} a_0 &= r \\ a_1 &= 1 - 2r\tau & \tau \in [-1; 1] \\ a_2 &= r - 2\tau \end{aligned}$$

The boundary is bilinear in  $r$  and  $\tau$ , i.e. for  $r$  constant it is a family of straight line segments beginning with  $\tau = -1$  on the edge 23 and ending with  $\tau = 1$  on the edge 01. For constant  $\tau \in [-1; 1]$  it is a family of straight lines through the edges 02 and 13. The relevant part of the stability boundary is the line segment for  $r \in [-1; 1]$ .

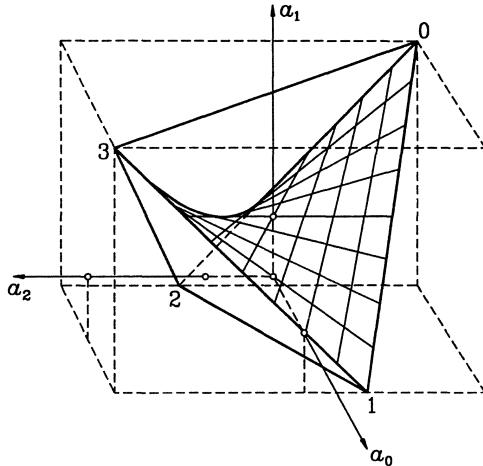


Fig. 10.8: Stability region in the polynomial coefficient space for  $n = 3$

The corners of the stability region correspond to the four polynomials with zeros in  $\{-1, 1\}$ , i.e.

$$p_0(z) = (z - 1)^3 = -1 + 3z - 3z^2 + z^3$$

$$p_1(z) = (z + 1)(z - 1)^2 = 1 - z - z^2 + z^3$$

$$p_2(z) = (z + 1)^2(z - 1) = -1 - z + z^2 + z^3$$

$$p_3(z) = (z + 1)^3 = 1 + 3z + 3z^2 + z^3$$

If the coefficients move along an edge from the corner  $i$  to the corner  $i + 1$ , then a real root migrates from  $z = 1$  to  $z = -1$  and the other roots remain at  $z = -1$  or  $z = 1$ . If the coefficients move along an edge from corner  $i$  to the corner  $i + 2$ , then a conjugate pair moves along the unit circle from  $z = 1$  to  $z = -1$  and the remaining real eigenvalue is located at  $z = 1$  or  $z = -1$ .

The complex boundary is the saddle surface visualized in Fig. 10.8. It is a ruled surface, that can easily be plotted by subdividing edge 02 and edge 13 into the same number of equal segments and connecting the corresponding points. The same procedure applies to the edges 01 and 23. The stable region is the one that contains the origin, in Fig. 10.8 below the saddle surface. Theorem 10.2 says for this example that the tetrahedron 0123 is the convex hull of the stability region.

We will refer to Fig. 10.8 several times in order to visualize various concepts for the case  $n = 3$ .  $\square$

If the polynomial coefficients are functions of uncertain parameters  $\mathbf{q}$ , then the Boundary Representation Theorem 9.2 provides a systematic way of generating the equations for mapping stability boundaries into the space of parameters  $\mathbf{q}$ .

Let us apply Theorem 9.2 to the unit circle

$$\begin{bmatrix} d_0 & d_1 & d_2 & \dots & d_n \\ 0 & d_0 & d_1 & \dots & d_{n-1} \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (10.4.15)$$

This general expression may be simplified [15] for the unit circle parameterized by the real part of  $z = \tau + j\eta$ , i.e.  $\tau \in [-1; 1]$ ,  $\tau^2 + \eta^2 = 1$ . Then

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 2\tau \\ d_{i+1} &= 2\tau d_i - d_{i-1}, \quad i = 1, 2, \dots, n-1 \end{aligned}$$

Premultiplying (10.4.15) by

$$s = \begin{bmatrix} 0 & 1 \\ -1 & 2\tau \end{bmatrix} \quad (10.4.16)$$

yields

$$\begin{bmatrix} 0 & d_0 & d_1 & d_2 & \dots & d_{n-1} \\ -d_0 & 0 & d_0 & d_1 & \dots & d_{n-2} \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (10.4.17)$$

The term  $d_n$  with the highest power of  $\tau$  has been removed. This reduction procedure may be continued  $n/2$  times for  $n$  even and  $(n+1)/2$  times for  $n$  odd. The resulting equations are for  $n$  even

$$\begin{bmatrix} -d_{\frac{n}{2}-2} & \dots & -d_0 & 0 & d_0 & d_1 & \dots & d_{\frac{n}{2}} \\ -d_{\frac{n}{2}-1} & \dots & -d_1 & -d_0 & 0 & d_0 & \dots & d_{\frac{n}{2}-1} \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (10.4.18)$$

and for  $n$  odd

$$\begin{bmatrix} -d_{\frac{n+1}{2}-2} & \dots & -d_0 & 0 & d_0 & d_1 & \dots & d_{\frac{n+1}{2}-1} \\ -d_{\frac{n+1}{2}-1} & \dots & -d_1 & -d_0 & 0 & d_0 & \dots & d_{\frac{n+1}{2}-2} \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (10.4.19)$$

In (10.4.18) and (10.4.19) the highest degree polynomials in  $\tau$  have been reduced to about half the degree arising in (10.4.15).

*Example 10.9.* Let

$$\begin{aligned} p(z, q_1, q_2) &= (-0.825 + 0.225q_1 + 0.1q_2) + (0.895 + 0.025q_1 + 0.09q_2)z + \\ &\quad + (-2.475 + 0.675q_1 + 0.3q_2)z^2 + z^3 \end{aligned}$$

The real root boundaries  $p(1, q_1, q_2) = 0$  and  $p(-1, q_1, q_2) = 0$  yield two straight lines in the  $(q_1, q_2)$ -plane of Fig. 10.9. The complex root boundary is described by

$$\begin{bmatrix} -d_0 & 0 & d_0 & d_1 \\ -d_1 & -d_0 & 0 & d_0 \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 0 & 1 & 2\tau \\ -2\tau & -1 & 0 & 1 \end{bmatrix} \mathbf{a}(q) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

For the example

$$\mathbf{a}(\mathbf{q}) = \begin{bmatrix} a_0(\mathbf{q}) \\ a_1(\mathbf{q}) \\ a_2(\mathbf{q}) \\ 1 \end{bmatrix} = \begin{bmatrix} -0.825 \\ 0.895 \\ -2.475 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.225 & 0.1 \\ 0.025 & 0.09 \\ 0.675 & 0.3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

The resulting equations

$$\begin{bmatrix} -1.65 + 2\tau \\ 0.105 + 1.65\tau \end{bmatrix} + \begin{bmatrix} 0.45 & 0.2 \\ -0.025 - 0.45\tau & -0.09 - 0.2\tau \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

can be solved for

$$\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 3.592 \\ 0.169 \end{bmatrix} + \begin{bmatrix} -5.070 \\ 1.408 \end{bmatrix} \tau + \begin{bmatrix} -11.268 \\ 25.352 \end{bmatrix} \tau^2, \quad \tau \in [-1; 1]$$

The complex boundary is also plotted in Fig. 10.9. This figure is a cross section through an affine image of the stability region of Fig. 10.8.  $\square$

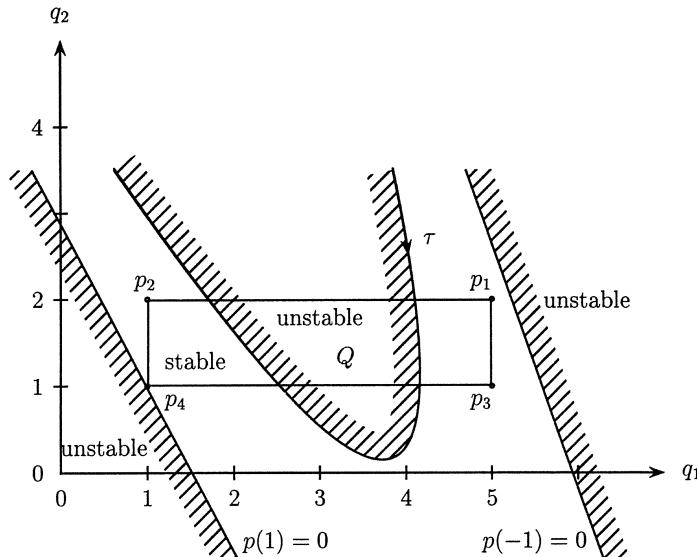


Fig. 10.9: The vertices of the operating domain  $Q$  are stable, but  $Q$  is not robustly stable

For  $n \geq 4$  we cannot visualize the stability region but some conclusions are obvious.

1. The stability region is finite and simply connected. It is contractable to the origin of the coefficient space corresponding to the polynomial  $p(z) = z^n$ .

2. The stability region is bounded by two hyperplanes corresponding to  $p(1) = 0$  and  $p(-1) = 0$  and by a complex root boundary surface.
3. The stability region for  $n \geq 3$  is nonconvex.
4. The convex hull of the stability region is a polytope whose vertices correspond to the  $n + 1$  polynomials

$$p_i(z) = (z+1)^i(z-1)^{n-i}, \quad i = 0, 1, \dots, n$$

see Theorem 10.2.

5. The edges with a vertex number difference of one or two are part of the stability boundary. Edges with a vertex number difference  $m > 2$  are outside of the stability region. A motion along these edges corresponds to a motion of a complex conjugate root pair along the root locus of  $(z-1)^m + K(z+1)^m = 0$ , which consists of circles passing through  $z = 1$  and  $z = -1$  whose tangents intersect at  $180^\circ/m$ .

*Remark 10.4.* A sufficient stability condition is due to Cohn [47]. The polynomial

$$p(z) = a_0 + a_1 z + \dots + a_{n-1} z^{n-1} + z^n$$

is stable if

$$\sum_{i=0}^{n-1} |a_i| < 1 \tag{10.4.20}$$

For  $n = 3$  this is the octahedron shown in Fig. 10.10.

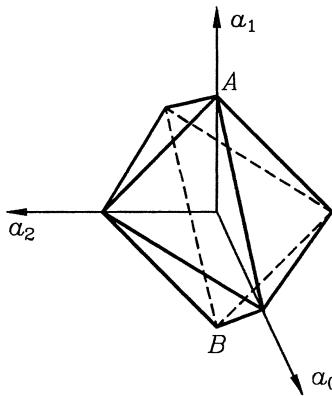


Fig. 10.10: Sufficient stability condition  $|a_0| + |a_1| + |a_2| < 1$  for  $n = 3$

The point  $A$  with coordinates  $[0 \ 1 \ 0]$  is the saddle point, see Fig. 10.8. The point  $B$  with coordinates  $[0 \ -1 \ 0]$  is the center of the edge 12 of Fig. 10.8. The illustration for  $n = 3$  shows the conservativeness of the sufficient condition.  $\square$

## 10.5 Testing Sets

### *Extreme point results*

In Section 9.7 it was shown that the unit circle is not a weak Kharitonov region , i.e. Schur stability of an interval polynomial cannot be concluded from Schur stability of all extreme points. Therefore, extreme point results hold only under very restrictive additional assumptions and special problem formulations.

Weak extreme point results (i.e. all vertices of the coefficient box  $\mathcal{A}$  must be tested) obviously exist for the convex necessary condition of Theorem 10.2 and for the sufficient condition (10.4.20). If all vertices of  $\mathcal{A}$  are stable (and therefore inside the convex hull of Theorem 10.2), then the entire  $\mathcal{A}$  is inside the convex hull, i.e. a necessary stability condition is satisfied everywhere in  $\mathcal{A}$ , and only the condition

$$\inf_{\mathbf{a} \in \mathcal{A}} \det \mathbf{S}(\mathbf{a}) > 0 \quad (10.5.1)$$

must be tested. This is however the difficult part of the problem.

Similarly, if all vertices of  $\mathcal{A}$  satisfy the sufficient stability condition (10.4.20), then the same applies to the entire  $\mathcal{A}$ . This result is however extremely conservative, as can be seen in Fig. 10.10. The figure shows a small subset of the stability region of Fig. 10.8 and now again a small subset in form of a box is taken from the octahedron.

We may as well permit the coefficients to vary in the region constrained by the  $\ell_1$ -norm (10.4.20). Results of this type have been shown by Mansour, Kraus and Anderson [109, 123] and by Tempo [163].

A weak extreme point result for the original  $\mathcal{A}$ -box was found by Hollot and Bartlett [83]. The assumption on the coefficients  $a_i \in [a_i^-; a_i^+]$  is that  $a_i^- = a_i^+$  for  $i = [\frac{n}{2}] + 1, [\frac{n}{2}] + 2, \dots, n$ , where  $[\frac{n}{2}]$  denotes the largest integer less than or equal to  $n/2$ . Then the interval polynomial family is robustly Schur stable if and only if all extreme polynomials are Schur stable.

### *Convex directions*

An interesting research direction for obtaining further extreme point results was initiated by Rantzer [144] with his notion of convex directions, see also the survey by Barmish [33].

Two examples will illustrate the idea of convex directions from the point of view of the condition (10.5.1). Consider the following question:

For which  $a_i$  is it possible that  $a_i^-$  and  $a_i^+$  yield stable polynomials but for a value in between an instability occurs? By (10.5.1) this situation can occur only if the complex root condition

$$c(a_i) = \det \mathbf{S}(a_i) > 0 \quad (10.5.2)$$

has a minimum in the interval  $[a_i^-; a_i^+]$ , i.e.

$$\frac{\partial c(a_i)}{\partial a_i} = 0 \quad \text{and} \quad \frac{\partial^2 c(a_i)}{\partial a_i^2} > 0 \quad (10.5.3)$$

*Example 10.10.*  $n = 3$

$c = 1 - a_1 + a_0a_2 - a_0^2 = 0$ . An extremum can occur only in the  $a_0$ -direction, it is a maximum because  $\partial^2 c / \partial a_0^2 = -2$ . An extreme point result holds in all directions  $a_0, a_1$ , and  $a_2$ . This result was obtained by Cieslik [46] and Kraus et al. [108].

□

*Example 10.11.*  $n = 4$

$$c = (1 - a_0)^2(1 - a_2 + a_0) + (a_3 - a_1)(a_1 - a_0a_3)$$

$a_2$  enters linearly, for  $a_1$

$$\frac{\partial^2 c(a_1)}{\partial a_1^2} = -2$$

i.e. the extremum in  $a_1$  direction is a maximum. For  $a_0$

$$\frac{\partial^2 c(a_0)}{\partial a_0^2} = 6a_0 - 2(1 + a_2)$$

A minimum occurs if  $a_0 < (a_2 + 1)/3$ . With respect to  $a_3$  we have

$$\frac{\partial^2 c(a_3)}{\partial a_3^2} = -2a_0$$

A minimum occurs if  $a_0 < 0$ . Extreme point results exist in the  $a_1$  and  $a_2$  directions. In  $a_0$  direction an extreme point result exists, if  $a_0^- > (a_2^+ + 1)/3$  and in  $a_3$  direction an extreme point result exists if  $a_0^- > 0$ . The last condition is violated for example for  $a_0 = -0.5, a_1 = 0, a_2 = 0.6$  and  $a_3 \in [-0.8; 0.8]$ . This polynomial is stable at  $a_3 = -0.8$  and at  $a_3 = 0.8$  but it is unstable at  $a_3 = 0$ .

The advantage of Kharitonov's theorem that only four polynomials have to be checked can be preserved by bilinear transformation and overbounding of the resulting polynomial in  $w$  that must be checked for Hurwitz stability, see (10.4.9).

*Example 10.12.* Consider Example 10.5 with the uncertainty intervals  $q \in [0.1; 2]$ ,  $k \in [0.1; 2]$ , like in Example 10.4. The coefficients of the transformed polynomial  $p_w(w, q, k) = b_0 + b_1w + b_2w^2$  lie in the intervals  $b_0 \in [0.0009; 3.459]$ ,  $b_1 \in [0.0003; 3.34]$ ,  $b_2 \in [0.3806; 4.49]$ . All coefficients are positive, i.e.  $p_w(w, q, k)$  is robustly Hurwitz stable and  $p(z, q, k)$  is robustly Schur stable. The example illustrates overbounding with a surprisingly good result, see also Fig. 10.6.

*Remark 10.5.* In addition the two cases of  $a_i$ -intervals of  $p(z)$  and  $b_i$ -intervals of  $p_w(w)$  there is a further alternative by the  $\delta$ -transform [128], where

$$\delta = \frac{z - 1}{T}$$

The transformed polynomial is

$$p_\delta(\delta) = p(\delta T + 1) = \sum_{i=0}^n a_i(\delta T + 1)^i = \sum_{i=0}^n c_i \delta^i$$

It may be suspected that there is a better correspondence between the continuous-time plant model and the coefficients  $c_i$  than with the  $a_i$ . The polynomial  $p(z)$  is Schur stable if and only if  $p_\delta(\delta)$  has all its roots inside a unit circle centered at  $\delta = -1/T$ . This shifted circle satisfies Rantzer's condition that both  $\Gamma$  and  $\Gamma^{-1}$  are convex, see Theorem 9.4. Thus, there holds a vertex result in the space of coefficients  $c_i$  for polynomials with complex coefficients, see also [155].  $\square$

#### *Edge result*

In situations where no vertex result holds, a useful testing set are the edges of the  $Q$ -box. The proof of the edge theorem [35] applies to the unit circle as well. Thus, the crucial condition is that the polynomial coefficients depend affinely on the uncertain parameters  $\mathbf{q}$ . For this case it suffices to check the edges of the  $Q$ -box. If this testing set is Schur stable, then the entire  $Q$ -box is Schur stable.

The following Schur stability test for a single edge was given in [13].

*Theorem 10.3. (Ackermann, Barmish)*

Let  $\mathbf{S}_b$  and  $\mathbf{S}_c$  be the stability testing matrices (10.4.5) of two polynomials

$$\begin{aligned} p_b(z) &= b_0 + b_1 z + \dots + b_n z^n \\ p_c(z) &= c_0 + c_1 z + \dots + c_n z^n \end{aligned}$$

respectively. The convex combination of the two polynomials

$$p(z, q) = (1 - q)p_b(z) + qp_c(z), \quad q \in [0; 1] \quad (10.5.4)$$

is Schur stable if and only if

1.  $p_b(z)$  is Schur stable
2.  $\text{sign } p_b(1) = \text{sign } p_c(1)$
3.  $\text{sign } p_b(-1) = \text{sign } p_c(-1)$
4. the matrix  $\mathbf{S}_b^{-1} \mathbf{S}_c$  has no nonpositive real eigenvalues.

$\square$

The proof follows closely the proof of Theorem 4.6.

*Example 10.13.* Consider Example 10.9 with the uncertainty rectangle  $Q = \{\mathbf{q} \mid q_1 \in [1; 5], q_2 \in [1; 2]\}$ . Fig. 10.9 shows the rectangle (and the stability boundaries).

The four extreme polynomials are

$$\begin{aligned} p_1(z) &= p(z, 5, 2), & p_3(z) &= p(z, 5, 1) \\ p_2(z) &= p(z, 1, 2), & p_4(z) &= p(z, 1, 1) \end{aligned}$$

and it is easily verified that their zeros are inside the unit circle. The four matrices  $\mathbf{S}_i = \mathbf{S}(p_i)$  are

$$\mathbf{S}_1 = \begin{bmatrix} 1 & 1 \\ -0.5 & -0.2 \end{bmatrix}, \quad \mathbf{S}_3 = \begin{bmatrix} 1 & 0.8 \\ -0.4 & -0.11 \end{bmatrix}$$

$$\mathbf{S}_2 = \begin{bmatrix} 1 & -0.8 \\ 0.4 & -0.1 \end{bmatrix}, \quad \mathbf{S}_4 = \begin{bmatrix} 1 & -1 \\ 0.5 & -0.01 \end{bmatrix}$$

Only two of the  $\mathbf{S}_i$  must be inverted to obtain  $\mathbf{S}_i \mathbf{S}_j^{-1}$  for the four edges.

i) Edge  $p_1 p_2$

$$\mathbf{S}_1 \mathbf{S}_2^{-1} = \frac{1}{0.22} \begin{bmatrix} -0.5 & 1.8 \\ 0.13 & -0.6 \end{bmatrix}$$

with eigenvalues  $\lambda_1 = -0.289$  and  $\lambda_2 = -4.711$ .

ii) Edge  $p_4 p_2$

$$\mathbf{S}_4 \mathbf{S}_2^{-1} = \frac{1}{0.22} \begin{bmatrix} 0.3 & -0.2 \\ -0.046 & -0.399 \end{bmatrix}$$

with eigenvalues  $\lambda_1 = 2.079$  and  $\lambda_2 = 1.098$ .

iii) Edge  $p_1 p_3$

$$\mathbf{S}_1 \mathbf{S}_3^{-1} = \frac{1}{0.21} \begin{bmatrix} 0.29 & 0.2 \\ -0.025 & 0.2 \end{bmatrix}$$

with eigenvalues  $\lambda_{1,2} = 1.167 \pm j0.260$ .

iv) Edge  $p_4 p_3$

$$\mathbf{S}_4 \mathbf{S}_3^{-1} = \frac{1}{0.21} \begin{bmatrix} -0.51 & -1.8 \\ -0.059 & -0.41 \end{bmatrix}$$

with eigenvalues  $\lambda_1 = -0.621$  and  $\lambda_2 = -3.760$ .

The edges  $p_4 p_2$  and  $p_1 p_3$  do not yield negative real eigenvalues, i.e. they are Schur stable. Edges  $p_1 p_2$  and  $p_4 p_3$ , however, have negative real eigenvalues. Thus, the  $Q$ -box is not robustly Schur stable.  $\square$

*Remark 10.6.* For small sampling intervals  $T$  the control system of Fig. 10.1 with  $c_z(z) = k$  behaves very similar to its continuous-time counterpart without sample and hold element. Therefore, an interesting question is: If the continuous system has a characteristic polynomial with affine coefficient functions such that an edge result holds, then it can be conjectured that an edge result also holds for the sampled-data system with sufficiently small  $T$ . Indeed this conjecture was proven in [89]. It is still an open question whether robust sampled-data stability for affine continuous plants can be deduced from edges.  $\square$

## 10.6 Construction of Value Sets

Presently there are no results available on tree-structures in characteristic polynomials of sampled-data systems. In [89] an example with affine continuous plant is considered. The value set  $p(e^{j\omega T}, q)$  of the sampled system is bounded not only by images of the edges of  $Q$ , but also by images of interior points of  $Q$ . Thus, the construction of value sets for sampled-data systems is a difficult problem.

## 10.7 Real Radius of Stability

The problem of finding the real radius of stability of a sampled-data control system in the space of plant parameters is extremely difficult. For controller parameters entering affinely into the coefficients the results of Chapters 7 and 9 can be used directly. Chapter 7 gives the method for determining the stability radius in case of the left half plane. Chapter 9 demonstrates the modification for general regions which can be applied for the unit circle by using the simplified boundary representations in (10.4.18) or (10.4.19).

## 10.8 Single-Loop Feedback Structures

In Section 8.3 we have discussed the use of a Nyquist value set for continuous time plants. The zero exclusion from the value set of the characteristic polynomial is thereby replaced by the exclusion of the critical point  $-1$  from the Nyquist value set. We will now translate this result to the discrete-time case.

Consider the single-loop sampled-data control system of Fig. 10.1 with open-loop frequency response

$$h_0(e^{j\omega T}, \mathbf{q}) = c_z(e^{j\omega T})h_z(e^{j\omega T}, \mathbf{q}) \quad (10.8.1)$$

The discrete-time frequency response has the following properties

1. it is periodic in  $\omega T$  with period  $2\pi$
2. it is symmetric with respect to the real axis, i.e.

$$\begin{aligned} \operatorname{Re} h_0(e^{j\omega T}, \mathbf{q}) &= \operatorname{Re} h_0(e^{-j\omega T}, \mathbf{q}) \\ \operatorname{Im} h_0(e^{j\omega T}, \mathbf{q}) &= -\operatorname{Im} h_0(e^{-j\omega T}, \mathbf{q}) \end{aligned} \quad (10.8.2)$$

3. for  $\omega T = \pi$

$$h_0(e^{j2\pi}, \mathbf{q}) = h_0(-1, \mathbf{q}) \quad (10.8.3)$$

is real, i.e.

$$h_0(-1, \mathbf{q}) = -1 \quad (10.8.4)$$

yields the real root boundary for  $z = -1$ .

- 4.

$$h_z(1, \mathbf{q}) = g_s(0, \mathbf{q}) \quad (10.8.5)$$

(see [14]). Thus,

$$h_0(1, \mathbf{q}) = c_z(1)h_z(1, \mathbf{q}) = c_z(1)g_s(0, \mathbf{q}) \quad (10.8.6)$$

From 1. and 2. follows that it suffices to sweep  $\omega T$  from zero to  $\pi$ .

The four conditions of Theorem 10.1 have their counterparts in terms of the open-loop frequency response as summarized in

*Theorem 10.4.*

A single-loop sampled-data feedback system with open-loop discrete frequency response  $h_0(e^{j\omega T}, \mathbf{q})$  is Schur stable for all  $\mathbf{q} \in Q$  if and only if

1. there exists a  $\mathbf{q}^0 \in Q$  such that the loop is Schur stable,
2.  $h_0(1, \mathbf{q}) \neq -1$  for all  $\mathbf{q} \in Q$
3.  $h_0(-1, \mathbf{q}) \neq -1$  for all  $\mathbf{q} \in Q$
4.  $h_0(e^{j\omega T}, \mathbf{q}) \neq -1$  for all  $\mathbf{q} \in Q$  and all  $\omega \in [0 ; \pi]$

□

Condition 1 is easily tested for an arbitrarily picked  $\mathbf{q}^0 \in Q$ . Condition 2 is relatively simple; in view of (10.8.5)  $\mathbf{q}$  does not enter exponentially. Conditions 3 and 4 however contain exponential terms in  $\mathbf{q}$  and are therefore difficult to test unless we are willing to grid  $Q$ .

An alternate formulation of conditions 3 and 4 is given by the Poisson sum form of the  $z$ -transfer function, see (10.1.9). For  $z = -1$ , i.e.  $\omega T = \pi$ , this sum may be simplified as follows

$$\begin{aligned}
h_z(-1, \mathbf{q}) &= 2 \sum_{m=-\infty}^{\infty} \frac{g_s[j(1+2m)\pi/T, \mathbf{q}]}{j(1+2m)\pi} \\
&= 2 \sum_{m=0}^{\infty} \frac{g_s[j(1+2m)\pi/T, \mathbf{q}]}{j(1+2m)\pi} + 2 \sum_{i=0}^{\infty} \frac{g_s[-j(1+2i)\pi/T, \mathbf{q}]}{-j(1+2i)\pi} \\
&= 2 \sum_{m=0}^{\infty} \left( \frac{g_s[j(1+2m)\pi/T, \mathbf{q}]}{j(1+2m)\pi} + \frac{g_s[-j(1+2m)\pi/T, \mathbf{q}]}{-j(1+2m)\pi} \right) \\
&= 4 \sum_{m=0}^{\infty} \operatorname{Re} \frac{g_s[j(1+2m)\pi/T, \mathbf{q}]}{j(1+2m)\pi}
\end{aligned} \tag{10.8.7}$$

Thus, Theorem 10.4 may be reformulated in terms of the continuous-time transfer function  $g_s(s, \mathbf{q})$ .

*Theorem 10.5.*

The single-loop sampled-data feedback system with continuous plant  $g_s(s, \mathbf{q})$ , sampler, hold and discrete compensator  $c_z(z)$  (see Fig. 10.1) is stable for all  $\mathbf{q} \in Q$  if and only if

1. there exists a  $\mathbf{q}^0 \in Q$  such that the loop is Schur stable
  2.  $c_z(1)g_s(0, \mathbf{q}) \neq -1$  for all  $\mathbf{q} \in Q$
  3.  $c_z(-1) 4 \sum_{m=0}^{\infty} \operatorname{Re} \frac{g_s[j(1+2m)\pi/T, \mathbf{q}]}{j(1+2m)\pi} \neq -1$  for all  $\mathbf{q} \in Q$  (10.8.8)
- and
4. for all  $\mathbf{q} \in Q$  and all  $\omega \in [0; \pi]$

$$c_z(e^{j\omega T})(1 - e^{j\omega T}) \sum_{m=-\infty}^{\infty} \frac{g_s(s + jm2\pi/T, \mathbf{q})}{sT + jm2\pi} \neq -1 \tag{10.8.9}$$

□

The advantage of the formulation of Theorem 10.5 is that for a proper plant  $g_s(s, \mathbf{q})$  the infinite series can be calculated with any desired accuracy by a finite number of terms and  $\mathbf{q}$  enters into each term only with the same complexity as in the continuous-time case.

Many plants  $g_s(s)$  have a low pass characteristic with relative degree larger than one. The convergence of the series (10.8.8) and 10.8.9 is enhanced by the  $1/s$  term from the hold element. In this case the sum converges rapidly and the dominant effect of sampling is described by the term  $m = 0$ , i.e. for the

*real root boundary* at  $z = -1$

$$h_z(-1, \mathbf{q}) \approx 4 \operatorname{Re} \frac{g_s(j\pi/T, \mathbf{q})}{j\pi} \tag{10.8.10}$$

*complex root boundary* at  $z = e^{j\omega T}$

$$h_z(e^{j\omega T}, \mathbf{q}) \approx \frac{1 - e^{-j\omega T}}{j\omega T} g_s(j\omega, \mathbf{q}) \quad (10.8.11)$$

Note that the factor  $(1 - e^{-j\omega T})/j\omega T$  is the frequency response of the hold element divided by  $T$ . It may also be written as

$$\begin{aligned} \frac{1 - e^{-j\omega T}}{j\omega T} &= \frac{e^{j\omega T/2} - e^{-j\omega T/2}}{j\omega T} \cdot e^{-j\omega T/2} \\ &= \frac{\sin \omega T/2}{\omega T/2} \cdot e^{-j\omega T/2} \end{aligned} \quad (10.8.12)$$

For the calculation of the complex root boundary the essential effect comes from the term  $e^{-j\omega T/2}$ , i.e. a phase lag equal to that of a dead time of half a sampling interval. The gain factor is one for  $\omega T = 0$  and decreases with increasing  $\omega T$ .

The approximation (10.8.11) is very good for small sampling intervals  $T$ , i.e. large frequency intervals  $2\pi/T$  at which values of the frequency response  $g_s(j\omega, \mathbf{q})/j\omega$  are taken in the sum 10.8.9. With increasing  $T$  the quality of approximation deteriorates. Then however at some sampling interval the type of oscillation illustrated by Fig. 10.3 takes over which is indicated by condition (10.8.10). For such sampling intervals the approximation is good again, because it includes a second term of 10.8.9.

*Example 10.14.* Consider Example 10.1 with  $T = 1$  again. The approximate boundary equations are for  $z = -1$

$$4 \operatorname{Re} \frac{k}{(j\pi)^2(1 + j\pi/q)} = -1$$

and for  $z = e^{j\omega T}$  from (10.1.12)

$$\frac{-(1 - e^{-j\omega})k}{(\omega + m2\pi)^2[1 + j(\omega + m2\pi)/q]} = -1$$

The approximate boundaries (dotted) are shown in Fig. 10.11; the exact boundaries (solid lines) are from Fig. 10.6. The approximation is good for small  $q$ . Note that a variable  $T$  would appear in a term  $qT$ , thus, small  $q$  has an effect like small  $T$ . For larger  $q$  the approximation of the complex root boundary  $k_c$  for  $z = e^{j\omega T}$  gets worse. A good approximation is then achieved for larger  $q$ , where the stability boundary  $k_{-1}$  for  $z = -1$  takes over.  $\square$

*Example 10.15.* For the automatic bus steering Examples 7.5 and 9.4 the compensator

$$c(s) = 25^3 \frac{0.15s^2 + 0.7s + 0.6}{(s^2 + 25s + 25^2)(s + 25)}$$

was discretized by the Tustin approximation (10.0.1) with a sampling interval  $T = 10$  [ms] yielding

$$c_z(z) = 9.3464 \frac{(z - 0.988986)(z - 0.965028)(z + 1)}{(z - 0.7788)(z^2 - 1.7238z + 0.7788)}$$

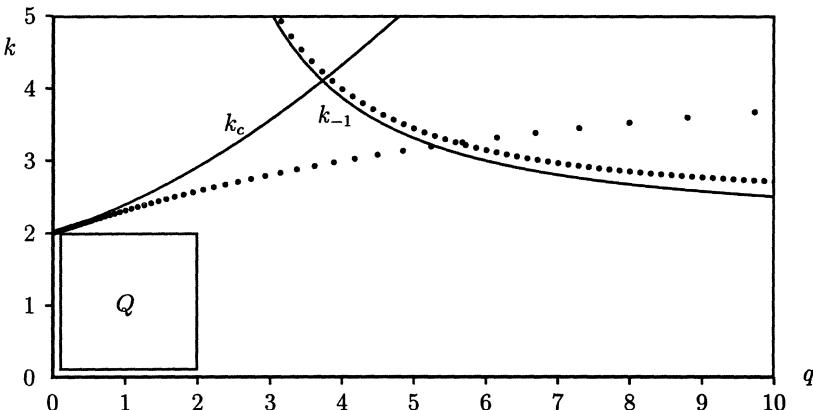


Fig. 10.11: Approximation of Fig. 10.6 by truncated Poisson series

The resulting stability boundaries are shown in Fig. 10.12.

The solid line is the  $\Gamma$ -stability boundary  $\partial\Gamma = \{\sigma + j\omega \mid \omega^2 = 25\sigma^2 - 49/16, \sigma \leq -0.25\}$  for the continuous system. The dashed line is the approximated  $\Gamma$ -stability boundary for the system with discretized compensator  $c_z(z)$ . The accuracy of the approximation was tested by finding the roots of some polynomials on both sides nearby the approximate boundary. The agreement was excellent, such that the dashed line is also the exact  $\Gamma$ -stability boundary for the discrete-time system. This effect is not surprising because the continuous-time plant has a relative degree three, i.e. the Poisson series converges like  $1/\omega^4$  and the sampling interval of  $T = 10$  [ms] used for the controller implementation is small for this plant. In this example the  $\Gamma$ -stable region is not much reduced by the compensator discretization.  $\square$

## 10.9 Circle Stability

In Section 10.3 we have discussed eigenvalue specifications for sampled-data control systems. A conclusion was that circles centered on the real axis are useful regions of  $\Gamma$ -stability for sampled-data systems. In this section we show how the results obtained for Schur stability can be generalized to  $\Gamma$ -stability for a circle with real center  $\tau_0$  and radius  $r$ , see Fig. 10.13.

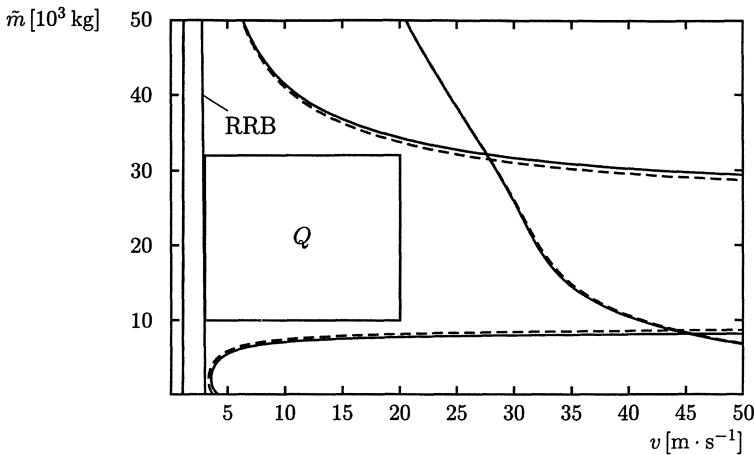


Fig. 10.12: Continuous time (solid) and approximated discrete time (dashed)  $\Gamma$ -stability boundaries in the  $(v, \tilde{m})$ -plane

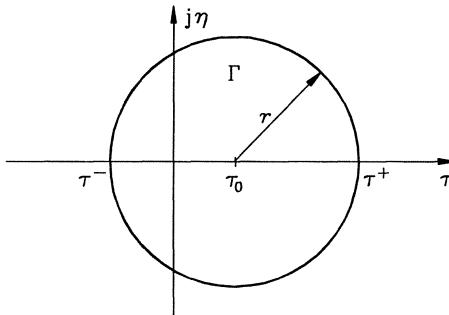


Fig. 10.13: Circle stability

*Definition 10.1.* We call a polynomial *circle stable* if all its roots are located in a circle with given real center and radius.  $\square$

Circle stability is specified by two parameters, e.g. center  $\tau_0$  and radius  $r$  or alternatively

$$\begin{aligned}\tau^- &:= \tau_0 - r \\ \tau^+ &:= \tau_0 + r\end{aligned}\tag{10.9.1}$$

*Remark 10.7.* For  $\tau^- > 0$  (or  $\tau^+ < 0$ , which is not of interest for sampled-data systems) both  $\Gamma$  and  $\Gamma^{-1}$  are convex and  $\Gamma$  is a Kharitonov region in the sense of Def. 9.1.  $\square$

The circle of Fig. 10.13 can be mapped onto the unit circle by an affine transformation

$$\tilde{z} := (z - \tau_0)/r, \quad z = r\tilde{z} + \tau_0\tag{10.9.2}$$

Thus, a polynomial  $p(z) = a_0 + a_1z + \dots + a_nz^n$  is circle stable if and only if the polynomial

$$\begin{aligned}\tilde{p}(\tilde{z}) &= p(r\tilde{z} + \tau_0) \\ &= \sum_{i=0}^n a_i(r\tilde{z} + \tau_0)^i \\ &= \sum_{i=0}^n \tilde{a}_i \tilde{z}^i\end{aligned}\tag{10.9.3}$$

is Schur stable.

The affine transformation (10.9.2) can be directly combined with the bilinear transformation (10.4.7)

$$w = \frac{\tilde{z} - 1}{\tilde{z} + 1} = \frac{z - \tau_0 - r}{z - \tau_0 + r}, \quad z = r \frac{1 + w}{1 - w} + \tau_0\tag{10.9.4}$$

or, written in terms of the parameters  $\tau^-, \tau^+$

$$w = \frac{z - \tau^+}{z - \tau^-}, \quad z = \frac{\tau^+ - \tau^- w}{1 - w}\tag{10.9.5}$$

A polynomial  $p(z) = a_0 + a_1z + \dots + a_nz^n$  is circle stable if and only if the polynomial

$$\begin{aligned}p_w(w) &= (1 - w)^n p\left(\frac{\tau^+ - \tau^- w}{1 - w}\right) \\ &= \sum_{i=0}^n a_i (\tau^+ - \tau^- w)^i (1 - w)^{n-i} \\ &= \sum_{i=0}^n b_i w^i\end{aligned}\tag{10.9.6}$$

is Hurwitz stable.

Again, the necessary stability conditions  $b_i > 0$ ,  $i = 0, 1, \dots, n$  describe the convex hull of the stability region in the space of coefficients  $a_i$ .

*Example 10.16.* The zeros of the polynomial  $p(z) = a_0 + a_1z + a_2z^2 + z^3$  should lie inside the circle with center  $\tau_0 = 0.4$  and radius  $r = 0.4$ , i.e.  $\tau^- = 0$ ,  $\tau^+ = 0.8$ .

$$p_w(w) = \sum_{i=0}^3 a_i 0.8^i (1 - w)^{3-i} = \sum_{i=0}^3 b_i w^i\tag{10.9.7}$$

$$\begin{aligned}b_0 &= p(0.8) = a_0 + 0.8a_1 + 0.64a_2 + 0.512 \\ b_1 &= -(3a_0 + 1.6a_1 + 0.64a_2) \\ b_2 &= 3a_0 + 0.8a_1 \\ b_3 &= -a_0\end{aligned}$$

The convex hull of the stability region is described by the inequalities  $b_i > 0$ , the critical conditions contributing to the boundary of the stable region are  $b_0 > 0, b_3 > 0$  and  $b_1 b_2 - b_0 b_3 > 0$ .  $\square$

The modified form of Theorem 10.2 can be shown by barycentric coordinates [5]. The result is

*Theorem 10.6.*

Consider the circle stability region of

$$p(z) = a_0 + a_1 z + \dots + a_{n-1} z^{n-1} + z^n \quad (10.9.8)$$

in the space of coefficients  $a_0, a_1, \dots, a_n$ . Its convex hull is a polytope with  $n+1$  vertices corresponding to the polynomials

$$p_i(z) = (z - \tau^-)^i (z - \tau^+)^{n-i}, \quad i = 0, 1, 2, \dots, n \quad (10.9.9)$$

$\square$

A boundary representation (9.2.15) for frequency domain approaches is obtained by substitution of

$$\eta^2 + (\tau - \tau_0)^2 = r^2, \quad \text{i.e. } \eta^2 + \tau^2 = r^2 + 2\tau_0\tau - \tau_0^2 \quad (10.9.10)$$

into (9.2.16)

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 2\tau \\ d_{i+1} &= 2\tau d_i - (r^2 + 2\tau_0\tau - \tau_0^2) d_{i-1} \end{aligned} \quad (10.9.11)$$

and (9.2.15) yields the boundary description.

*Example 10.17.* Consider again Example 10.16 with  $\tau_0 = 0.4, r = 0.4$

$$\begin{aligned} d_0 &= 1 \\ d_1 &= 2\tau \\ d_2 &= 4\tau^2 - 0.8\tau \\ d_3 &= 4\tau^3 - 3.2\tau^2 \end{aligned}$$

The coefficients  $a_i$  of a third degree polynomial with roots on the circle of radius  $r = 0.4$ , centered at  $\tau_0 = 0.4$  satisfy

$$\begin{bmatrix} d_0 & d_1 & d_2 & d_3 \\ 0 & d_0 & d_1 & d_2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2\tau & 4\tau^2 - 0.8\tau & 4\tau^3 - 3.2\tau^2 \\ 0 & 1 & 2\tau & 4\tau^2 - 0.8\tau \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \tau \in [0; 0.8]$$

The cubic term in  $\tau$  may be eliminated by a left multiplication of the above equations by

$$\begin{bmatrix} 1 & -\tau \\ 0 & 1 \end{bmatrix}$$

yielding

$$\begin{bmatrix} 1 & \tau & 2\tau^2 - 0.8\tau & -2.4\tau^2 \\ 0 & 1 & 2\tau & 4\tau^2 - 0.8\tau \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

□

## 10.10 Exercises

10.1. Consider a discrete-time plant

$$\mathbf{x}(i+1) = \begin{bmatrix} -3 & 2 \\ -7+q & 4 \end{bmatrix} \mathbf{x}(i) + \begin{bmatrix} 1 \\ 2+q \end{bmatrix} u(i)$$

with state feedback

$$u(i) = -[k_1 \ k_2] \mathbf{x}(i)$$

- a) Let  $q = 0$ . Find the deadbeat solution and plot its Schur stable neighborhood in the  $(k_1, k_2)$ -plane.
- b) For what maximal value of  $q = q^+$  is it possible to simultaneously stabilize  $q = 0$  and  $q = q^+$  and what is the state-feedback vector required for this?

10.2. Consider the polynomial

$$p(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4 + z^5$$

What are the necessary stability conditions (resulting from the convex hull of the stable region).

10.3. The Schur stability region for a polynomial  $p(z) = a_0 + a_1 z + a_2 z^2 + z^3$  is shown in Fig. 10.8. Plot a cross section for  $a_1 = 1.5$ .

10.4. Given

$$\begin{aligned}\mathbf{x}(i+1) &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0.6 & -2 & 0.8 & 0.2 \end{bmatrix} \mathbf{x}(i) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(i) \\ \mathbf{y}(i) &= \begin{bmatrix} -0.4 & -2.2 & 0 & 0 \\ 0.6 & -1 & 1 & 0 \end{bmatrix} \mathbf{x}(i)\end{aligned}$$

Find the set of all Schur stabilizing output feedback vectors

$$u(i) = -[k_a \ k_b] \mathbf{y}(i)$$

and its convex hull.

10.5. The polynomial

$$p(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + z^4$$

shall have all its roots in a circle with the center  $z = 0.5$  and the radius 0.5.

Formulate the algebraic conditions and test

$$p(z) = 0.25 - 1.2z + 2.3z^2 + 2.3z^3 + z^4$$

10.6. Consider the discrete-time plant

$$\mathbf{x}(i+1) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a & b & 1 \end{bmatrix} \mathbf{x}(i) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(i)$$

with state feedback

$$u(i) = [k_1 \ k_2 \ k_3] \mathbf{x}(i)$$

Do a deadbeat design for nominal parameter values  $a = 2$ ,  $b = 1$ . For fixed  $k_1, k_2, k_3$ , plot the stable neighborhood of the nominal parameter values in the  $(a, b)$ -plane.

10.7. Let

$$\begin{aligned}p(z, q_1, q_2) &= (q_1 + q_2) + (2q_1 + q_2)z + (3q_1 + q_2)z^2 \\ &\quad + (4q_1 + q_2)z^3 + z^4\end{aligned}$$

Plot the necessary stability conditions (resulting from the convex hull of the stability region) in the  $(q_1, q_2)$ -plane.

What further steps are necessary to decide about robust stability?

## **Part IV**

### **Some Design Tools for Robust Control Systems**

# 11 Parameter Space Design

## 11.1 Introduction to Design by Simultaneous Gamma-Stabilization

With Chapter 11 we are entering into Part IV of this book, the part that deals with design. For an uncertain closed-loop characteristic polynomial  $p(s, \mathbf{q}, \mathbf{k})$  we want to find a  $\mathbf{k} = \mathbf{k}^0$  such that the polynomial  $p(s, \mathbf{q}, \mathbf{k}^0)$  is  $\Gamma$ -stable for all  $\mathbf{q} \in Q$ . Generic situations are:

1. There exists such a  $\mathbf{k}^0$  and also a neighborhood  $K_\Gamma$  of  $\mathbf{k}^0$  has the same property, see Fig. 3.10. We want to pick the “best”  $\mathbf{k} \in K_\Gamma$  in consideration of further design requirements.
2. There does not exist such a  $\mathbf{k}^0$ , see Fig. 3.11 and Example 2.1.

Unfortunately, there does not yet exist a criterion that allows to test a given plant family and controller structure and to decide if it belongs to case 1 or 2. Even the simpler problem of simultaneously stabilizing a finite polynomial family  $\{p(s, \mathbf{q}^{(i)}, \mathbf{k}) \mid i = 1, 2, \dots, N\}$  was solved only for  $N = 2$  [174]. An obvious necessary condition is that all unstable eigenvalues must be controllable and observable, but this condition is not sufficient (as in the case of a nominal plant). Many authors have derived sufficient conditions using various restrictive assumptions or approximations. However, no general method for deciding the above 1 or 2 question has been found so far. We will not review this literature here and refer the interested reader to publications by Desoer *et al.* [54], Saeks and Murray [147], Vidyasagar *et al.* [173, 174], Ghosh *et al.* [77, 74–76], Kimura [107], Doyle *et al.* [58], Khargonekar *et al.* [104, 103], Wei *et al.* [34, 165, 178, 177], Wu *et al.* [179], Kwakernaak [113], Debowski and Kurylowicz [52], Howitt and Luus [88], Youla *et al.* [180], Emre [60], Djaferis [56], Leitmann [115], and Poolla *et al.* [139].

For the design of robust control systems there are two principally different categories of approaches. In the first category we strive for a one-shot procedure leading to a controller that is guaranteed to robustly stabilize the plant family such that no further robustness analysis is required. Typically, a scalar measure for robust performance must be formulated for this category. An example is Doyle’s  $\mu$ -optimization [25, 159].

In the second category there are tools that help the design engineer to understand the design conflicts between different operating conditions and different specifications and support an interactive design process involving trial and error procedures and robustness analysis of the resulting closed-loop control system. An example is the Quantitative Feedback Theory (QFT) developed by Horowitz [87].

From an engineering science point of view only the first category looks appealing. However, as long as the fundamental existence questions are unsolved, all design procedures have to be conservative. This often leads to controllers of unnecessary high order and with insufficient flexibility for tradeoffs with other design requirements. If the evaluation of the resulting closed-loop system is done by other criteria than those used directly in the design (for example by step responses or simulations with a nonlinear plant model), then design becomes a trial and error procedure anyway.

In Part IV of this book we shift the emphasis back from engineering science to engineering art. We develop some tools for the second category of design approaches. Chapter 11 deals with the parameter plane design [3, 5] and Chapter 12 treats the design by optimizing a vector performance index [110, 112]. The application of these design tools will be shown by the practical examples of Chapter 1.

First, we discuss the role of trial and error in design. Assume that a controller was designed only for a nominal operating point  $\mathbf{q}^0$  of the plant. If the control system is now analyzed for robust  $\Gamma$ -stability in a  $Q$ -box centered at  $\mathbf{q}^0$ , then it is very unlikely that we are so lucky to get a positive answer. The solution also does not give a hint in which direction to go in order to improve the covering of  $Q$  by the  $\Gamma$ -stability region. This would be “blind” trial and error. A much more directed approach is to find a controller that simultaneously  $\Gamma$ -stabilizes the vertices  $\mathbf{q}_v^{(i)}$  of the  $Q$ -box. The result of a  $\Gamma$ -stability robustness analysis is shown schematically in Fig. 3.13. It gives a clear indication how the  $Q$ -box may be better represented in the next design iteration by including the operating point E (see Fig. 3.13). For a larger number of uncertain parameters the radius of stability (see Chapter 7) from the center of the  $Q$ -box is determined. If it includes the vertices of the  $Q$ -box, then robust stability is guaranteed for all  $\mathbf{q} \in Q$ . Otherwise, the worst case operating condition in the  $Q$ -box is found and used in the next design iteration to improve the representation of the  $Q$ -box by a finite plant family.

For interval plants with a first order compensator it was shown in Section 8.1 that simultaneous stabilization of the vertices of the  $Q$ -box implies robust stabilization for all  $\mathbf{q} \in Q$ . In this case the robustness analysis step is not needed. There is good reason to believe that such an extreme point result holds for many plant families. In fact, in all our practical applications we did not find a single case that required further representatives in addition to the vertices  $\mathbf{q}_v^{(i)}$ , we had “trial and success”. Based on this experience we are convinced that the working hypothesis “vertex  $\Gamma$ -stabilization suffices” is a very helpful and practical assumption. We remind you here of the third basic rule of robust control:

*Be a pessimist in analysis, then you can afford to be an optimist in design.*

With the above justification we now have to solve the problem of simultaneous  $\Gamma$ -stabilization for the vertices of the  $Q$ -box, also called the *multi-model problem formulation*.

## 11.2 Pole Region Assignment

In this section a state-feedback controller

$$u = -\mathbf{k}^T \mathbf{x} \quad (11.2.1)$$

is assumed. As discussed in the previous section it should simultaneously  $\Gamma$ -stabilize the vertices of the  $Q$ -box. As a first step we  $\Gamma$ -stabilize only one operating point  $\mathbf{q}^{(1)}$  (e.g. one of the vertices or the center point of the  $Q$ -box). An obvious possibility is to specify all  $n$  closed-loop eigenvalues in  $\Gamma$  and to place the poles there. For a single-input system, pole placement yields a unique  $\mathbf{k}$ , i.e. there remains no flexibility for accommodating other operating conditions.

The discussion in Chapter 3 on eigenvalue specifications shows that good performance does not require an exactly specified location of all eigenvalues. It rather suffices to place the eigenvalues of the controlled system in the desired pole region  $\Gamma$  for all  $\mathbf{q} \in Q$ . We want to find the set  $K_{\Gamma}^{(1)}$  of all state-feedback vectors  $\mathbf{k}$  which  $\Gamma$ -stabilize the plant for the fixed operating condition  $\mathbf{q}^{(1)}$ .

In the preceding chapters robust stability tests for an uncertain polynomial  $p(s, \mathbf{q})$  were developed. Now  $\mathbf{k}$  plays the role of the uncertain parameters: we want to find the set of all  $\mathbf{k}$ , for which the polynomial  $p(s, \mathbf{q}^{(1)}, \mathbf{k})$  is  $\Gamma$ -stable. These  $\Gamma$ -stabilizing  $\mathbf{k}$  form the set  $K_{\Gamma}^{(1)}$ . It follows from the Boundary Crossing Theorem introduced in Chapter 4 that for a point  $\mathbf{k}_B$  on the boundary  $\partial K_{\Gamma}^{(1)}$  of the set of  $\Gamma$ -stabilizing controller parameters  $K_{\Gamma}^{(1)}$ , at least one root of  $p(s, \mathbf{q}^{(1)}, \mathbf{k}_B)$  lies exactly on the boundary  $\partial\Gamma$  of the eigenvalue region. Thus, it suffices to map the boundary of the eigenvalue region  $\Gamma$  into  $\mathbf{k}$ -space. The resulting image will divide the  $\mathbf{k}$ -space into a finite number of sets. By checking an arbitrary point of each set, the stabilizing set  $K_{\Gamma}$  can be determined. Fig. 11.1 shows a schematic representation of this mapping. Here, the image of  $\partial\Gamma$  results in a curve which divides the  $\mathbf{k}$ -space into three subsets. From each set a point has to be selected (for example  $\mathbf{k}_1$ ,  $\mathbf{k}_2$ , and  $\mathbf{k}_3$ ) and the  $\Gamma$ -stability of the characteristic polynomials  $p(s, \mathbf{q}^0, \mathbf{k}_i)$ ,  $i = 1, 2, 3$  has to be checked. For the example  $p(s, \mathbf{q}^{(1)}, \mathbf{k}_1)$  turned out to be  $\Gamma$ -stable, thus, region 1 represents the desired set  $K_{\Gamma}^{(1)}$ .

For single-input systems the characteristic polynomial depends in an affine manner on the controller parameters  $\mathbf{k}$ . A graphical display of  $K_{\Gamma}^{(1)}$  is easy if we fix  $n - 2$  gains of the state feedback vector  $\mathbf{k}$ , say  $k_3, k_4, \dots, k_n$ , and plot  $\partial K_{\Gamma}^{(1)}$  in the  $(k_1, k_2)$ -plane. The boundary of the eigenvalue region  $\Gamma$  is mapped into the two-dimensional  $\mathbf{k}$ -space. The boundary  $\partial\Gamma$  is represented by (9.4.2), where  $\mathbf{a} = \mathbf{a}(k_1, k_2)$  is the coefficient vector of

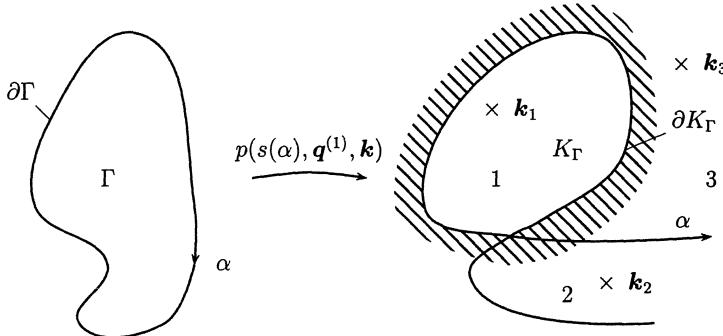


Fig. 11.1: Mapping of the boundary of the pole region  $\Gamma$  into the  $\mathbf{k}$ -space via a nominal characteristic polynomial  $p(s, q^{(1)}, \mathbf{k})$ . The boundary  $\partial\Gamma$  is parametrized by  $\alpha$ .

the closed-loop characteristic polynomial. The two rows of (9.4.2) are

$$\begin{aligned} \operatorname{Re}(p(\alpha)) &= b_0(\alpha) + b_1(\alpha) \cdot k_1 + b_2(\alpha) \cdot k_2 = 0 \\ \operatorname{Im}(p(\alpha)) &= c_0(\alpha) + c_1(\alpha) \cdot k_1 + c_2(\alpha) \cdot k_2 = 0 \end{aligned}$$

which is explicitly solved for  $\mathbf{k}(\alpha)$ :

$$\mathbf{k}(\alpha) = \begin{bmatrix} k_1(\alpha) \\ k_2(\alpha) \end{bmatrix} = - \begin{bmatrix} b_1(\alpha) & b_2(\alpha) \\ c_1(\alpha) & c_2(\alpha) \end{bmatrix}^{-1} \cdot \begin{bmatrix} b_0(\alpha) \\ c_0(\alpha) \end{bmatrix} \quad (11.2.2)$$

The boundary parameter  $\alpha$  parametrizes the boundary of  $\Gamma$  for  $\alpha \in [\alpha^-; \alpha^+]$ . The boundaries in the  $(k_1, k_2)$ -plane are computed and displayed on a graphics terminal. If the matrix in (11.2.2) cannot be inverted, then the boundaries in the  $(k_1, k_2)$ -plane tend towards infinity or an isolated frequency occurs.

Pole region assignment is illustrated by the following example.

*Example 11.1.*  $\Gamma$ -stabilize the crane as given in (1.1.6) with  $\ell = 12$  [m],  $m_L = 1500$  [kg],  $m_C = 1000$  [kg], and  $g = 10$  [m · s<sup>-2</sup>], by state feedback

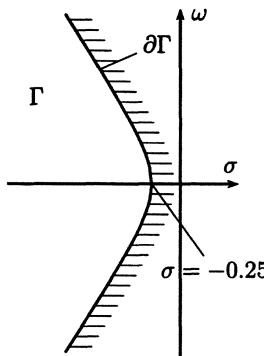
$$u = - \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}^T \mathbf{x}$$

The rope angle rate  $x_4$  is hard to measure. Omit this sensor and set  $k_4 = 0$ . For a typical transition from a position  $x_1(0)$  to position 0 the initial force is proportional to  $k_1 \cdot x_1(0)$ . The controller parameter  $k_1$  is determined from the maximal possible transition  $\hat{x}_1(0)$  in order to limit the initial force, e.g.  $k_1 = 500$ . The remaining problem is to determine the set of  $\Gamma$ -stabilizing controllers in the  $(k_2, k_3)$ -plane.

The hyperbola

$$\omega^2 = 4\sigma^2 - 0.25 \quad (11.2.3)$$

was chosen as boundary of the eigenvalue region  $\Gamma$  to guarantee sufficient damping and short settling time. It is displayed in Fig. 11.2. Then the mapping equation (9.4.2) is

Fig. 11.2: Eigenvalue region  $\Gamma$  for the crane

with  $\alpha = \sigma$

$$\begin{bmatrix} 1 & 2\alpha & -\alpha^2 + 0.25 & -12\alpha^3 + \alpha & 19 - \alpha^4 + 0.5\alpha^2 + 0.0625 \\ 0 & 1 & 2\alpha & -\alpha^2 + 0.25 & -12\alpha^3 + \alpha \end{bmatrix} \mathbf{a}(k_1, k_2) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (11.2.4)$$

The characteristic polynomial of the crane with the given parameter values for  $m_L$ ,  $m_C$ ,  $\ell$ ,  $g$ ,  $k_1$ , and  $k_4$ , follows from (2.2.16) as

$$p(s, k_2, k_3) = 5000 + 10k_2s + (31000 - k_3)s^2 + 12k_2s^3 + 12000s^4 = 0$$

i.e.  $\mathbf{a}(k_1, k_2) = [5000 \ 10k_2 \ 31000 - k_3 \ 12k_2 \ 12000]^T$  must be substituted in (11.2.4):

$$\begin{bmatrix} 13500 - 25000\alpha^2 - 228000\alpha^4 & + & \alpha(32 - 144\alpha^2)k_2 & + & (-0.25 + \alpha^2)k_3 \\ 74000\alpha - 144000\alpha^3 & + & (13 - 12\alpha^2)k_2 & - & 2\alpha k_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

These two linear equations are solved for  $k_2$  and  $k_3$ . The complex root boundary is then represented by

$$k_2(\alpha) = 2000 \frac{17\alpha + 120\alpha^3 - 1200\alpha^5}{13 - 320\alpha^2 + 1200\alpha^4}$$

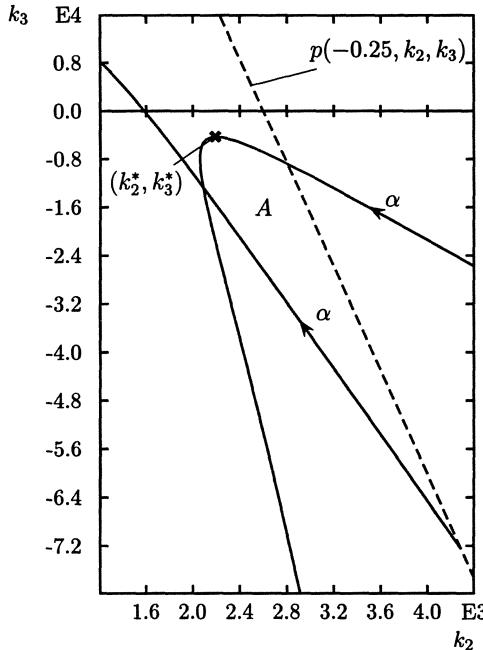
and

$$k_3(\alpha) = 2000 \frac{351 - 5710\alpha^2 + 25200\alpha^4 - 36000\alpha^6}{13 - 320\alpha^2 + 1200\alpha^4}$$

with  $\alpha \in (-\infty; -0.25]$ . An asymptote occurs for  $13 - 320\alpha^2 + 1200\alpha^4 = 0$ , i.e.  $\alpha = -\frac{1}{2}\sqrt{13/15}$  in the specified  $\alpha$ -interval. There is also a real root boundary at  $s = -0.25$ . It is

$$r(k_2, k_3) = p(-0.25, k_2, k_3) = 6984.37 - 2.6875k_2 - 0.0625k_3 = 0$$

The plots are shown in Fig. 11.3. The real root boundary is plotted with dashed lines. The  $(k_2, k_3)$ -plane is divided into 6 regions. By checking stability for an arbitrary  $k_2, k_3$

Fig. 11.3: Set of  $\Gamma$ -stabilizing controller parameters

for each of these regions it turns out that region  $A$  is  $\Gamma$ -stable,  $A$  is the set of admissible controllers.

Each of the  $(k_2, k_3)$ -pairs in the region  $A$   $\Gamma$ -stabilizes the crane. An interesting choice is the minimum-norm feedback. The solution with minimal distance  $\sqrt{k_2^2 + k_3^2}$  from the origin is determined as  $k_2^* = 2191$  and  $k_3^* = -4299$ , see Fig. 11.3. The feedback law is then

$$u = - \begin{bmatrix} 500 & 2191 & -4299 & 0 \end{bmatrix} x \quad (11.2.5)$$

□

### Invariance Planes

In the previous example the four-dimensional controller coefficient space was reduced by some simple considerations to a two-dimensional space. This simplified the graphical representation of the stabilizing sets. If a reduction in the number of free controller parameters is not possible, then a cross section plane has to be selected in the state-feedback gain space. The boundaries are displayed in this two-dimensional plane. Of course it is crucial how to place this cross section, especially if the  $k$ -space is high dimensional. In [20] a systematic approach for choosing the cross section plane was introduced. It is based on the idea that at a certain design stage it is easy to decide which two eigenvalues are the “worst” ones and should be moved into the  $\Gamma$ -stable region. All other eigenvalues should be kept at their previous positions.  $\Gamma$ -stabilization

is then a sequential procedure in which only two eigenvalues are shifted in each design step.

For a given state-feedback  $\mathbf{k}_1$  from a previous design iteration we can calculate the eigenvalues and determine the two worst ones, e.g.  $p(s) = h(s) \cdot d(s)$ , where  $d(s) = d_0 + d_1 s + s^2$  contains the worst eigenvalues. A good strategy is to shift only the worst eigenvalues, while the less critical eigenvalues in  $h(s) = h_0 + h_1 s + \dots + h_{n-2} s^{n-2}$  remain at their old locations. The new characteristic polynomial is then  $h(s) \cdot t(s)$ , where  $t(s) = t_0 + t_1 s + s^2$  contains the new location of the eigenvalues originating from  $d(s)$ . A new controller  $\mathbf{k}_2$  has to be determined. In the next design step again only the two worst eigenvalues are shifted. This procedure is continued until all eigenvalues are finally contained in the desired region  $\Gamma$ .

The problem is how to change the controller parameters starting from  $\mathbf{k} = \mathbf{k}_1$  such that  $(n-2)$  eigenvalues remain at their old locations. It is solved by Ackermann's formula [2, 5, 99].

*Theorem 11.1. (Ackermann)*

For a controllable single input system  $(\mathbf{A}, \mathbf{b})$  the feedback vector

$$\mathbf{k}^T = \mathbf{e}^T p(\mathbf{A}) \quad (11.2.6)$$

with

$$\mathbf{e}^T = \begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} \mathbf{b} & \mathbf{Ab} & \mathbf{A}^2\mathbf{b} & \dots & \mathbf{A}^{n-1}\mathbf{b} \end{bmatrix}^{-1} \quad (11.2.7)$$

assigns the eigenvalues of  $\mathbf{A} - \mathbf{b}\mathbf{k}^T$  to the roots of the polynomial  $p(s)$ .

□

With the closed-loop characteristic polynomial vector  $\hat{\mathbf{a}}$  (i.e.  $p(s)/a_n = [1 \ s \ \dots \ s^{n-1} \ s^n][\hat{\mathbf{a}}^T \ 1]^T$ ) the formula (11.2.6) may be written as

$$\mathbf{k}^T = \hat{\mathbf{a}}^T \mathbf{E} \quad (11.2.8)$$

where

$$\mathbf{E} = \begin{bmatrix} \mathbf{e}^T \\ \mathbf{e}^T \mathbf{A} \\ \vdots \\ \mathbf{e}^T \mathbf{A}^n \end{bmatrix}$$

is called the *pole placement matrix*.

According to this theorem the required feedback vector for shifting two eigenvalues of a controllable system  $(\mathbf{A}, \mathbf{b})$  to their new location  $h(s) \cdot t(s)$  is

$$\mathbf{k}^T = \mathbf{e}^T \cdot h(\mathbf{A}) \cdot t(\mathbf{A}) = \mathbf{e}_h^T \cdot t(\mathbf{A}) \quad (11.2.9)$$

where  $\mathbf{e}_h^T = \mathbf{e}^T h(\mathbf{A})$  is known.

$$\mathbf{k}^T = \mathbf{e}_h^T \cdot (t_0 \cdot \mathbf{I} + t_1 \cdot \mathbf{A} + \mathbf{A}^2) = \begin{bmatrix} t_0 & t_1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{e}_h^T \\ \mathbf{e}_h^T \mathbf{A} \\ \mathbf{e}_h^T \mathbf{A}^2 \end{bmatrix} \quad (11.2.10)$$

For  $t(s) = d(s)$  the eigenvalues of the control system are identical to the open-loop eigenvalues. This can only be achieved for  $\mathbf{k}^T = \mathbf{0}^T$ :

$$\mathbf{0}^T = \mathbf{e}_h^T \cdot d(\mathbf{A}) \quad (11.2.11)$$

Forming the difference of (11.2.9) and (11.2.11) yields

$$\mathbf{k}^T = \mathbf{e}_h^T \cdot (t(\mathbf{A}) - d(\mathbf{A})) = \begin{bmatrix} t_0 - d_0 & t_1 - d_1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{e}_h^T \\ \mathbf{e}_h^T \mathbf{A} \\ \mathbf{e}_h^T \mathbf{A}^2 \end{bmatrix}$$

which can be reduced to

$$\mathbf{k}^T = [\kappa_a \ \kappa_b] \begin{bmatrix} \mathbf{e}_h^T \\ \mathbf{e}_h^T \mathbf{A} \end{bmatrix} \quad (11.2.12)$$

with  $\kappa_a = t_0 - d_0$ ,  $\kappa_b = t_1 - d_1$ . In the  $n$ -dimensional  $\mathbf{k}$ -space the vectors  $\mathbf{e}_h^T$  and  $\mathbf{e}_h^T \mathbf{A}$  span a two-dimensional linear subspace such that the eigenvalues of  $h(s)$  are unobservable from  $\mathbf{k}^T \mathbf{x}$ . Varying the parameters  $\kappa_a$  and  $\kappa_b$  in this *invariance plane* means moving two eigenvalues while the rest remains in their old location. This plane is used for graphical representation.

### Example 11.2. Crane

Again the crane with the data given in Example 11.1 is considered. The system is controlled by full state feedback, the desired pole region with the hyperbolic boundary (11.2.3) was shown in Fig. 11.2.

The state space representation is given by (1.1.6)

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 15 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -25/12 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 1/1000 \\ 0 \\ -1/12000 \end{bmatrix} u$$

and the open-loop characteristic polynomial is

$$p(s) = s^2 \left( \frac{25}{12} + s^2 \right)$$

The open-loop system has a double pole at the origin,  $d(s) = s^2$ , and a complex conjugate pole pair on the  $j\omega$ -axis,  $h(s) = 25/12 + s^2$ . Choose an invariance plane in which only the double pole at the origin is shifted. The conjugate complex pole

pair remains in its old location and the double pole at the origin is shifted to a new destination  $t_0 + t_1 s + s^2$ . The closed-loop polynomial  $p(s)$  is then

$$p(s) = t(s) \cdot h(s) = (t_0 + t_1 s + s^2) \left( \frac{25}{12} + s^2 \right)$$

For the given operating point, (11.2.7) yields

$$\mathbf{e}^T = [ \ 1200 \ 0 \ 14400 \ 0 \ ]$$

and according to (11.2.9),  $\mathbf{e}_h$  is calculated as

$$\begin{aligned} \mathbf{e}_h^T &= \mathbf{e}^T h(\mathbf{A}) = [ \ h_0 \ h_1 \ 1 \ ] \begin{bmatrix} \mathbf{e}^T \\ \mathbf{e}^T \mathbf{A} \\ \mathbf{e}^T \mathbf{A}^2 \end{bmatrix} \\ &= [ \ \frac{25}{12} \ 0 \ 1 \ ] \begin{bmatrix} 1200 & 0 & 14400 & 0 \\ 0 & 1200 & 0 & 14400 \\ 0 & 0 & -1200 & 0 \end{bmatrix} \\ &= [ \ 2500 \ 0 \ 18000 \ 0 \ ] \end{aligned}$$

With (11.2.12) the feedback vector is

$$\mathbf{k}_1^T = [\kappa_a \ \kappa_b] \begin{bmatrix} 2500 & 0 & 18000 & 0 \\ 0 & 2500 & 0 & 18000 \end{bmatrix} = [ \ 2500\kappa_a \ 2500\kappa_b \ 18000\kappa_a \ 18000\kappa_b \ ] \quad (11.2.13)$$

which defines the invariance plane. If full state-feedback is available, then any desired location of the movable eigenvalues as defined by  $t(s) = t_0 + t_1 s + s^2$  is achieved by  $\kappa_a = t_0 - d_0 = t_0$  and  $\kappa_b = t_1 - d_1 = t_1$ . The  $\Gamma$ -stable region in the  $(\kappa_a, \kappa_b)$ -plane is now obtained by its boundary description  $t_0(\alpha) = \sigma^2(\alpha) + \omega^2(\alpha)$ ,  $t_1(\alpha) = -2\sigma(\alpha)$ ,  $\sigma < -0.25$ , and with  $\omega^2 = 4\sigma(\alpha)^2 - 0.25$ ,  $\sigma = \alpha$ :

$$\left. \begin{array}{l} \kappa_a = 5\alpha^2 - 0.25 \\ \kappa_b = -2\alpha \end{array} \right\} \alpha < -0.25$$

The real root boundary is  $t(-0.25) = t_0 - 0.25t_1 + 0.0625 = \kappa_a - 0.25\kappa_b + 0.0625$ . Both boundaries are plotted in Fig. 11.4. Now choose for example

$$\boldsymbol{\kappa}^{(1)} = \begin{bmatrix} \kappa_a \\ \kappa_b \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.6 \end{bmatrix}, \text{ i.e. } \mathbf{k}_1^T = [ \ 250 \ 1500 \ 1800 \ 10800 \ ]$$

that  $\Gamma$ -stabilizes  $t(s)$ . It shifts the double open-loop eigenvalue at  $s = 0$  to the roots of  $t(s) = 0.1 + 0.6s + s^2$ . In the second design step the roots of  $t(s)$  are fixed and the roots of  $h(s) = 25/12 + s^2$  are  $\Gamma$ -stabilized by a second term  $\mathbf{k}_2^T$  in the state feedback vector  $\mathbf{k} = \mathbf{k}_1 + \mathbf{k}_2$ .  $\mathbf{k}_2$  must be chosen such that the roots of  $h(s)$  are shifted to the roots of  $r(s) = r_0 + r_1 s + s^2$  with the same  $\Gamma$ -stability constraint for  $r(s)$  as for  $t(s)$ . For the calculation of the pole placement matrix of the pair  $(\mathbf{A} - b\mathbf{k}_1^T, b)$  we need a useful general result; we interrupt the example to present it.  $\square$

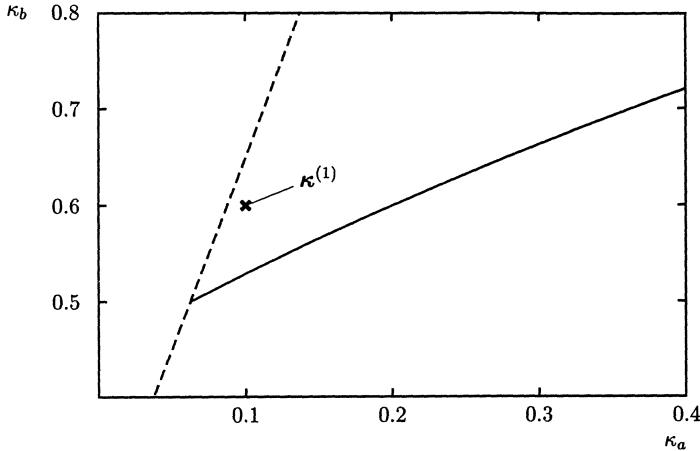


Fig. 11.4: The two eigenvalues on the imaginary axis are invariant in this plane. The two eigenvalues at the origin are  $\Gamma$ -stabilized for example by  $\kappa^{(1)}$ .

*Theorem 11.2. (Ackermann)*

The pole placement matrices  $E$  and  $E_F$  of two pairs  $(A, b)$  and  $(F, b) = (A - bk^T, b)$  are related by

$$E_F = \begin{bmatrix} e_F^T \\ e_F^T F \\ \vdots \\ e_F^T F^{n-1} \\ e_F^T F^n \end{bmatrix} = \begin{bmatrix} e^T \\ e^T A \\ \vdots \\ e^T A^{n-1} \\ e^T A^n - k^T \end{bmatrix} = E - \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ k^T \end{bmatrix} \quad (11.2.14)$$

where  $e_F^T = [0 \dots 0 \ 1][b \ Fb \ F^{n-1}b]^{-1}$ . For the proof the reader is referred to [5].

□

By Theorem 11.2 the invariance plane is obtained in all design iterations from the same matrix

$$\bar{E} = \begin{bmatrix} e_F^T \\ e_F^T F \\ \vdots \\ e_F^T F^{n-1} \end{bmatrix} = \begin{bmatrix} e^T \\ e^T A \\ \vdots \\ e^T A^{n-1} \end{bmatrix} \quad (11.2.15)$$

Equation (11.2.12) may be written as

$$k^T = [\kappa_a \ \kappa_b] \begin{bmatrix} e_h^T \\ e_h^T A \end{bmatrix}$$

$$= [\kappa_a \ \kappa_b] \begin{bmatrix} h_0 & h_1 & \dots & h_{n-2} & 0 \\ 0 & h_0 & \dots & h_{n-3} & h_{n-2} \end{bmatrix} \bar{\mathbf{E}} \quad (11.2.16)$$

*Example 11.3.* (continued)

With Theorem 11.2 we have for the second design step  $\mathbf{F} = \mathbf{A} - \mathbf{b}\mathbf{k}_1^T$  and

$$\mathbf{e}_t^T := \mathbf{e}_F^T t(\mathbf{F}) = [t_0 \ t_1 \ 1] \begin{bmatrix} \mathbf{e}_F^T \\ \mathbf{e}_F^T \mathbf{F} \\ \mathbf{e}_F^T \mathbf{F}^2 \end{bmatrix} = [t_0 \ t_1 \ 1] \begin{bmatrix} \mathbf{e}^T \\ \mathbf{e}^T \mathbf{A} \\ \mathbf{e}^T \mathbf{A}^2 \end{bmatrix}$$

Now (11.2.12) takes the form

$$\mathbf{k}_2^T = [\kappa_a \ \kappa_b] \begin{bmatrix} \mathbf{e}_t^T \\ \mathbf{e}_t^T \mathbf{F} \end{bmatrix}$$

The term  $\mathbf{e}_t^T \mathbf{F}$  contains powers of  $\mathbf{A}$  only up to  $\mathbf{A}^3$ . Thus, by (11.2.14),  $\mathbf{e}_t^T \mathbf{F} = \mathbf{e}_t^T \mathbf{A}$  and

$$\mathbf{k}_2^T = [\kappa_a \ \kappa_b] \begin{bmatrix} \mathbf{e}_t^T \\ \mathbf{e}_t^T \mathbf{A} \end{bmatrix}$$

In other words, we obtain the same result  $\mathbf{k} = \mathbf{k}_1 + \mathbf{k}_2$  no matter if we first shift  $h(s)$  or  $d(s)$ . For fixed  $t_0 = 0.1$ ,  $t_1 = 0.6$  we have

$$\begin{aligned} \mathbf{e}_t^T &= [0.1 \ 0.6 \ 1] \begin{bmatrix} 1200 & 0 & 14400 & 0 \\ 0 & 1200 & 0 & 14400 \\ 0 & 0 & -12000 & 0 \end{bmatrix} \\ &= [120 \ 720 \ -10560 \ 8640] \\ \mathbf{k}_2^T &= [\kappa_c \ \kappa_d] \begin{bmatrix} 120 & 720 & -10560 & 8640 \\ 0 & 120 & -7200 & -10560 \end{bmatrix} \end{aligned}$$

where  $\kappa_c = r_0 - h_0 = r_0 - 25/12$ ,  $\kappa_d = r_1 - h_1 = r_1$ . The total state-feedback gain vector is

$$\mathbf{k} = \mathbf{k}_1 + \mathbf{k}_2 = [k_1 \ k_2 \ k_3 \ k_4]^T$$

with

$$\begin{aligned} k_1 &= 250 + 120\kappa_c \\ k_2 &= 1500 + 720\kappa_c + 120\kappa_d \\ k_3 &= 1800 - 10560\kappa_c - 7200\kappa_d \\ k_4 &= 10800 + 8640\kappa_c - 10560\kappa_d \end{aligned}$$

The  $\Gamma$ -stability region in the  $(\kappa_c, \kappa_d)$ -plane has the same shape as Fig. 11.4 with  $\kappa_a$  replaced by  $\kappa_c + 25/12$  and  $\kappa_b$  replaced by  $\kappa_d$ . It is shown in Fig. 11.5.

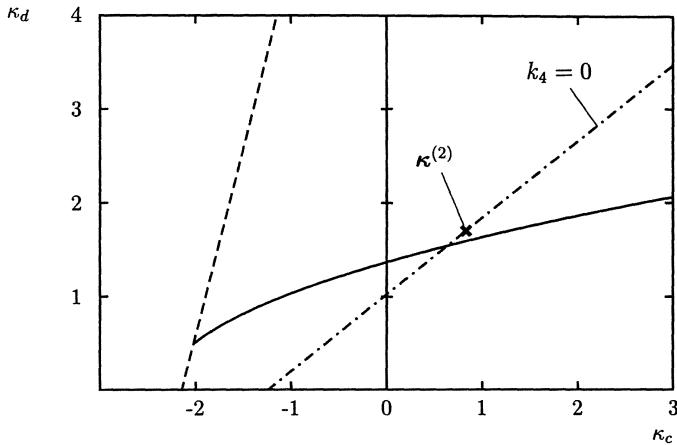


Fig. 11.5: Invariance plane for the second design step

A practically useful choice of  $\kappa_c, \kappa_d$  from the  $\Gamma$ -stable region is one for which we do not have to feed back the rope angle rate, i.e.

$$k_4 = 10800 + 8640\kappa_c - 10560\kappa_d = 0$$

This line is also shown in Fig. 11.5. We choose  $\kappa_c = 0.83$  and  $\kappa_d = 1.7$  on this line. Then the second pair of roots moves from the imaginary axis to the roots of  $r(s) = \kappa_c + 25/12 + \kappa_d s + s^2 = 2.91 + 1.7s + s^2$ , and the total feedback vector is

$$\mathbf{k}^T = [ 349 \ 2300 \ -19181 \ 0 ]$$

□

The example has shown that pole region assignment by sequential shifting of pole pairs into  $\Gamma$  is feasible. We will come back to this approach at the end of the next section for the simultaneous stabilization of vertex plants.

### 11.3 Intersections in Controller Parameter Space

We return now to the problem of simultaneous stabilization of vertex plants. The general concept of simultaneous  $\Gamma$ -stabilization was explained already in Section 3.3, see Fig. 3.14. With the results of Section 11.2 we are now able to compute the boundary of the simultaneously  $\Gamma$ -stabilizing controller parameter set  $K_\Gamma^{(1)}$  in a two-dimensional cross section of the feedback gain space. The same procedure must be applied to all vertices in order to determine the intersection  $K_\Gamma$  of simultaneous  $\Gamma$ -stabilizers. The procedure is first shown for a controller structure with only two free parameters. Note that it may happen that the intersection  $K_\Gamma$  is empty. Then some design specifications have to be weakened or the controller structure must be modified.

*Example 11.4.* Consider again Example 11.1, but now let the rope length  $\ell$  vary in the interval  $[8; 16]$  [m] and the load mass  $m_L$  in the interval  $[1000; 2000]$  [kg]. First, do a robustness analysis for the feedback loop with the feedback law (11.2.5) that was designed for a nominal operating point  $\mathbf{q}^0 = [m_L^0; \ell^0]^T = [1500 \text{ [kg]}; 12 \text{ [m]}]^T$ . The  $\Gamma$ -stability boundaries in the  $(m_L, \ell)$ -plane are shown in Fig. 11.6. By minimization of  $k_2^2 + k_3^2$  the feedback was designed such that the nominal operating condition  $\mathbf{q}^0$  yields

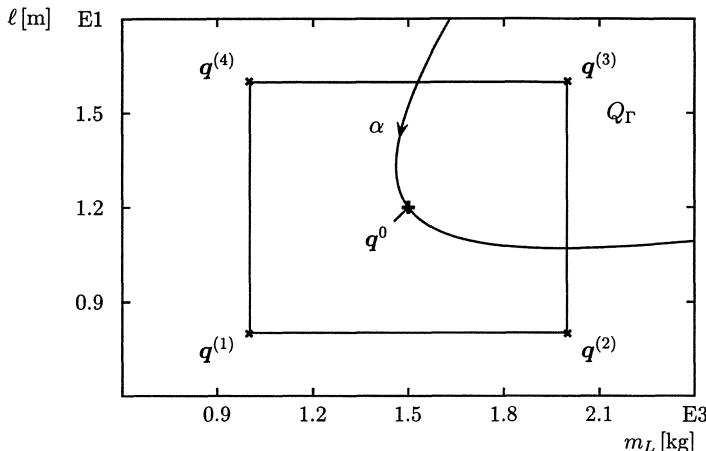


Fig. 11.6:  $\Gamma$ -Stability region in the  $(m_L, \ell)$ -plane for the feedback vector (11.2.5)

eigenvalues on the boundary of the  $\Gamma$ -stability region. Therefore, a  $\Gamma$ -stability boundary passes through  $\mathbf{q}^0$  in Fig. 11.6.

This example illustrates a typical conflict between robustness and optimization with respect to other performance criteria, in the example  $\|\mathbf{k}\|$ . Optimization has pushed the solution to the boundary of  $\Gamma$ -stability and has not left any room for robustness.

We continue the example now with a redesign of the controller  $u = -[500 \ k_2 \ k_3 \ 0]\mathbf{x}$  such that it simultaneously stabilizes the four vertex plants with the parameter vectors

$$\mathbf{q}^{(1)} = \begin{bmatrix} 1000 \\ 8 \end{bmatrix} \quad \mathbf{q}^{(2)} = \begin{bmatrix} 2000 \\ 8 \end{bmatrix} \quad \mathbf{q}^{(3)} = \begin{bmatrix} 2000 \\ 16 \end{bmatrix} \quad \mathbf{q}^{(4)} = \begin{bmatrix} 1000 \\ 16 \end{bmatrix} \quad (11.3.1)$$

The closed-loop characteristic polynomial with  $m_C = 1000$  [kg] and  $g = 10$  [ $\text{m} \cdot \text{s}^{-2}$ ] is

$$p(s, k_3, k_4, m_L, \ell) = 5000 + 10k_2 s + (10000 - k_3 + 500\ell + 10m_L)s^2 + k_2\ell s^3 + 1000\ell s^4$$

The  $\Gamma$ -stability boundaries are generated for each of the four vertex points and displayed in the  $(k_2, k_3)$ -plane. The results are shown in Figures 11.7–11.10.

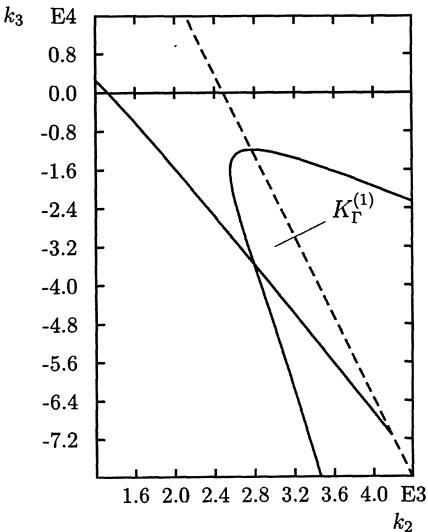


Fig. 11.7: Set of  $\Gamma$ -stabilizing controllers for  $\ell = 8$  [m],  $m_L = 1000$  [kg]

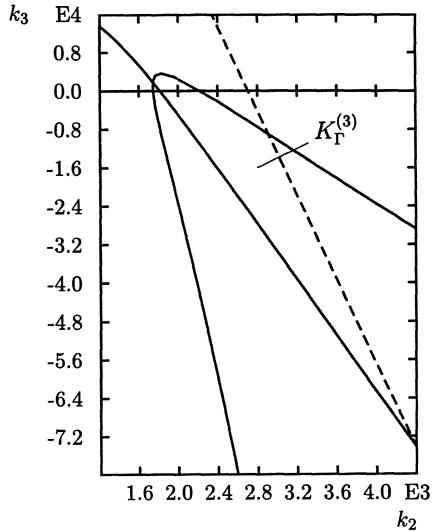


Fig. 11.9: Set of  $\Gamma$ -stabilizing controllers for  $\ell = 16$  [m],  $m_L = 2000$  [kg]

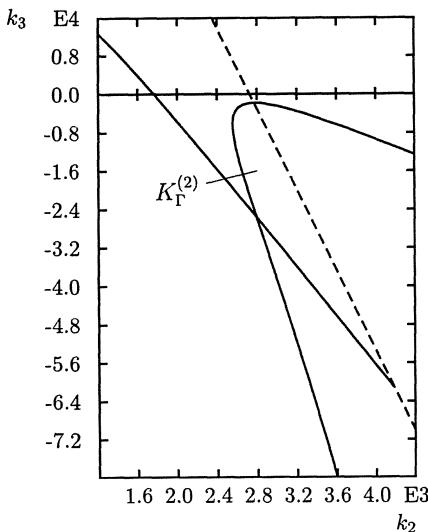


Fig. 11.8: Set of  $\Gamma$ -stabilizing controllers for  $\ell = 8$  [m],  $m_L = 2000$  [kg]

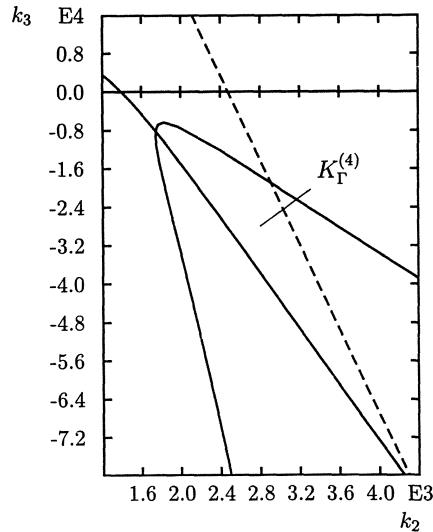


Fig. 11.10: Set of  $\Gamma$ -stabilizing controllers for  $\ell = 16$  [m],  $m_L = 1000$  [kg]

The intersection  $K_\Gamma$  of the four sets  $K_\Gamma^{(i)}$  is displayed in Fig. 11.11. Fig. 11.12 gives a

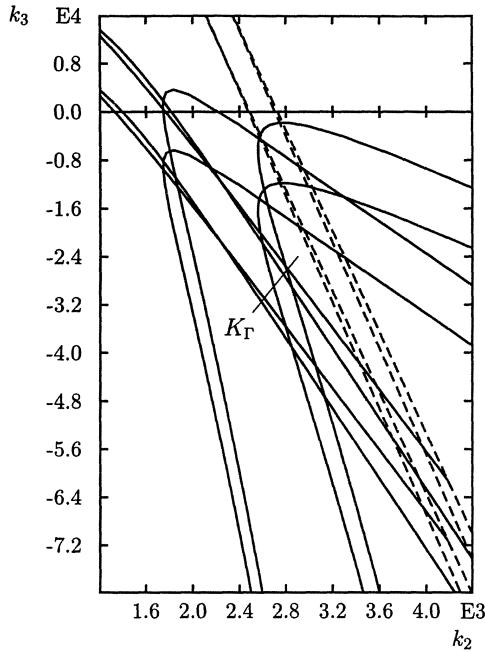


Fig. 11.11: Set of  $\Gamma$ -stabilizing controller parameters for the vertex plants

detailed view of the simultaneously stabilizing set. Each controller in this set simulta-

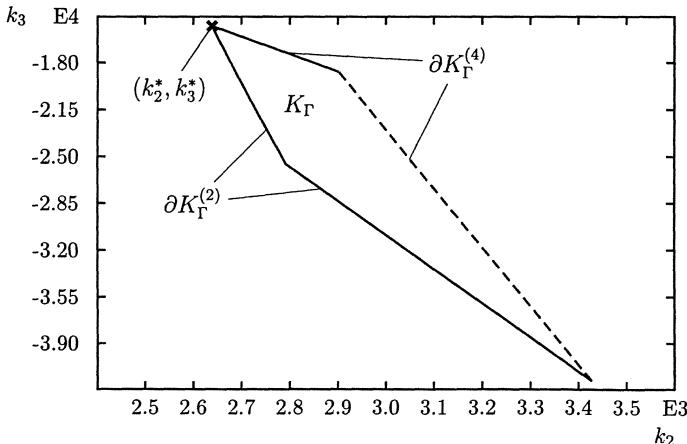


Fig. 11.12: Detailed view of the  $\Gamma$ -stabilizing set

neously stabilizes the four vertices of the rectangular operating domain  $Q$ . Following

the procedure in Example 11.1 the point in the solution set which minimizes  $k_2^2 + k_3^2$  is selected. It has the coordinates  $k_2^* = 2639$  and  $k_3^* = -15255$ . Hence, the control law is

$$u = - \begin{bmatrix} 500 & 2639 & -15255 & 0 \end{bmatrix} \mathbf{x} \quad (11.3.2)$$

Again a robustness analysis is done. The point in Fig. 11.12 which minimizes  $k_2^2 + k_3^2$  in the  $\Gamma$ -stabilizing set lies exactly on an intersection of the complex roots boundaries of  $K_{\Gamma}^{(2)}$  and  $K_{\Gamma}^{(4)}$ . According to the Boundary Crossing Theorem 4.3 the operating conditions  $\mathbf{q}^{(2)}$  and  $\mathbf{q}^{(4)}$  must yield complex eigenvalues on  $\partial\Gamma$ . This is verified in Fig. 11.13, where the  $\Gamma$ -stability boundaries for the resulting closed-loop system in the  $(m_L, \ell)$ -plane are displayed. The complex roots boundary passes through the points  $\mathbf{q}^{(2)}$  and  $\mathbf{q}^{(4)}$  but does not intersect the box anywhere else. By shifting the selected point in Fig. 11.12 marginally into the  $\Gamma$ -stabilizing set a robust  $\Gamma$ -stabilizing controller will be obtained.  $\square$

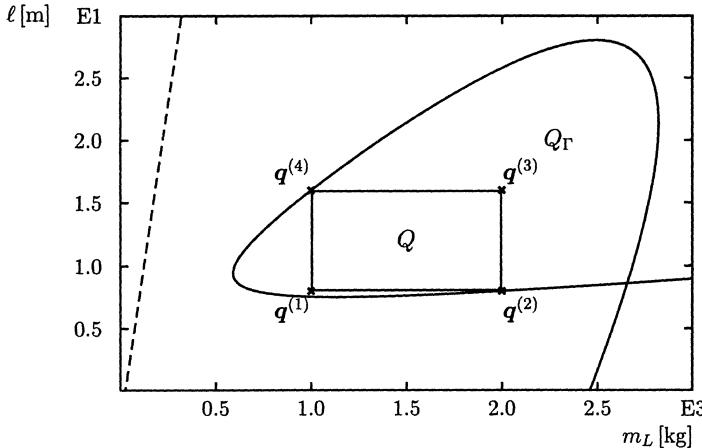


Fig. 11.13:  $\Gamma$ -Stability region in the  $(m_L, \ell)$ -plane for the feedback vector (11.3.2)

#### *Simultaneous Gamma-Stabilization in an Invariance Plane*

Assume now that  $m$  ( $m > 2$ ) feedback gains are available for robust  $\Gamma$ -stabilization. The set of admissible controllers  $K_{\Gamma}$  is the intersection of sets in an  $m$ -dimensional space. It is a difficult mathematical problem to find a solution point that is well centered in  $K_{\Gamma}$ . It is more difficult than finding the radius of stability. There we know the nominal point  $\mathbf{q}^0$  and want to find the smallest destabilizing perturbation in  $\mathbf{q}$ . Now we want for example to find  $\mathbf{k}^0$  such that it has the largest stability radius in  $\mathbf{k}$ -space.

We pursue here a graphical approach. It is naturally limited to two or three dimensions. In 3D displays [142] it is difficult to recognize intersections of objects unless we calculate the penetration lines of boundary surfaces. This, however, would be a time-consuming task. In 2D graphics it is easy for us to recognize intersections like in Fig. 11.11. Note that for the calculation of the curves only function evaluations  $k_1(\alpha)$ ,  $k_2(\alpha)$  must be

made and no iterations. The determination of line intersections (which correspond to the 3D penetration lines) is a time-consuming iteration. Thus, in 2D graphics the detection and interpretation of intersections is left to the design engineer. Since 2D graphics are fast they can be modified in on-line computations, for example by moving the cross section plane through higher-dimensional  $\Gamma$ -stability regions.

For an uncertain plant with state-feedback  $u = -\mathbf{k}^T \mathbf{x}$  the idea of an invariance plane is a useful tool in design strategies that have to be adapted to the special problem formulation. The idea is that the feedback vector is expressed as

$$\mathbf{k} = \mathbf{k}_1 + \mathbf{k}_2 + \dots + \mathbf{k}_N$$

where first  $\mathbf{k}_1$  is determined such that at least the two worst eigenvalues, associated with a worst operating condition  $j^{(0)}$ , are  $\Gamma$ -stabilized, while the remaining eigenvalues for the operating condition  $j^{(0)}$  remain unchanged. By these requirements a cross section plane (11.2.12) for a two-dimensional display is fixed. The  $\Gamma$ -stability boundaries for all vertices can now be displayed in the above cross section plane. Starting from  $\kappa_1 = 0$ ,  $\kappa_2 = 0$  a path is sought such that as many eigenvalues as possible are  $\Gamma$ -stabilized. Next, the modified plant with dynamics matrix  $\mathbf{A}^{(1)} = \mathbf{A} - \mathbf{b}\mathbf{k}_1^T$  is analyzed. In the next iteration the two worst eigenvalues associated with a worst operating condition  $j^{(1)}$  are selected and a new cross section plane is determined in which as many eigenvalues as possible are  $\Gamma$ -stabilized, etc. Note that it is not necessary to recalculate the matrix  $\bar{\mathbf{E}}$  in (11.2.16) in the iterations because  $\bar{\mathbf{E}}$  is feedback invariant, i.e. the pair  $(\mathbf{A}, \mathbf{b})$  and the pair  $(\mathbf{A} - \mathbf{b}\mathbf{k}_1^T, \mathbf{b})$  yield the same  $\bar{\mathbf{E}}$  matrix in (11.2.15). Also, after  $\mathbf{k}$  has been fixed in an iteration, the characteristic polynomial by (2.2.12) uses the same matrix  $\mathbf{W}(\mathbf{A}, \mathbf{b}) = \mathbf{W}(\mathbf{A} - \mathbf{b}\mathbf{k}^T, \mathbf{b})$  and can be used in each iteration step. Note that  $\mathbf{W}$  and  $\bar{\mathbf{E}}$  are related by  $\bar{\mathbf{E}} = \mathbf{W}^{-1}$  [5]. An example for a controller design in an invariance plane will be given in Section 11.5.

It is possible also to modify the above iterative design strategy by using different  $\Gamma$ -stability regions in the design iterations. One may start with a wide region such that  $\partial\Gamma$  separates just the two worst eigenvalues from all others. In further design iterations  $\Gamma$  may be contracted to the desired final  $\Gamma$ -region. An interactive method for simultaneous  $\Gamma$ -stabilization was also developed in [96]. By stepwise contraction of the pole region the eigenvalues of the system are shifted towards the desired pole region  $\Gamma$ . This was accomplished by selecting the smallest pole region which contained all eigenvalues for one or several nominal operating points. Then, the controller parameters were optimized such that the distance from the stability boundaries in the  $\mathbf{k}$ -space was maximized, i.e. the eigenvalues were shifted away from the boundaries of the pole region into the region. This procedure was repeated until the pole region was close enough to the desired pole region  $\Gamma$ .

In case of single loop feedback with a compensator we still can define an arbitrary two-dimensional cross section plane in  $\mathbf{k}$ -space and find there the intersection set for a finite plant family. If this intersection set is very small or does not exist, the cross section plane has to be moved. Due to the simplicity of the mapping equations (11.2.2), these boundaries can be generated in less than one second on a workstation or a fast PC even for a larger number of nominal plants. A cross section plane is defined by

fixing all free controller parameters  $k_3$  to  $k_m$ . The set boundaries are then displayed in the  $(k_1, k_2)$ -plane. By changing the fixed parameters interactively using adequate input devices (for example mouse, dial box, steering ball) we can watch the shrinkage or growth of the intersection set in the displayed plane. After a short time we will find out the directions in the  $(m-2)$ -dimensional space, for which the intersection set gets smaller and those directions for which the intersection set grows. In [16] it was shown that this intuitive search will result quickly in useful controller parameters.

## 11.4 Selection of a Controller from the Admissible Set

So far we have tried to obtain a global picture of an admissible solution set. Of course we finally have to select a point with desirable properties from this set. A common procedure is to define design as an optimization problem. Here, however, design is understood as a tradeoff between several goals formulated by “soft” inequalities, i.e. performance bounds which may be shifted during the design process. A design method should offer some insights into which requirements are in conflict and which are not. Also the designer would like to learn easily which additional or tightened specifications could be satisfied, and for which specifications a high price must be paid.

In this section several typical design requirements are discussed which can be easily interpreted in  $\mathbf{k}$ -space. The next two sections show the application to an automatic steering and a flight control problem.

### *Simulations with Nonlinear Plant*

The multi-model problem frequently arises from a nonlinear plant which is linearized in different operating conditions. In the flight control example the linear description by (1.4.2) holds only for small deviations from flight conditions at constant altitude and with constant velocity. Also the slow phugoid mode and the coupling with the lateral motion is neglected. For changing flight conditions a nonlinear simulation is required to test and refine the controller. The solution obtained by solving the multi-model problem satisfies only a necessary condition for nice stability of the nonlinear plant, not a sufficient one. A controller structure and an admissible parameter region are obtained which can be used as promising controller candidates for the nonlinear simulation. In other words: many controllers are excluded at the beginning from the expensive simulation because they do not satisfy the necessary condition that the locally linearized airplane is  $\Gamma$ -stabilized.

### *Solutions with Small Loop Gains*

In control systems frequently only a constrained input amplitude  $|u| \leq U$  is available, in particular if  $u$  is a force accelerating a mass. For state feedback  $u = -\mathbf{k}^T \mathbf{x}$  the

magnitude  $|u|$  is bounded by

$$|u| = |\mathbf{k}^T \mathbf{x}| \leq \|\mathbf{k}^T\| \times \|\mathbf{x}\| \quad (11.4.1)$$

Usually, very little is known about the probability distribution of the states  $\mathbf{x}$ ; a uniform distribution over the unit ball is the simplest assumption, i.e.  $\|\mathbf{x}\| = |\mathbf{x}^T \mathbf{x}| = 1$ . Here it is assumed that the individual state variables  $x_i$  are always individually scaled with respect to their maximal possible or expected values. The worst case of equality arises in (11.4.1) if  $\mathbf{k} = c\mathbf{x}$  for some real scalar  $c$ . Thus, the norm  $\|\mathbf{k}\|$  is immediately suitable as a measure for the maximal necessary input amplitude  $|u|$ .

There is a second practical reason why the loop gain should not be chosen too high. Practically,

$$u = -\mathbf{k}^T(\mathbf{x} + \Delta\mathbf{x}) \quad (11.4.2)$$

is formed instead of the ideal control law  $u = -\mathbf{k}^T \mathbf{x}$ .  $\Delta\mathbf{x}$  can be a measurement noise or a quantization error in the analog-digital conversion, for example.

As the gain  $\|\mathbf{k}\|$  decreases, the effect of such an error  $\mathbf{k}^T \Delta\mathbf{x}$  is reduced as well. According to these considerations it is reasonable to select the point from the allowable solution set which is closest to the origin in  $\mathbf{k}$ -space. In the example of Fig. 11.14 this is the corner D of the admissible quadrilateral.

#### *Safety Margin from the Boundary Surfaces*

If a corner point is chosen in the admissible solution set, as the point D in Fig. 11.14, then small gain changes in  $k_1$  or  $k_2$  bring the system out of the admissible region. This can be caused by a quantized storage of the control coefficients as  $\mathbf{k}^T + \Delta\mathbf{k}^T$ , or it can arise from the fact that the system models were not exact. Consequently, in practice a safety margin from the boundary of the  $\Gamma$ -stability region should be maintained which, for example, ensures that the hypercube  $k_i \pm \Delta k$ ,  $i = 1, 2, \dots, n$  for a given maximum quantization error  $\Delta k$  is contained in the admissible region. In the example of Fig. 11.14 this is a square with the edge lengths of  $2\Delta k$  parallel to the axes. (A hypersphere can also be used. In the two-gain example this is a circle around  $\mathbf{k}$ .) In Fig. 11.14 point E produces the maximal safety margin  $\Delta k$  and point F represents a suitable compromise with the demands of small loop gain  $\|\mathbf{k}\|$  for given  $\Delta k_0$ .

Also for solutions found by optimization procedures, it is helpful to examine the neighborhood of the solution, especially its distance from the boundary, in different two-dimensional intersections (stability margin analysis).

#### *Gain Reduction Margins*

In some design problems we are interested in stability or  $\Gamma$ -stability margins for gain reductions. This is for example true if one of several parallel components may fail or if a saturation nonlinearity as in Fig. 11.15 occurs in the actuator.

For a maximum input amplitude  $A > 1$  a reduction of the linearized gain to  $1/A$  occurs. Here we have two design possibilities:

- a) Use a small loop gain in order to keep the actuator in its linear operating range. Within this range the pole region assignment guarantees  $\Gamma$ -stability.

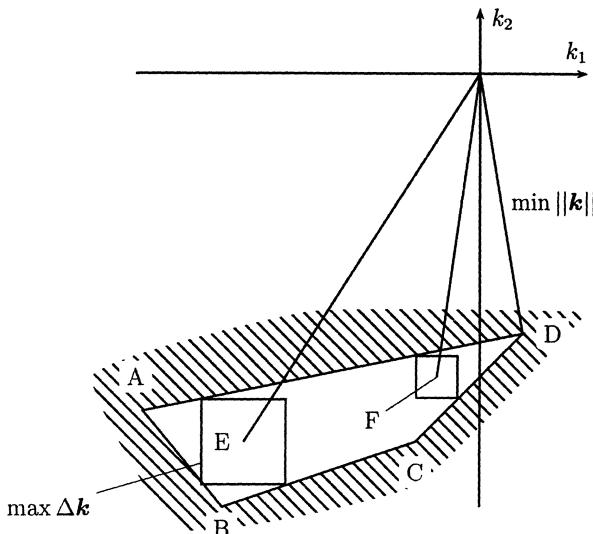


Fig. 11.14: D: Minimal loop gain  $\|k\|$ , E: Maximal safety margin  $\Delta k$ , F: Compromise for given  $\Delta k_0$

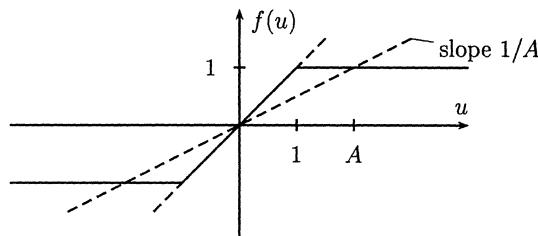


Fig. 11.15: Saturation and its gain reduction effect

- b) Let the actuator go into saturation to make use of its available maximum output magnitude. Then the stability of the nonlinear loop must be tested. The Popov criterion is suited for this purpose. In this case absolute stability in the sector  $1/A \leq k \leq 1$  must be assured. This requires a linear gain reduction margin from 1 to  $1/A$ .

The position in the loop where the gain reduction may occur is important. Consider the three candidates  $a$ ,  $b$ , and  $c$ , in Fig. 11.16. Assume that the  $\Gamma$ -stability region in  $(k_1, k_2)$ -plane is the triangle of Fig. 11.17.

The point  $A$  has the largest gain reduction margin in the direction  $k_1$ , i.e. for case  $a$  in Fig. 11.16. Correspondingly, the point  $B$  is favorable for case  $b$  and point  $C$  for case  $c$ .

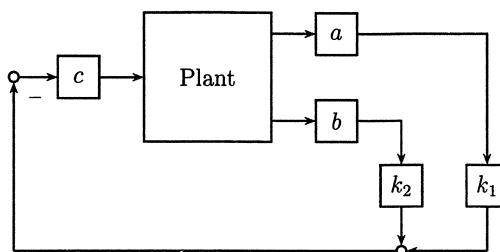
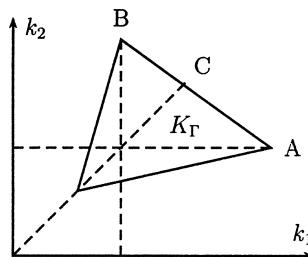
Fig. 11.16: Gain reductions may occur at  $a, b$  or  $c$ 

Fig. 11.17: Maximal gain reduction margin.

### *Robustness against Sensor Failures*

Usually, control systems are designed under the assumption that sensors do not fail. Redundancy management has to provide then the required measurements with only very short interruptions by failure of individual sensors. If the plant is an unstable aircraft, for example, this means that failure detection is vital for stabilization. Detection must operate quickly and this is in conflict with the requirement of low probability of false alarms.

An alternative is the use of a hierarchical concept. Its basic level is a fixed gain control system which is designed such that pole region requirements are robust with respect to component failures and uncertain parameters [4]. All the more sophisticated tasks, like failure detection and redundancy management, plant parameter identification and controller parameter adaptation or gain scheduling are assigned to higher levels if they are required for best performance. The higher levels process more information and are operating in a slower time scale than the basic level. Since the higher levels are not vital for stabilization they can make their decisions without panic.

Assume that a sensor failure has two effects as illustrated by Fig. 11.18.

1. The multiplicative effect reduces the gain  $V$  from its nominal value 1 to zero or some value in between.
2. The additive effect introduces a bias or noise  $d$  at the output.

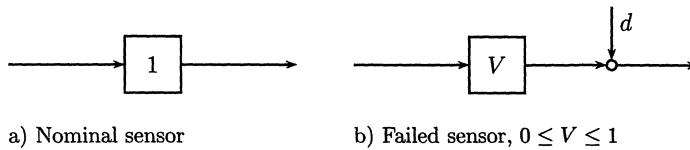
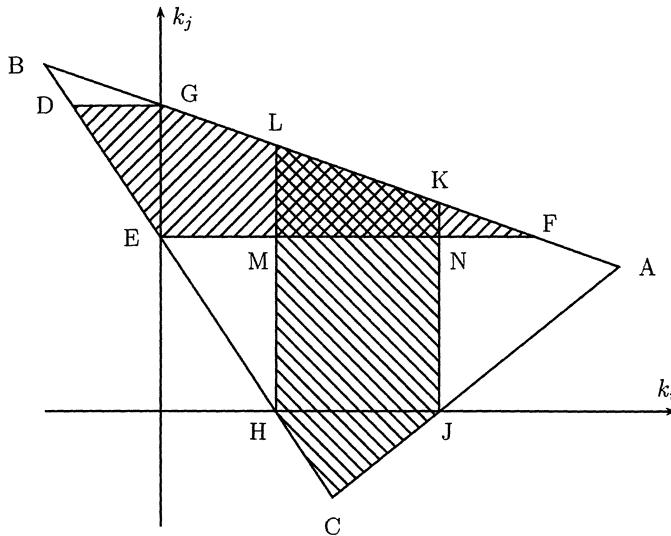


Fig. 11.18: Model of sensor failures

As far as the eigenvalue location is concerned, only the multiplicative effect is important. The additive effect may require that the failed sensor signal is switched off. This decision of a failure detecting system may be slow, e.g. if the plant operates in a steady state and one of the sensors sticks at one value.

If the measured variables are used as state variables (“sensor coordinates”), then the sensor failure is equivalent to reducing the corresponding feedback gain. For  $V = 0$  a solution for which  $\Gamma$ -stability is robust against failure of the sensor for the state variable  $x_i$  is characterized in gain space by the fact that the projection of  $k$  on the subspace  $k_i = 0$  is contained in  $K_\Gamma$ . A two-dimensional cross section through gain space is shown in Fig. 11.19. Assume that the admissible region  $K_\Gamma$  is the triangle ABC. Robustness against failure of sensor  $i$  is achieved if the projection of the appropriate gain is on GE. This property holds for DEFG. Similarly, HCJKL is robust against failure of sensor  $j$  and KLMN is robust against failure of either  $i$  or  $j$ .

Fig. 11.19: In KLMN  $\Gamma$ -stability is robust against failures of the type  $k_i = 0$  or  $k_j = 0$ 

If no such intersection of the  $\Gamma$ -stability region with the axes exists, then the designer has two choices:

1. Ask for the best possible system property that can still be achieved under sensor

failure.

2. Use redundant sensors in parallel.

The first possibility is illustrated by Fig. 11.20. Robustness against sensor failure cannot

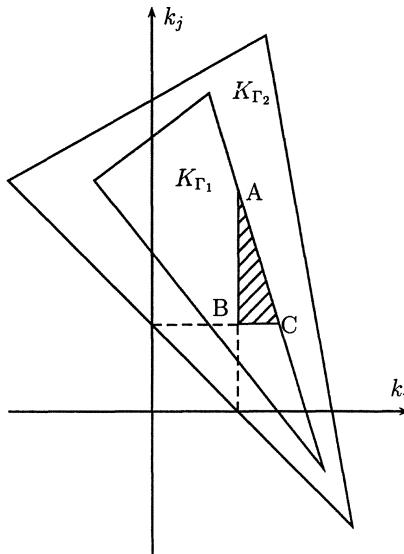


Fig. 11.20: Sensor failure robustness cannot be achieved for  $\Gamma_1$ -stability but for  $\Gamma_2$ -stability

be achieved for  $\Gamma_1$ -stability. For the case of sensor failure, an emergency specification may exist. With this specification a relaxed pole region  $\Gamma_2$  can be formulated. In the example of Fig. 11.20 we can determine a set  $K_{\Gamma_{2a}} \subset K_{\Gamma_2}$  which guarantees  $\Gamma_2$ -stability also for the case of sensor failure. A good choice is a controller from the intersection between this set and  $K_{\Gamma_1}$ : During normal operation the system is  $\Gamma_1$ -stable and in the case of sensor failure at least  $\Gamma_2$ -stability is retained. In the example of Fig. 11.20 this is guaranteed for the triangle ABC.

For the second approach the simplest possibility is to use two parallel sensors. Their outputs are multiplied by a factor 1/2 and added to produce  $x_i$  in the unfailed case. If one of the two sensors fails, then  $k_i$  is reduced by 50%. In order to achieve robustness,  $k_i$  has to be selected such that it has a 50% gain reduction margin in  $k_i$  for  $\Gamma$ -stability. If  $K_\Gamma$  is the triangle ABC in Fig. 11.21, then DEF is the region for which  $\Gamma$ -stability is achieved after a 50% reduction of  $k_i$ . Thus, the triangle EGH contains the admissible points which are  $\Gamma$ -stable and also remain  $\Gamma$ -stable after 50% gain reduction.

If an intersection still does not exist for two parallel sensors, then three sensors in parallel may be considered. An advantage of this choice is that it can be combined with a failure detection procedure on the next higher hierarchical level. Its structure is shown in Fig. 11.22.

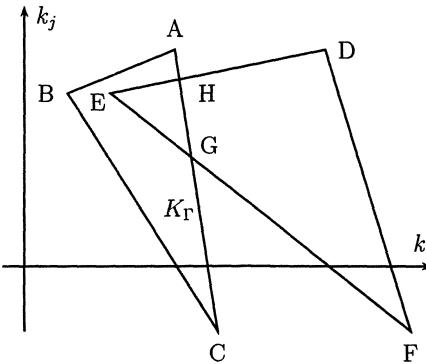


Fig. 11.21: In EGH  $\Gamma$ -stability is preserved under 50% gain reduction in  $k_i$

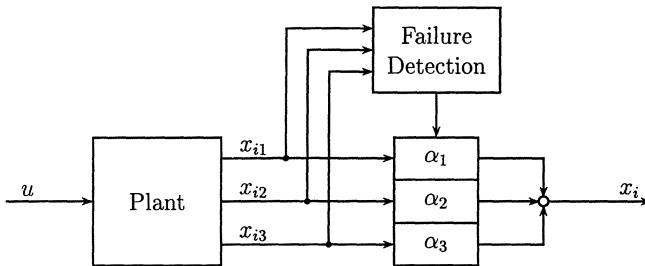


Fig. 11.22: A triplex system

The failure detection system forms the three decision functions

$$\begin{aligned} d_1(t) &= [x_{i1}(t) - x_{i2}(t)][x_{i1}(t) - x_{i3}(t)] \\ d_2(t) &= [x_{i2}(t) - x_{i3}(t)][x_{i2}(t) - x_{i1}(t)] \\ d_3(t) &= [x_{i3}(t) - x_{i1}(t)][x_{i3}(t) - x_{i2}(t)] \end{aligned} \quad (11.4.3)$$

The  $d_k$ 's are nominally zero;  $|d_k| \geq \varepsilon$  indicates a failure of sensor  $k$ . In order to avoid false alarms from short impulses,  $d_k(t)$  is low-pass filtered first and then compared to a threshold value.

$$\begin{aligned} \dot{f}_1(t) &= af_1(t) + d_1(t) \\ \dot{f}_2(t) &= af_2(t) + d_2(t) \\ \dot{f}_3(t) &= af_3(t) + d_3(t) \end{aligned} \quad (11.4.4)$$

The decision logic is then

Nominal state:

$$\begin{aligned} |f_1(t)| < \varepsilon, \quad |f_2(t)| < \varepsilon, \quad |f_3(t)| < \varepsilon \\ \alpha_1 = \alpha_2 = \alpha_3 = 1/3 \end{aligned} \quad (11.4.5)$$

Failure of sensor  $k$ :

$$\begin{aligned} |f_k(t)| &> \varepsilon, \quad |f_j(t)| < \varepsilon \text{ for } j \neq k \\ \alpha_k &= 0, \quad \alpha_j = 1/2 \text{ for } j \neq k \end{aligned} \quad (11.4.6)$$

No changes are made in the  $\alpha$ 's after a second failure. The parameters  $a$  and  $\varepsilon$  are chosen in view of safety against false alarms, i.e. both are not too small. This means that the decision may take some time. Between failure and decision times the gain is reduced to  $2/3$ . If a second failure occurs after the first decision, then the gain is reduced to  $1/2$ . In the unlikely case that a second failure occurs before the first one is detected, the gain is only  $1/3$ . Thus, the basic robust control system should be designed for this gain reduction margin of 50% or 67% for  $\Gamma$ -stability.

In applications where the sensors are expensive, it is desirable to substitute some measurements by feedback variables generated by feedback dynamics. The observer-state-feedback structure is feasible for this purpose, but other structures may be more convenient in the design process. This is particularly true for the design for robustness with respect to sensor failures. An observer with reduced sensor gain still has the full gain for the control input  $u$ . Thus, its behavior is very different from the nominal observer. The filter feedback structure of Fig. 11.23 is more convenient.

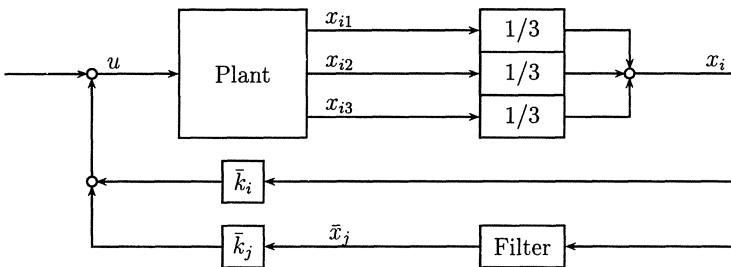


Fig. 11.23: Substituting the measurement of  $x_j$  by a filter producing  $\bar{x}_j$

Here a filter produces a substitute feedback variable  $\bar{x}_j$  for the true state variable  $x_j$ . The filter transfer function is chosen such that the transfer function from  $u$  to  $\bar{x}_j$  at least crudely approximates the transfer function from  $u$  to  $x_j$  within the desired closed-loop bandwidth and over the range of plant parameter variations. The controller structure of Fig. 11.23 is especially useful if the transfer function from  $u$  to  $x_i$  is minimum-phase, because then cancellations or near cancellations by filter poles can be made. Instability of the transfer function is no disadvantage if the same instability also occurs, as is usual, in the transfer function from  $u$  to  $x_j$ . In Section 11.6 this concept will be illustrated by the example of an unstable aircraft.

In the filter structure a failure effects both feedback channels simultaneously, i.e. both gains  $\bar{k}_i$  and  $\bar{k}_j$  are reduced. For example let a failure of one sensor occur, such that the sensor and feedback gain is reduced by a factor  $2/3$ . If ABC in Fig. 11.24 is the  $\Gamma$ -stability region, then DEF is the region for which  $\Gamma$ -stability is maintained after one third gain reduction in  $\bar{k}_i$  and  $\bar{k}_j$ . Thus, AGH is the region in which  $\Gamma$ -stability is achieved both nominally and after the sensor failure.

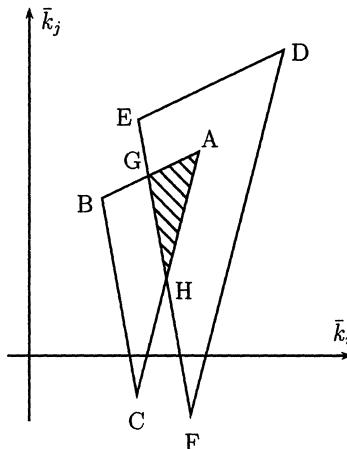


Fig. 11.24: In AGH  $\Gamma$ -stability is preserved under 1/3 gain reduction in both channels

The configuration of Fig. 11.23 with one measured state variable immediately gives the same gain reduction margin in both channels in case of failure of one of several parallel sensors.

## 11.5 Case Study: Automatic Steering of a Bus

In Part I the problem formulation for automatic steering of vehicles was introduced. A linearized mathematical model (1.3.4) was derived from the nonlinear system equations of a single track model. This linear model is used now for a controller design. In the following we want to design an automatic steering system for a bus O 305 with the data given in Table 1.3. The bus only allows front wheel steering.

The given design specifications require that the displacement from the guideline must not exceed 15 [cm], the maximal steering angle is 40 [deg], and the steering angle rate is limited to 23 [deg · s<sup>-1</sup>]. Further, the maximal lateral acceleration has to be lower than 4 [m · s<sup>-2</sup>] because of safety reasons. Passengers feel comfortable if it does not exceed 2 [m · s<sup>-2</sup>]. If we want to solve the design problem with the help of the parameter space method, the specifications have to be “translated” into an adequate  $\Gamma$ -region. If a controller was designed, then the specifications have to be verified by simulations in the time domain. Especially critical driving maneuvers like entering a narrow bus stop bay at low speed or the transition from manual to automatic steering have to be considered here.

In Chapter 1 the number of uncertain parameters of the bus was reduced by some considerations from initially four (physical mass, velocity, moment of inertia, and adhesion coefficient) to two: velocity  $v$  and virtual mass  $\tilde{m}$ . Their ranges are  $v \in$

$[3; 20] [\text{m} \cdot \text{s}^{-1}]$  and  $\tilde{m} \in [9950; 32000] [\text{kg}]$ . The corresponding operating domain is shown in Fig. 11.25.

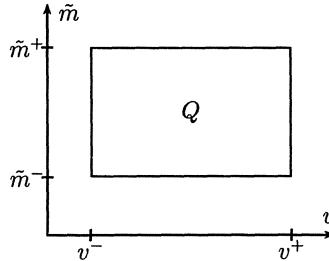


Fig. 11.25: Operating domain of the bus

A first design approach for this problem was presented in [20] using state feedback design concepts. The bus was equipped with three sensors: two displacement sensors at the front and rear of the bus and a sensor for measuring the front steering angle  $\delta_f$ . Additionally, the derivatives of front and rear displacement were used for feedback. However, they were not measured directly but approximated by differentiation and low pass filtering. This was done by a filter

$$f_D(s) = \frac{s}{s^2/\omega_0^2 + 2D s/\omega_0 + 1} \quad (11.5.1)$$

A representation of this controller structure was shown in Fig. 2.5. The conventional power steering of the bus was used as an actuator. It is modelled as a low pass with a time constant of  $1/4.7 [\text{s}]$ . For the design of the feedback vector full state feedback was assumed. Later, the filter coefficients of  $f_D(s)$  are determined such that the dominant eigenvalues are not shifted significantly. For the design the sensor coordinates were chosen for the state vector:

$$\mathbf{x} = [y_f \dot{y}_f y_r \dot{y}_r \delta_f]^T$$

where  $y_f$  and  $y_r$  are front and rear displacement. The state space model (1.3.4) uses the state vector  $[\beta \ r \ \Delta\psi \ y_f]^T$ . It can be transformed to the new state space coordinates by the transformation

$$\begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ v & \ell_{sf} & v & 0 \\ 0 & 0 & -(\ell_{sf} + \ell_{sr}) & 1 \\ v & -\ell_{sr} & v & 0 \end{bmatrix} \begin{bmatrix} \beta \\ r \\ \Delta\psi \\ y_f \end{bmatrix}$$

$\ell_{sf}$  and  $\ell_{sr}$  are the distance between CG and front and rear displacement sensors of the bus. For the given bus O 305 they are  $\ell_{sf} = 6.12 [\text{m}]$  and  $\ell_{sr} = 4.99 [\text{m}]$ . The new state space model has to be augmented by the additional state  $\delta_f$ . The resulting state space

model has the form

$$\begin{bmatrix} \dot{y}_f \\ \ddot{y}_f \\ \dot{y}_r \\ \ddot{y}_r \\ \dot{\delta}_f \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ 0 & 0 & 0 & 1 & 0 \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ 0 & 0 & 0 & 0 & -4.7 \end{bmatrix} \begin{bmatrix} y_f \\ \dot{y}_f \\ y_r \\ \dot{y}_r \\ \delta_f \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 4.7 \end{bmatrix} u$$

The open-loop system has two poles at the origin, an actuator pole at  $s = -4.7$ , and two pole pairs varying with mass and velocity. The location of these poles for the extremal operating conditions are given in Table 11.1.

$\tilde{m}$ [kg]	$v$ [ $\text{m} \cdot \text{s}^{-1}$ ]	open-loop eigenvalues
9950	3	$s_1 = -13.48, s_2 = -22.58$
32000	3	$s_1 = -4.33, s_2 = -6.89$
9950	20	$s_{1,2} = -2.71 \pm j1.09$
32000	20	$s_{1,2} = -0.84 \pm j0.69$

Table 11.1: Location of the operating dependent poles of the bus for the vertex plants

The hyperbola

$$\left(\frac{\sigma}{0.08}\right)^2 - \left(\frac{\omega}{0.12}\right)^2 = 1 \quad (11.5.2)$$

was chosen as boundary of the stability region  $\Gamma$ . Thus, each of the pole pairs in Table 11.1 is  $\Gamma$ -stable. The problem is shifting the double pole at  $s = 0$  to the left of the hyperbola (11.5.2) without destroying the  $\Gamma$ -stability of the other poles. The problem can be solved with the help of an invariance plane. For a nominal operating point we determine such a plane where all eigenvalues but the two at  $s = 0$  remain at their old locations. As nominal operating point  $\mathbf{q}^0 =: \mathbf{q}_{inv}$  for which this plane is computed we select  $v = 10$  [ $\text{m} \cdot \text{s}^{-1}$ ] and  $\tilde{m} = \tilde{m}^+ = 32000$  [kg]. The system matrix  $\mathbf{A}(\mathbf{q}_{inv})$  is then

$$\mathbf{A}(\mathbf{q}_{inv}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1.59 & -1.45 & -1.59 & -0.32 & 19.0 \\ 0 & 0 & 0 & 1 & 0 \\ 2.11 & -0.43 & -2.11 & -1.92 & -4.28 \\ 0 & 0 & 0 & 0 & -4.7 \end{bmatrix} \quad (11.5.3)$$

The input vector  $\mathbf{b} = [0 \ 0 \ 0 \ 0 \ 4.7]^T$  is independent of the uncertain parameters  $\tilde{m}$  and  $v$ . The open-loop characteristic polynomial is

$$p(s, \mathbf{q}^0) = s^2(s + 4.7)(s^2 + 3.36s + 3.16)$$

and the desired closed-loop polynomial is

$$\begin{aligned} p(s, \mathbf{q}_{inv}, \mathbf{k}^0) &= (s^2 + t_1 s + t_0)(s + 4.7)(s^2 + 3.36s + 3.16) \\ &= (s^2 + t_1 s + t_0)(s^3 + 8.06s^2 + 18.97s + 14.84) \end{aligned}$$

where  $s^2 + t_1 s + t_0$  is the new location of the prior unstable double pole at the origin, i.e.

$$h(s) = s^3 + 8.06s^2 + 18.97s + 14.84$$

The invariance plane is determined by (11.2.16) as

$$\begin{aligned} \mathbf{k}^T &= [\kappa_a \ \kappa_b] \begin{bmatrix} \mathbf{e}_h(\mathbf{q}_{inv})^T \\ \mathbf{e}_h(\mathbf{q}_{inv})^T \mathbf{A}(\mathbf{q}_{inv}) \end{bmatrix} \\ &= [\kappa_a \ \kappa_b] \begin{bmatrix} 14.84 & 18.97 & 8.06 & 1 & 0 \\ 0 & 14.84 & 18.97 & 8.06 & 1 \end{bmatrix} \bar{\mathbf{E}}_{inv} \end{aligned} \quad (11.5.4)$$

with

$$\bar{\mathbf{E}}_{inv}^{-1} = \mathbf{W}_{inv} = \begin{bmatrix} 221 & 178 & 89.4 & 0 & 0 \\ 0 & 221 & 178 & 89.4 & 0 \\ 221 & -67.6 & -20.1 & 0 & 0 \\ 0 & 221 & -67.6 & -20.1 & 0 \\ 0 & 0 & 14.8 & 15.8 & 4.7 \end{bmatrix}$$

The invariance plane (11.5.4) is then

$$\mathbf{k}^T = [\kappa_a \ \kappa_b] \begin{bmatrix} 0.0711 & 0.0141 & -0.0040 & 0.0132 & 0 \\ 0.0504 & 0.0449 & -0.0504 & -0.0338 & 0.2128 \end{bmatrix} \quad (11.5.5)$$

For the design the  $\Gamma$ -stability boundaries for the vertex plants of the bus are displayed in this two-dimensional cross section. The four plots are shown in Figures 11.26–11.29. The intersection of the four  $\Gamma$ -stabilizing sets is shown in Fig. 11.30.

Fig. 11.30 shows that also high gain solutions are possible. With respect to the design specifications like actuator constraints and limited lateral acceleration it is recommended to choose a low gain solution. Fig. 11.31 gives a detailed view of the simultaneously  $\Gamma$ -stabilizing set with small gain solutions. The point with the coordinates  $\kappa_a = 0.18$  and  $\kappa_b = 0.75$  was selected. This leads to a feedback vector

$$\mathbf{k}^T = [0.0506 \ 0.0363 \ -0.0385 \ -0.0231 \ 0.1596] \quad (11.5.6)$$

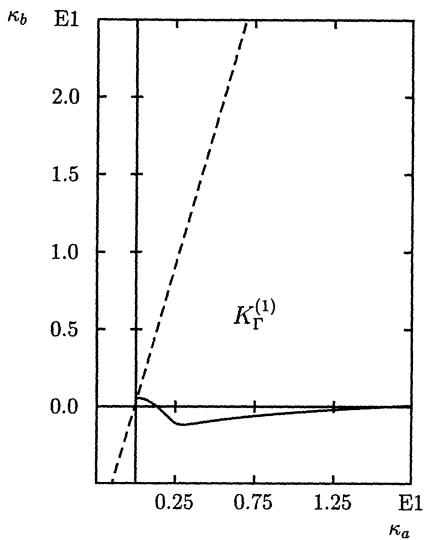


Fig. 11.26: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950$  [kg],  $v = 3$  [ $\text{m} \cdot \text{s}^{-1}$ ]

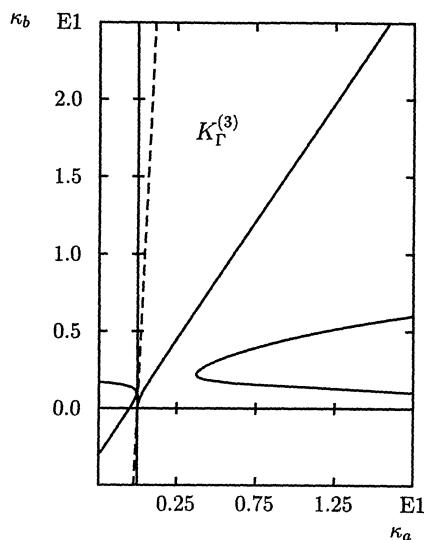


Fig. 11.28: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000$  [kg],  $v = 20$  [ $\text{m} \cdot \text{s}^{-1}$ ]

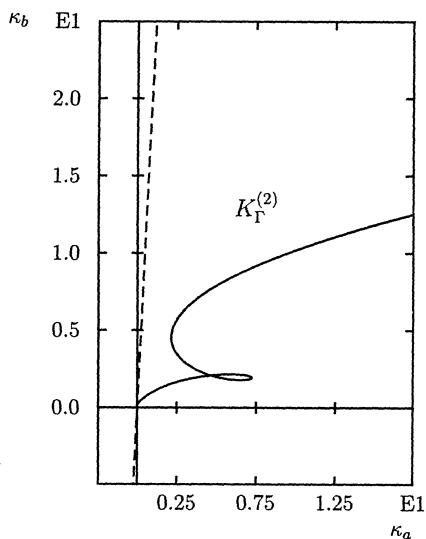


Fig. 11.27: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950$  [kg],  $v = 20$  [ $\text{m} \cdot \text{s}^{-1}$ ]

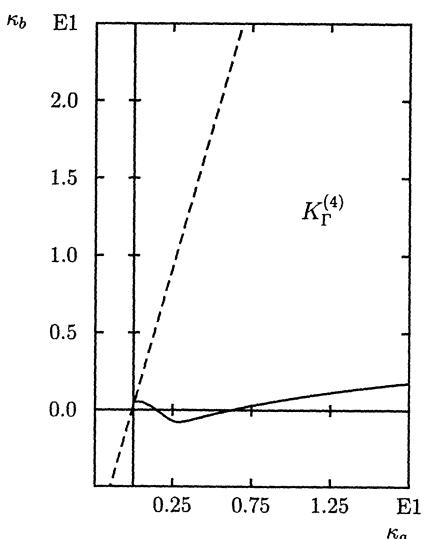


Fig. 11.29: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000$  [kg],  $v = 3$  [ $\text{m} \cdot \text{s}^{-1}$ ]

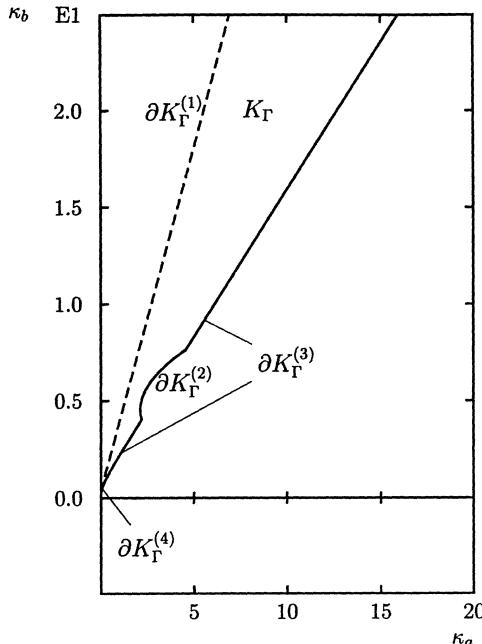


Fig. 11.30: Set of simultaneously  $\Gamma$ -stabilizing controllers for the vertex plants of the bus

The controller guarantees  $\Gamma$ -stability for the vertex plants of the bus, but not necessarily for the whole continuum of all operating conditions. Therefore, a  $\Gamma$ -stability analysis must be made. Fig. 11.32 shows the stability boundaries of the closed-loop system in the  $(\tilde{m}, v)$ -plane. The vertices are  $\Gamma$ -stable and the  $Q$ -box is not intersected by a  $\Gamma$ -stability boundary. Thus, the bus is robustly  $\Gamma$ -stable for the selected controller. The stability analysis has to be followed by simulations, where the design specifications mentioned above are tested in detail.

In a later design study [131] some of the design assumptions have been changed. The control structure in Fig. 2.5 requires three sensors, two for displacement at the front and rear of the bus and the third for the front steering angle  $\delta_f$ . Each of the sensors has to be fail-safe as they are vital parts of the control system. To limit the costs it is desirable to keep the number of sensors as low as possible. Especially, the sensor for the front steering angle  $\delta_f$  causes a significant rise in costs: The sensor itself is a cheap potentiometer, which is mounted under the chassis. There it has to be protected from water, snow, salt, and dust. This is only possible with high expenses compared to the costs of the sensor. For the next design this sensor will be omitted.

The sensor antenna for measuring the rear displacement may have quite large deviations from the guideline, for example when the bus enters a narrow curve. Also the metal car body of the bus changes the electromagnetic field from the guiding wire in the road. Therefore, the rear distance measurement is of poor quality and this sensor is omitted, too. Thus, the only remaining sensor will be the displacement sensor for  $y_s$  at the front

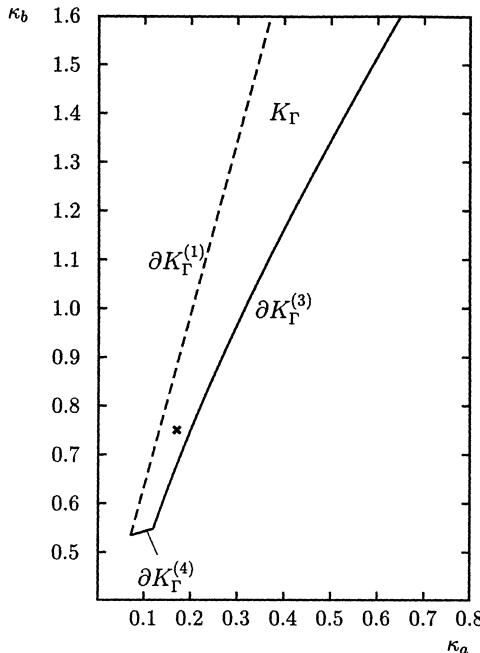


Fig. 11.31: Detailed view of the set of simultaneously  $\Gamma$ -stabilizing low gain controllers

of the bus.

For a new design the power steering of the bus was replaced by an additional integrating hydraulic actuator without position feedback. Thereby, the original power steering for the driver was unmodified for the safety backup by manual steering. The integrating actuator also guarantees zero steady state error under crosswind. The resulting control system is shown in Fig. 11.33, a controller  $g_C(s)$  was designed in [131].

Practical test drives showed that the eigenvalue region bounded by (11.5.2) was too cautious with respect to passengers comfort. The compensator has to be designed to guarantee robust stability for the new eigenvalue region  $\Gamma$  bounded by

$$\left(\frac{\sigma}{0.35}\right)^2 - \left(\frac{\omega}{1.75}\right)^2 = 1 \quad (11.5.7)$$

The bus is now a single-input, single-output system with input  $u$  for the integrating hydraulic actuator and output  $y_s$ , the displacement of the front antenna from the guideline. The transfer function (1.3.7) from actuator input  $u$  to lateral deviation  $y_s$  for the data given in Table 1.3 is

$$\frac{y_s(s)}{u(s)} = \frac{4.803 \cdot 10^{10}v^2 + 3.866 \cdot 10^{11}vs + 6.079 \cdot 10^5mv^2s^2}{s^3(2.690 \cdot 10^{11} + 1.663 \cdot 10^4mv^2 + 9.818 \cdot 10^5mvs + m^2v^2s^2)}$$

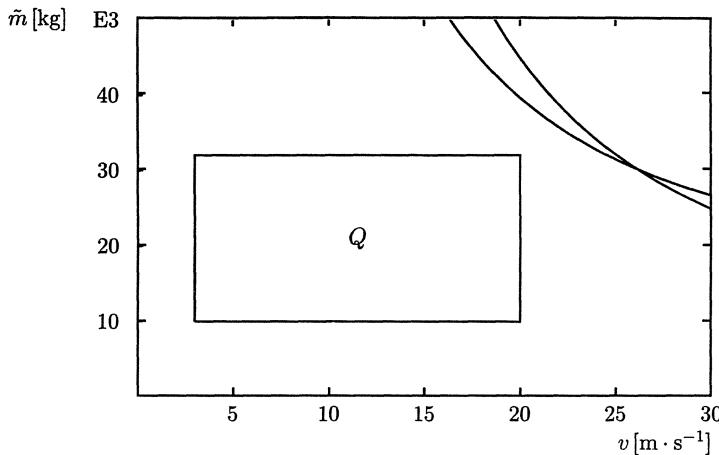
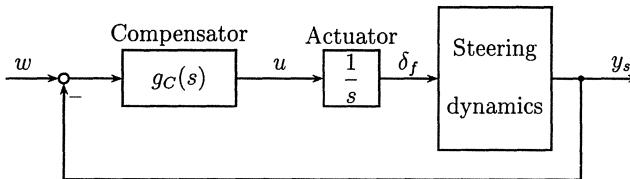
Fig. 11.32:  $\Gamma$ -Stability boundaries in the  $(\tilde{m}, v)$ -plane for the feedback vector (11.5.6)

Fig. 11.33: Track guided bus with sensor for front displacement and integrating actuator

In Chapter 2 a compensator structure (2.3.15) was derived from root locus considerations:

$$g_C(s) = \frac{k_1 + k_2 s + k_3 s^2}{(1 + s/\omega_0 + s^2/\omega_0^2)(1 + s/\omega_0)}$$

These considerations showed that it is essential to place a complex zero pair in the neighborhood of the origin to attract the two branches passing into the right half plane into the  $\Gamma$ -stable region. For the denominator of the compensator transfer function a Butterworth configuration at a larger distance from the origin was assumed. Though this pole configuration is of course not preserved in the closed loop it is a good starting point for the design.

The  $(k_2, k_3)$ -plane is selected for a graphical display of the set of stabilizing controllers. The initial values of  $\omega_0$  and  $k_1$  are chosen as  $\omega_0 = 100$  and  $k_1 = 2.5$ . The  $\Gamma$ -stabilizing sets for the four extremal operating conditions of the bus are shown in Figures 11.34–11.37. An intersecting set can be found easily (Fig. 11.38).

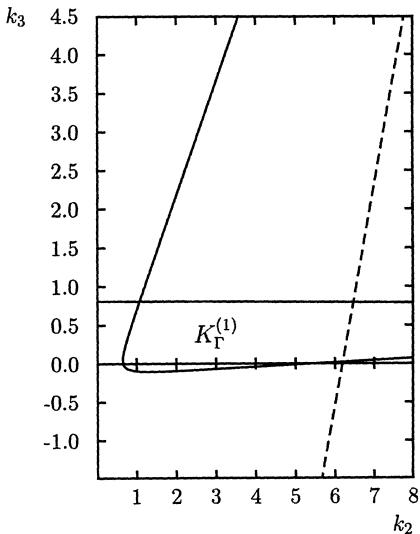


Fig. 11.34: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950 \text{ [kg]}$ ,  $v = 3 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 100$ ,  $k_1 = 2.5$

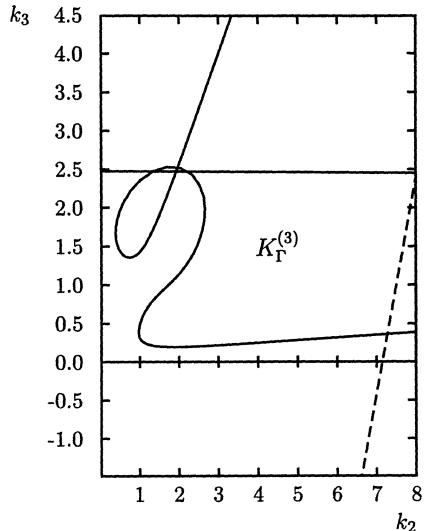


Fig. 11.36: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000 \text{ [kg]}$ ,  $v = 20 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 100$ ,  $k_1 = 2.5$

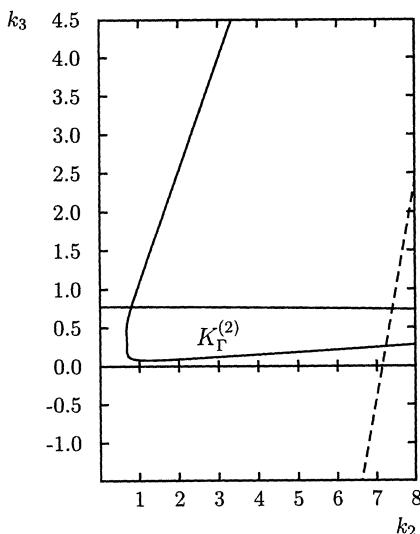


Fig. 11.35: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950 \text{ [kg]}$ ,  $v = 20 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 100$ ,  $k_1 = 2.5$

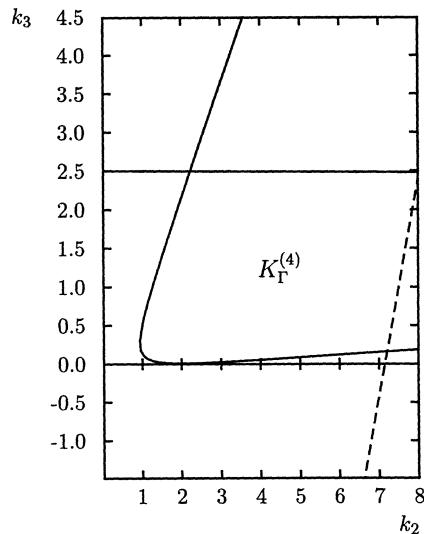


Fig. 11.37: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000 \text{ [kg]}$ ,  $v = 3 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 100$ ,  $k_1 = 2.5$

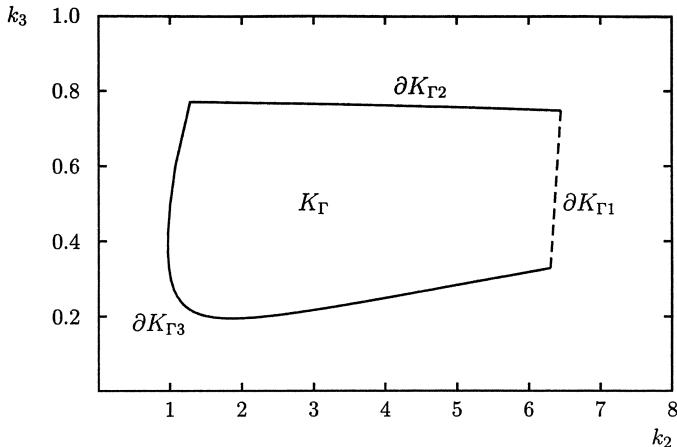


Fig. 11.38: Simultaneously  $\Gamma$ -stabilizing set for the extremal plants for  $\omega_0 = 100$ ,  $k_1 = 2.5$

The initial value for the bandwidth is quite large. The bus would react very fast even to small kinks in the guiding wire or other disturbances, which is not desired. The stabilizing region is large enough and leaves some freedom in changing  $k_1$  and  $\omega_0$ . Starting from  $\omega_0 = 100$  and  $k_1 = 2.5$ , from which we know that a solution can be found, we reduce step by step the bandwidth  $\omega_0$ . During this procedure the parameter  $k_1$  is tuned such that the stabilizing region gets as large as possible. Finally, we reach a point where no further reduction of the bandwidth is possible. The resulting parameters for  $k_1$  and  $\omega_0$  are then  $\omega_0 = 25$  and  $k_1 = 0.6$ . The sets of  $\Gamma$ -stabilizing controllers for the vertex plants are displayed in Figures 11.39–11.42. The intersection of these four sets is shown in Fig. 11.43.

From the set we pick the points  $k_2 = 0.7$  and  $k_3 = 0.15$ . The resulting controller is then

$$F_c(s) = 25^3 \frac{0.6 + 0.7s + 0.15s^2}{(s + 25)(s^2 + 25s + 625)}$$

This controller  $\Gamma$ -stabilizes the plant at least for the extremal operating conditions. In a stability analysis the  $\Gamma$ -stability of this controller is verified for the whole range of operating conditions. The stability boundaries in the  $(v, \tilde{m})$ -plane for this controller were already shown in Fig. 9.4. The vertices are stable and none of the boundaries intersects the  $Q$ -box. A real root boundary passes by very close to the  $Q$ -box. This happens, however, for low velocities and is therefore not critical because a sluggish time response resulting from a real pole close to the origin means only a short driving distance at low speeds.

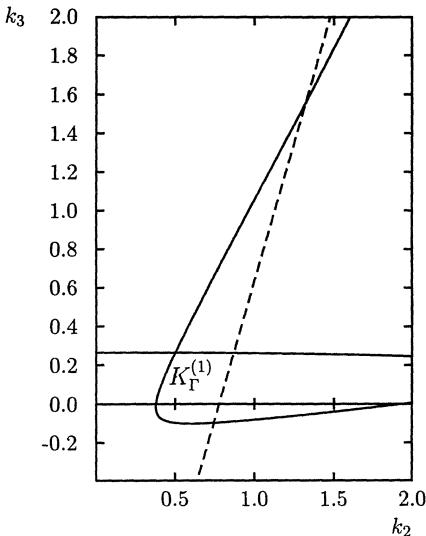


Fig. 11.39: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950 \text{ [kg]}$ ,  $v = 3 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 25$ ,  $k_1 = 0.6$

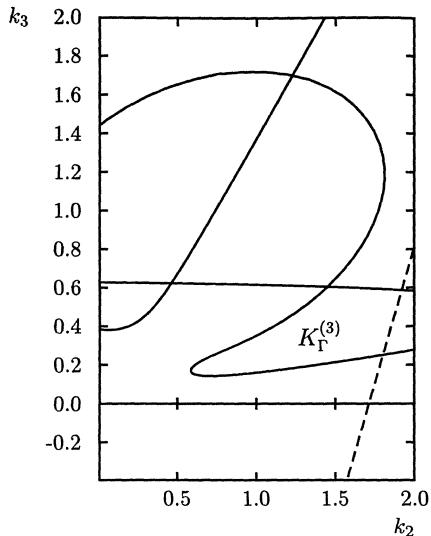


Fig. 11.41: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000 \text{ [kg]}$ ,  $v = 20 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 25$ ,  $k_1 = 0.6$

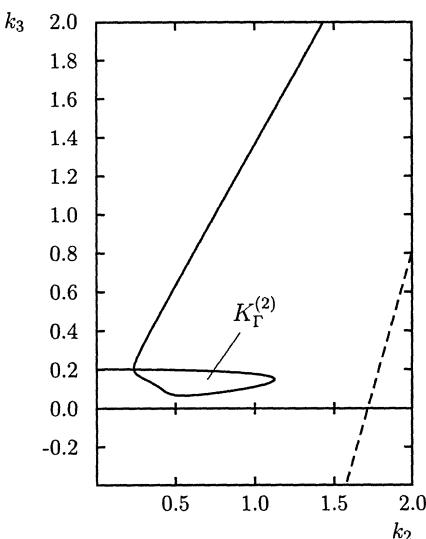


Fig. 11.40: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 9950 \text{ [kg]}$ ,  $v = 20 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 25$ ,  $k_1 = 0.6$

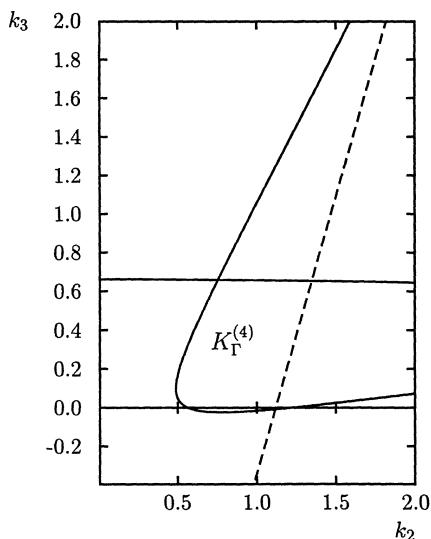


Fig. 11.42: Set of  $\Gamma$ -stabilizing controllers for  $\tilde{m} = 32000 \text{ [kg]}$ ,  $v = 3 \text{ [m} \cdot \text{s}^{-1}\text{]}$ ,  $\omega_0 = 25$ ,  $k_1 = 0.6$

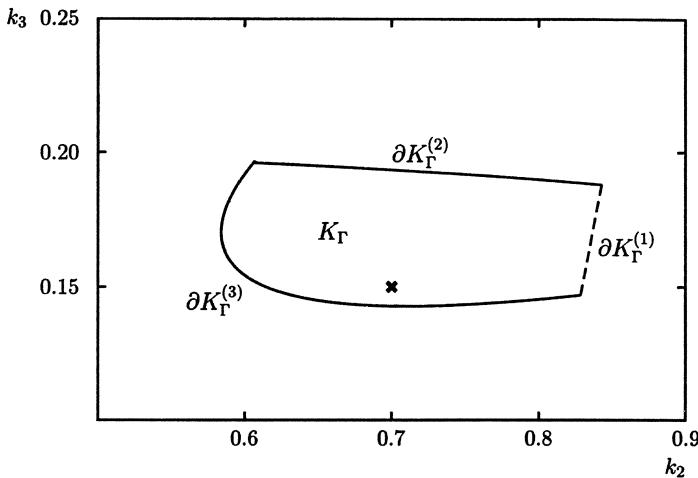


Fig. 11.43: Simultaneously stabilizing set for the extremal plants for  $\omega_0 = 25$ ,  $k_1 = 0.6$

The controller has to be tested now in simulations where the design specifications are checked. A simulation of the transition from manual to automatic steering is shown in Fig. 11.44. This case is especially critical because here the largest deviations from

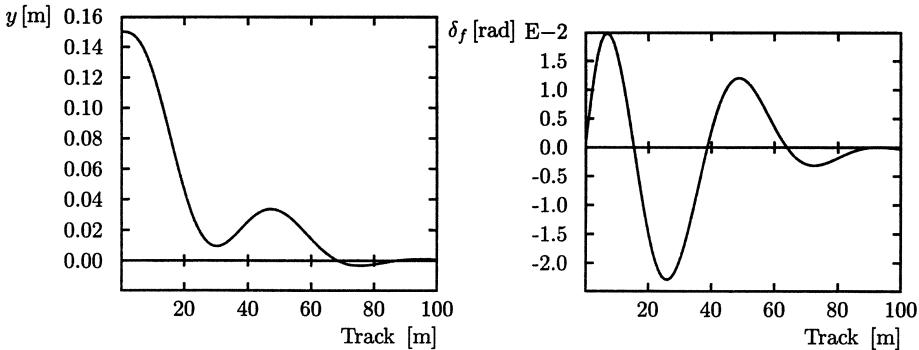


Fig. 11.44: Simulation of a transition from manual to automatic steering for  $v = 20 [\text{m} \cdot \text{s}^{-1}]$  and  $\bar{m} = 32000 [\text{kg}]$

the guideline may occur. Additionally, this is the only operating case where a step input appears. The controller is not started from a zero initial condition, instead, the controller is connected to the plant before switching. Then, only the controller output is switched to the actuator. Thus, no differentiation of a step input occurs. All other inputs from the guideline are smooth, for example if the bus enters a curve. The simulation was made for an initial deviation of 15 [cm]. At  $t = 0$  the driver switches to automatic steering. Also in other cases like crosswind, entering a curve, etc., the controller shows good behavior.

*Remark 11.1.* In Chapter 2 it was shown that structural decoupling for vehicles is possible under some mild assumptions. The control law was given in (2.5.4): The yaw rate  $r$  has to be fed back to the front steering angle  $\delta_f$ . In the ideal case a pole zero cancellation in the transfer function from steering wheel input to the lateral acceleration at the front axle occurs. This simplifies the bus dynamics.

The bus O 305 violates the necessary assumptions. However, with yaw rate feedback an almost cancellation of a pole pair occurs, which has therefore little influence on the bus dynamics. A controller design in [17] for the track guided bus with additional yaw rate feedback showed significant improvement.  $\square$

## 11.6 Case Study: Flight Control

The robust control problem was introduced in Section 1.4.

### *Robustness with Respect to Flight Conditions*

Based on pilot ratings admissible intervals for damping and frequency of the short period mode have been determined and must be verified in the qualification tests for the certification of a new aircraft [1]. For the characteristic polynomial  $p(s) = a_0 + a_1 s + s^2 = \omega_0^2 + 2D\omega_0 s + s^2$  the following bounds are prescribed for damping  $D$  and natural frequency  $\omega_0$

$$0.35 \leq D \leq 1.3; \quad \omega_a \leq \omega_0 \leq \omega_b \quad (11.6.1)$$

The bounds  $\omega_a$  and  $\omega_b$  depend on the flight condition and are given in Table 11.2.

Flight Condition	Velocity (Mach)	Altitude (Feet)	$\omega_a$	$\omega_b$
			(Radian / Second)	(Radian / Second)
1	0.50	5000	2.02	7.23
2	0.85	5000	3.50	12.60
3	0.90	35000	2.19	7.86
4	1.50	35000	3.29	11.80

Table 11.2: Military specifications for the admissible natural frequency  $\omega_0$  of the short period mode of an aircraft

Comparing the eigenvalue distances from the origin for the open-loop poles in Table 1.2 and the closed-loop pole specifications in Table 11.2 it is obvious that the second basic rule of robust control is observed: *Keep a fast system fast and keep a slow system slow.*

The admissible region is defined in the plane of coefficients  $a_0, a_1$  by (11.6.1). It is bounded by the lines  $a_0 = \omega_a^2$  and  $a_0 = \omega_b^2$  as well as the two parabolas  $a_0 = \omega_0^2$ ,  $a_1 = 2D_{min}\omega_0 = 0.7\omega_0$  and  $a_0 = \omega_0^2$ ,  $a_1 = 2D_{max}\omega_0 = 2.6\omega_0$ . The region for flight

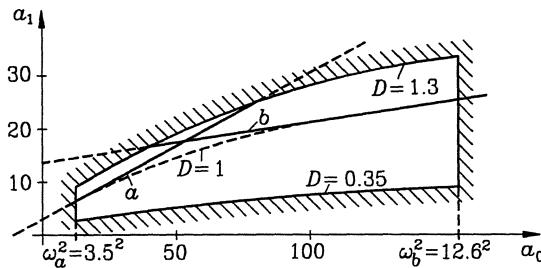


Fig. 11.45: Required stability region for flight condition 2 in coefficient plane and replacement of the boundary  $D = 1.3$  by the two boundaries  $a$  and  $b$

condition 2 is represented in Fig. 11.45. If  $D$  is increased beyond one, then the complex pole pair in the  $s$ -plane unites at a branching point for  $D = 1$  and then separates into two real poles of which one moves to the left and the other to the right as  $D$  increases. The pole moving to the right often leads to an undesired decrease in the bandwidth of the closed loop; this corresponds in the time domain to a sluggish response. These less desirable domains lie above the parabola for  $D = 1$  in Fig. 11.45. Therefore, the admissible region in coefficient space will be contracted by replacing the upper bound for  $a_1$  by the straight lines  $a$  and  $b$  (the tangents of the curve  $D = 1$  at  $\omega_a^2 = 3.5^2$  and  $\omega_b^2 = 12.6^2$ ). The coefficient region between  $a$ ,  $b$  and the curve  $D = 1$  describes exactly the polynomials with a pair of real roots in the interval  $s \in [-\omega_a; -\omega_b]$ . The reduced coefficient region corresponds to the “pineapple segment”, which is displayed in Fig. 11.46 with solid lines. The expanded region shown in dashed lines is decisive for further eigenvalues, which do not belong to the short period mode, see Chapter 3. The values  $\omega_a$ ,  $\omega_b$  and the open loop eigenvalues 1, 2, and 3 are displayed in Fig. 11.46 for flight condition 2.

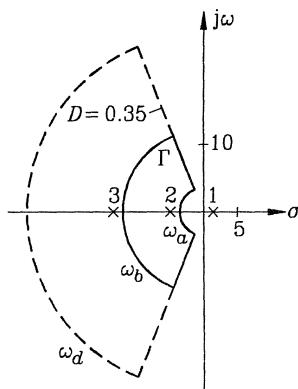


Fig. 11.46:  $\Gamma$ -stability region in the  $s$ -plane

As already discussed in Chapter 2, the controller structure which is easiest to implement

is

$$u = - \begin{bmatrix} k_{n_z} & k_q & 0 \end{bmatrix} x = -\mathbf{k}^T x \quad (11.6.2)$$

where  $x_1$  is measured by an accelerometer and  $x_2$  with a gyro. The resulting two-dimensional cutting plane through the three-dimensional region of  $\Gamma$ -stability is shown in Fig. 11.47 for  $\omega_d = 70$  radians/second and flight condition 2.

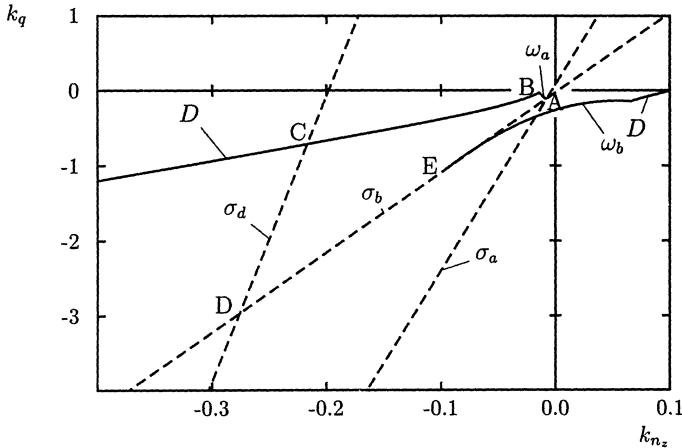


Fig. 11.47: Two-dimensional cross section of the  $\Gamma$ -stability region in  $\mathbf{k}$ -space

For  $\mathbf{k} = 0$  the eigenvalues are those of the open loop, see 1, 2, and 3, in the  $s$ -plane figure. If the boundary  $\sigma_a$  in  $\mathbf{k}$ -space is crossed by variation of  $\mathbf{k}$  starting with  $\mathbf{k} = 0$ , then eigenvalue 2 moves across the point  $\sigma_a$  to the right.

The eigenvalue pair (1, 2) which describes the short period longitudinal mode, now moves as a complex conjugate pole pair into the desired region if the boundaries  $\omega_a$  or  $D$  are crossed. The desired region is thus ABCDEA. On AB the eigenvalues 1 and 2 lie on the circle  $\omega_a$ , on BC they lie on  $D$ , and on CD eigenvalue 3 lies at  $\sigma_d$ . On DE one of the eigenvalues (1, 2) lies at  $\sigma_b$ , on EA eigenvalue 3 lies at  $\sigma_a$ .

*Remark 11.2.* A one-to-one correspondence of the eigenvalues of the closed loop to those of the open loop is preserved on the described path in the  $(k_q, k_{n_z})$ -plane. Such a relation can be lost, for example, if eigenvalues 2 and 3 combine to form a complex pair, then a branching point is crossed. This would be the case, for example, if after leaving  $\mathbf{k} = 0$ , first  $\sigma_b$  and subsequently  $\omega_b$  were crossed. Pole 3 then moves to the right first and the pair (2, 3) moves over the boundary  $\omega_b$  in the  $s$ -plane.  $\square$

Correspondingly, the  $\Gamma$ -stability regions are determined for the other three flight conditions. All four regions intersect in the region represented in Fig. 11.48. Thus, the assumed controller structure leads to an admissible solution set.

Fig. 11.48 provides the design engineer with essential information as to which demands are critical in which flight cases. The flight condition numbers are used as indices at the boundary line names. The stimulation of structural oscillations is especially critical

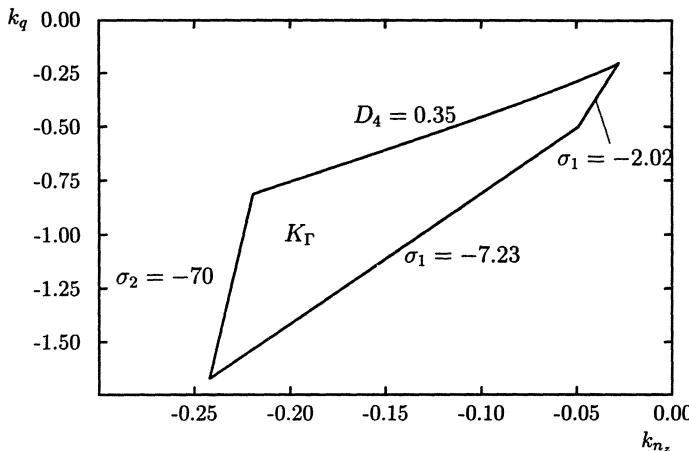


Fig. 11.48: Intersection of the regions of  $\Gamma$ -stability for all four flight conditions

near the boundary  $\sigma_2 = -70$ , i.e. for flight condition 2 (high velocity, low-level). Insufficient damping is critical near the boundary  $D_4$  for flight condition 4 (fast flight at high altitude). The other two boundaries originate from flight condition 1 (landing approach). Near the boundary  $\sigma_1 = -2.02$  a real pole moves toward the origin. A real pole would cross over to the left at the boundary  $\sigma_1 = -7.23$ . Flight condition 3 is not critical in this controller structure.

Information on the possible demands that can be met by a control system and the possibilities for compromise is usually more helpful for the control engineer than an optimization process, which requires that the trade-offs for all conceivable conflicts are decided beforehand by the choice of weightings in a performance criterion.

Fig. 11.48 suggests different possibilities for coming closer to the selection of a point from the solution set. One possibility is to reduce the eigenvalue region  $\Gamma$  of Fig. 11.46. Fig. 11.49 shows the tightened region obtained by a reduction of  $\omega_d$  from 70 to 50 radians/second, an increase of the minimum damping from 0.35 to 0.5 and an increase of all minimum natural frequencies  $\omega_a$  by 50%. There still exists an intersection for the four flight conditions and the designer may decide which of these specifications he wants to tighten even more and thereby narrow the admissible region further.

However, other requirements can also be incorporated into the solution choice which have not been taken into consideration so far. As an example, suppose the limitation of the elevator deflection  $x_3$  and its derivative  $\dot{x}_3 = -14x_3 + 14k^T x$  is essential. Then, a solution with a smaller loop gain must be chosen as explained in Section 11.4. In order to illustrate this effect, the  $c^*$ -response to a unit step input given by the pilot has been calculated, as well as the required elevator deflection  $x_3$  for the points  $g_1$ ,  $g_2$ , and  $g_3$ , in Fig. 11.49, see Fig. 11.51.

$$c^* = (n_z + 12.43q)/c_\infty$$

is a common variable in flight mechanics for the evaluation of the step response. The stationary value  $c_\infty$  is used for normalization. The  $c^*$ -step response should lie in the

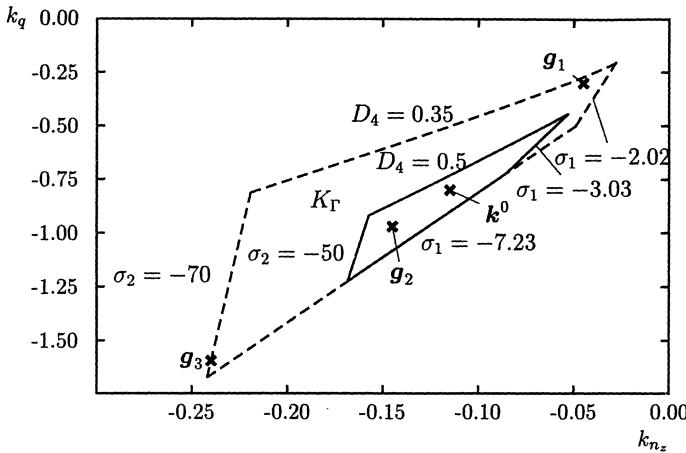


Fig. 11.49: Tightened specifications reduce the admissible solution set

envelope shown in Fig. 11.50. If the eigenvalues are suitably located in the eigenvalue region  $\Gamma$ , then the step responses are slightly faster than required. By a small delay in a prefilter they can be modified such that they fit into the envelope in Fig. 11.50. For this design study the prefilter  $1/(s + 6)$  was given.

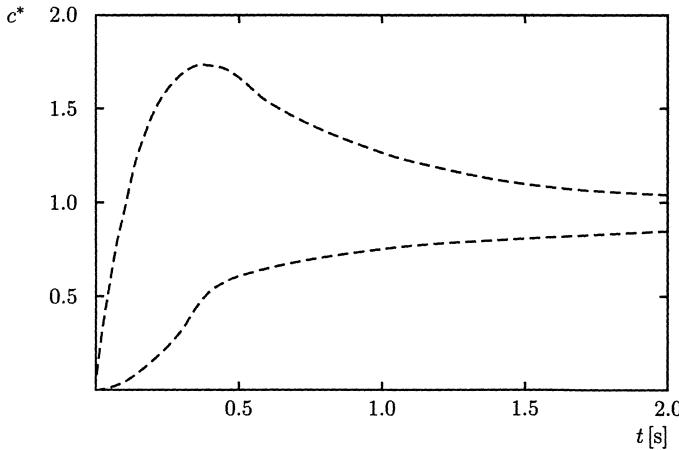


Fig. 11.50: Admissible envelope for the  $c^*$ -response

The elevator deflection is decreased significantly for a small loop gain  $\mathbf{g}_1$  and the step response is correspondingly slow. A high loop gain  $\mathbf{g}_3$  is unsatisfactory due to too much overshoot of the step response, greater amplitude, and extension of the bandwidth to the vicinity of the structural-oscillation frequency. The medium loop gain  $\mathbf{g}_2$  is the best in this comparison, it produces a solution which is closer to  $\mathbf{g}_3$  than to  $\mathbf{g}_1$  and a point

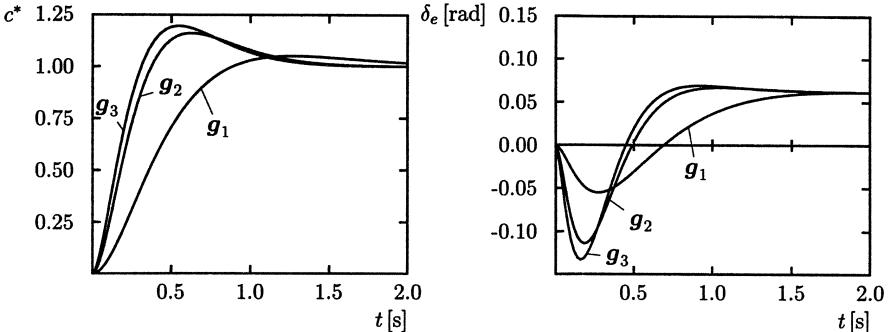


Fig. 11.51:  $c^*$  response and elevator deflection  $\delta_e$  for low ( $g_1$ ), medium ( $g_2$ ) and high ( $g_3$ ) loop gain. The elevator deflection is normalized to the stationary value of  $-\delta_e(t, g_1)$ .

$k^0$  slightly closer to  $g_1$  was finally selected. The controller is

$$u = -k^{0T} x = \begin{bmatrix} -0.115 & -0.8 & 0 \end{bmatrix} x \quad (11.6.3)$$

$k^0$  is indicated in Fig. 11.49. Calculating and testing the eigenvalues shows that they lie in their respective prescribed regions for all flight conditions.

#### Robustness against Sensor Failures

In a further refinement of the design the requirement of robustness with respect to gyro or accelerometer failure is introduced. Fig. 11.48 shows that the  $\Gamma$ -stability region does not intersect the axes  $k_{n_z}$  and  $k_q$ . Therefore, one of the two feedbacks alone is not sufficient.

A first attempt would be to use two parallel gyros and two parallel accelerometers. From Fig. 11.48, it is seen that there are no points in the  $\Gamma$ -stability region which admit 50% gain reduction of both  $k_{n_z}$  and  $k_q$ . Only emergency requirements (level 3 of the flying qualities [1]) can be achieved [68]. Interestingly, the shape of  $K_\Gamma$  in Fig. 11.48 with its lengthy extension away from the origin is such that the flying qualities are improved if a second failure in the other channel occurs, i.e. for a simultaneous 50% reduction of gyro and accelerometer gain. This indicates that a controller structure as in Fig. 11.23 with only gyros or with only accelerometers is advantageous. Here, any gain reduction occurs in both feedback gains simultaneously.

Of course the unmeasured variable cannot be really reconstructed for all four flight conditions simultaneously by a constant filter. Also the system order is increased by the filter such that additional eigenvalues occur. Therefore, the new stability region in the plane of the two feedback gains must be established. A similar form of the stability region can be expected, at least if the filter is chosen such that it yields an approximation for the unmeasured variable.

In deciding whether to use only gyros or only accelerometer we consider the zeros of the two transfer functions. In the four flight conditions the zeros in the accelerometer transfer function vary between  $-0.4 \pm j5.7$  and  $-0.9 \pm j9.1$ . They are minimum phase but outside the  $\Gamma$ -stability region. Also they may vary widely such that an approximate

cancellation by the polynomial of averaged zeros  $s^2 + 1.172s + 49.9$  is not advisable. This would generate closed-loop poles which are not  $\Gamma$ -stable, possibly not even stable.

The gyro transfer function has one real zero varying between  $-0.64$  and  $-1.57$ . Here, an approximate cancellation by the polynomial of averaged zeros  $s + 0.98$  is no problem. By the almost cancellation, this filter pole will be only weakly controllable from  $u$  and will not be shifted much by closing the loop. Also, it is weakly observable from the pitch rate and can be exempted from the pole region requirement.

Both transfer functions from  $u$  to  $n_z = x_1$  and  $q = x_2$  have the same poles; one pole is unstable in subsonic flight. The filter must replace the numerator of the gyro transfer function by the numerator of the accelerometer transfer function. Taking the averaged zeros of the two transfer function the filter is

$$f(s) = a \cdot \frac{s^2 + 1.172s + 49.9}{s + 0.98} \cdot \frac{10}{s + 10} \quad (11.6.4)$$

The term  $10/(s + 10)$  was introduced to make the filter realizable. The gain ratio  $a$  varies between 0.527 and 0.577 in the four flight conditions and was chosen at the average value  $a = 0.543$ .

By introducing the filter, the system order is raised to  $n = 5$ . The chosen controller structure of Fig. 11.23 defines a plane of the free controller parameters  $\bar{k}_{n_z}$  and  $\bar{k}_q$ . In this plane the intersection of  $\Gamma$ -stability regions for the four flight conditions must be analyzed. It is represented in Fig. 11.52. The allowable region is clearly different

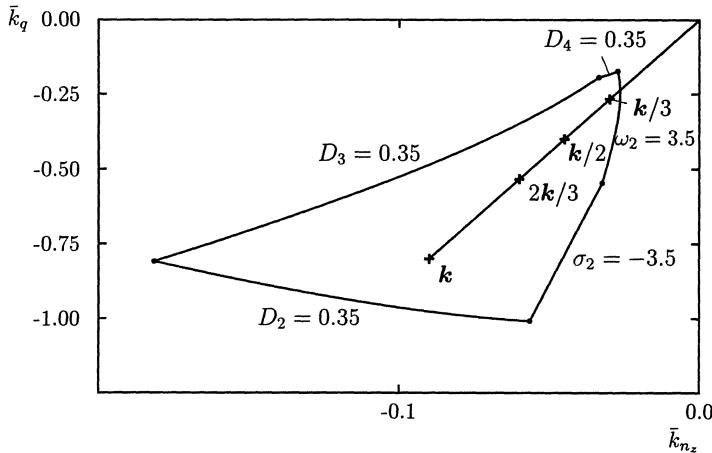


Fig. 11.52: Intersection of the  $\Gamma$ -stability regions for the four flight conditions for the controller structure using only gyros and a feedback filter

from the region in Fig. 11.48. Also flight condition 3 instead of flight condition 1 now contributes to the boundary. Nevertheless, it confirms our expectation that a large simultaneous gain-reduction margin is now achievable, in the extreme case 80%. If one chooses the indicated point  $k = [-0.09 \quad -0.8 \quad 0]^T$ , then the points  $2k/3$ ,  $k/2$ , and  $k/3$ , lie well in the admissible region. Therefore, the system can handle the failure of

two of the three gyros, even if the second failure occurs before the first one is detected. For the  $c^*$ -step responses in Fig. 11.53 it is also worth remarking how little the step response is changed in all four flight conditions if  $\mathbf{k}$  (steeply climbing curve) is reduced to  $\mathbf{k}/2$ . If the two feedback paths via  $\bar{k}_q$  and  $\bar{k}_{n_z}$  are combined, then the controller

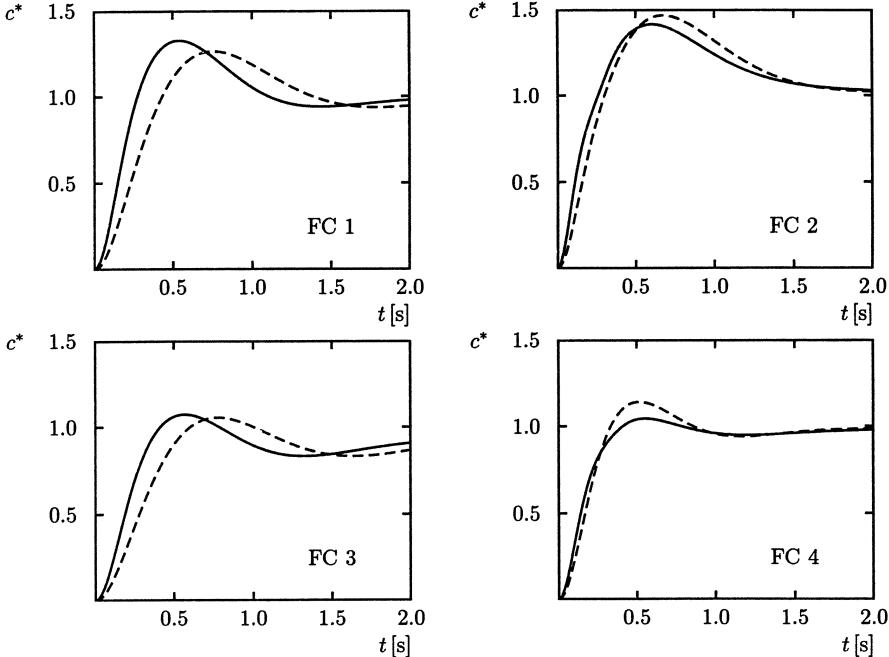


Fig. 11.53:  $c^*$ -step response in four flight conditions. Each case shows nominal and 50% reduced loop gains. The nominal gain results in the steeper initial rise.

$$\begin{aligned} \frac{-u_S(s)}{x_{2S}(s)} &= -0.8 - 0.09 \cdot 0.543 \cdot \frac{s^2 + 1.172s + 49.9}{s + 0.98} \cdot \frac{10}{s + 10} \\ &= -\frac{1.29s^2 + 9.36s + 32.23}{s^2 + 10.98s + 9.8} \end{aligned} \quad (11.6.5)$$

is obtained. The minus sign is explained by the common flight mechanics definitions of the sign of the elevator deflection  $\delta_e$  and the pitch angle  $q$  which causes a minus sign of the plant transfer function.

The procedure for the design of robust stabilization for the F4-E is certainly not a general design recipe, but similar results can be expected for other aircrafts. In the robust design of a back-up controller for the Swedish fighter JAS 39 it was assumed from the beginning that only the gyro is used for feedback. The controller in this case was assumed as a proportional channel plus two channels with first order delay. A robust level 1 controller for 10 flight conditions was designed using both the parameter space technique and the vector optimization described in Chapter 12 [80].

This extensive F4-E example shows how the design tool of two-dimensional cross sections through regions of  $\Gamma$ -stability is applied and combined with other design considerations. The controller structure was developed in steps. The design would be less transparent if the second order controller of (11.6.5) with five free controller parameters had been assumed from the beginning.

## 11.7 Exercises

- 11.1. Consider the crane with uncertain load mass  $m_L \in [0; 5000] [\text{kg}]$ . All other parameters are constant with  $m_C = 1000 [\text{kg}]$ ,  $\ell = 5 [\text{m}]$ , and  $g = 10 [\text{m} \cdot \text{s}^{-1}]$ . The crane is controlled by state feedback

$$u = - \begin{bmatrix} k_1 & k_2 & 0 & 0 \end{bmatrix} \boldsymbol{x}$$

Determine the set of simultaneously  $\Gamma$ -stabilizing controller parameters for the two extremal operating conditions such that the closed-loop eigenvalues lie to the left of the line

$$\partial\Gamma = \{-0.25 + j\omega \mid \omega \in [0; \infty]\}$$

Select a controller and check  $\Gamma$ -stability for the entire interval of  $m_L$ . What is the maximal possible variation in the rope length for the selected controller such that the system still remains  $\Gamma$ -stable?

- 11.2. Consider the crane with  $m_C = 1000 [\text{kg}]$ ,  $\ell = 10 [\text{m}]$ ,  $g = 10 [\text{m} \cdot \text{s}^{-1}]$ , and uncertain load mass  $m_L \in [50; m_{L,\max}] [\text{kg}]$ . The crane is controlled by state feedback

$$u = - \begin{bmatrix} 500 & k_2 & k_3 & 0 \end{bmatrix} \boldsymbol{x}$$

What is the maximal load mass  $m_{L,\max}$  for which the two extremal plants still can be simultaneously  $\Gamma$ -stabilized, where  $\Gamma$  is the region to left of the hyperbola

$$\partial\Gamma = \{\sigma + j\omega \mid \omega^2 = 4\sigma^2 - 0.25, \sigma < -0.25\}$$

What are the controller coefficients then?

- 11.3. Consider the crane with  $\ell \in [8; 16] [\text{m}]$ ,  $m_L \in [1000; 2000] [\text{kg}]$ ,  $m_C = 1000 [\text{kg}]$ , and  $g = 10 [\text{m} \cdot \text{s}^{-2}]$ . The system has to be stabilized by state feedback

$$u = - \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix} \boldsymbol{x}$$

Design a controller in an invariance plane such that the closed-loop eigenvalues lie to the left of the hyperbola (11.2.3) for the entire operating domain. Try to find a feedback vector with  $k_4 = 0$ .

- 11.4. Consider the system

$$\dot{\mathbf{x}} = \begin{bmatrix} q & q+2 \\ 2 & 3-2q \end{bmatrix} \mathbf{x} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

which is controlled by state feedback

$$u = - \begin{bmatrix} k_1 & k_2 \end{bmatrix} \mathbf{x} + w$$

The uncertain parameter  $q$  varies in the interval  $q \in [1; 5]$ . The eigenvalues of the closed-loop system should lie to the left of the pair of straight lines with constant damping  $\omega < -5\sigma$ . Determine the set of  $\Gamma$ -stabilizing controllers for the two vertex plants. What is the maximal damping for which still simultaneous  $\Gamma$ -stabilization of both plants can be achieved?

- 11.5. Consider the track guided bus O 305 with the control structure given in Fig. 11.33. The bus is additionally equipped with a gyro and the yaw rate is fed back to the hydraulic actuator input with

$$\dot{\delta}_f = u - 0.89r$$

Design a controller which  $\Gamma$ -stabilizes the bus for all operating conditions  $\tilde{m} \in [9950; 32000] [\text{kg}]$  and  $v \in [3; 20] [\text{m} \cdot \text{s}^{-1}]$ . Verify the specifications given in Section 11.5 with simulations of the maneuvers switching from manual to automatic steering and entering a circular arc. A critical maneuver is the entering of a narrow bus stop bay at minimal speed  $v = 3 [\text{m} \cdot \text{s}^{-1}]$ . The reference input is given in Fig. 11.54. Redesign your controller if it does not meet the specifications for

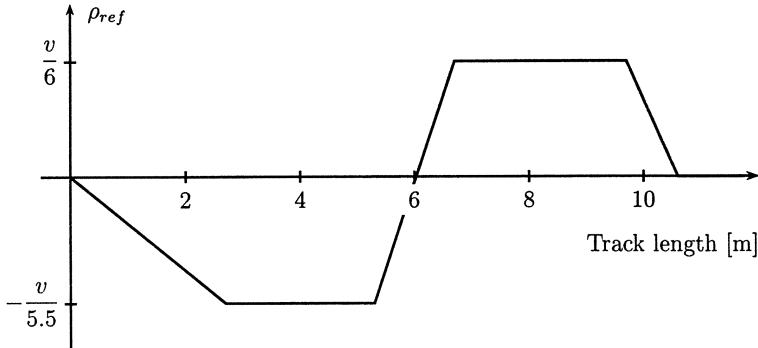


Fig. 11.54: Reference input  $\rho_{ref}$  for entering a bus stop bay

this maneuver. If you succeed, then mail your solution to the authors.

## 12 Design by Optimizing a Vector Performance Index

When applying the parameter space design method of the preceding chapter, regions of controller coefficients are first determined, guaranteeing simultaneous  $\Gamma$ -stability of a finite plant family. The design method of [111, 110], which will be presented in this chapter, also searches for simultaneously stabilizing controller coefficients. However, in contrast to the parameter space method no stability boundaries in the space of controller coefficients are generated. Rather, the coefficients are determined by optimizing a vector performance index, the components of which rate different design specifications.

The controller design procedure progresses iteratively. Free controller coefficients in a given controller structure are determined in such a way that a systematic improvement of the design result in each iteration step is guaranteed, even if a great many of design specifications have to be considered. For each iteration step free design parameters are chosen in a highly systematic policy: Those components of the performance vector, which shall be small, are reduced without violating predefined upper bounds for the remaining performance components. By this policy one is able to force the design result in each step in a desired direction until finally a best compromise is found.

Starting the design with controller coefficients already  $\Gamma$ -stabilizing a given plant family (for instance determined by the parameter space design method) any additional design requirement can be included by a corresponding criterion to improve it in further design iterations, without loosing the simultaneous  $\Gamma$ -stability property. Thus, from the beginning we move in the admissible solution set of  $\Gamma$ -stabilizing controller coefficients (without necessarily knowing its boundary). But the design can also be started with controller coefficients outside the  $\Gamma$ -stability set. Then  $\Gamma$ -stability criteria for the given plant family have to be reduced first, until the controller coefficient result is contained in the  $\Gamma$ -stabilizing set.

In the first section it will be explained how system specifications are included into the design. In the second section it is shown how in successive optimization steps the design result can be improved systematically. The design procedure is then demonstrated for a car steering problem.

We start now with the multi-model problem. For that we assume that a finite set of operating points

$$\mathbf{q}^{(1)}, \mathbf{q}^{(2)}, \dots, \mathbf{q}^{(N)} \in Q, \quad \mathbf{q}^{(i)} = [q_1^{(i)} \dots q_\ell^{(i)}]^T \quad (12.0.1)$$

which can be considered as relevant representatives for the system behavior over the complete parameter continuum  $Q$ , have already been chosen, see Fig. 12.1.

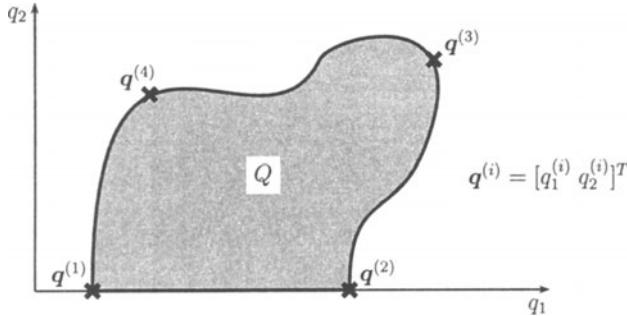


Fig. 12.1: Chosen relevant operating points for the multi-model problem

Furthermore we suppose that some principal considerations about the given plant have already led to a suitable controller structure. Examples for the choice of such a controller structure are given in Chapter 2. It remains the problem to determine the free coefficients

$$\mathbf{k} = [k_1 \dots k_m]^T \quad (12.0.2)$$

in the chosen controller structure in such a way that given specifications for the discrete operating points  $\mathbf{q}^{(i)}, i = 1, \dots, N$  are satisfied simultaneously, see Fig. 12.2. The coefficients  $k_i$  represent poles and zeros of controller transfer functions or coefficients of controller matrices.

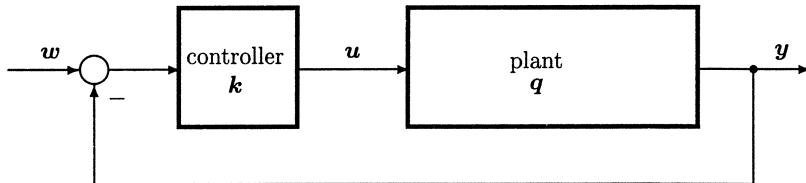


Fig. 12.2: Closed loop system with free controller coefficients  $\mathbf{k}$  and uncertain plant parameters  $\mathbf{q}$

## 12.1 Formulation of Design Specifications for a Finite Plant Family

For arbitrary but fixed values of the controller coefficients  $\mathbf{k}$ , properties of the closed loop system behavior can be analyzed by known numerical methods (representative

simulations for tracking and disturbance behavior, e.g. step- or initial value responses, computation of frequency responses, eigenvalues etc.). Such a system analysis allows for qualitative judgement of the system performance relative to the individual design specifications and in addition to quantify the system performance by a set of performance criteria.

For this purpose each design specification is associated with a performance criterion  $c_i$  in such a way that its value is always nonnegative and decreases with an improvement of the respective design specification,

$$c_i = c_i(\mathbf{k}) \geq 0 \quad (12.1.1)$$

$$c_i \text{ decreases with better satisfaction of the respective design specification} \quad (12.1.2)$$

In most cases there are several possibilities of associating a criterion to a given design specification.

#### *Eigenvalue specifications*

For robust control, stability of the closed loop system in each representative operating point  $\mathbf{q}^{(i)} \in Q, i = 1, \dots, N$  is a design specification which necessarily has to be satisfied. Transforming these stability requirements for the chosen operating points  $\mathbf{q}^{(i)}$  to criteria with the above mentioned properties (12.1.1), (12.1.2) is simple and can for instance be done by the following expressions,

$$c_i(\mathbf{k}) = \begin{cases} 1/[1 - \max_j [\operatorname{Re} s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))]] & \text{for } \max_j \operatorname{Re} s_j < 0 \\ 1 + \max_j [\operatorname{Re} s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))] & \text{otherwise} \end{cases}, \quad i = 1, \dots, N \quad (12.1.3)$$

Here  $s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))$  denotes an eigenvalue  $s_j$  of the closed loop system matrix  $\mathbf{A}_c^{(i)}(\mathbf{k}) = \mathbf{A}_c(\mathbf{q}^{(i)}, \mathbf{k})$  for operating point  $\mathbf{q}^{(i)}$ . Clearly the criterion has a value always greater than zero and is smaller the further the right most eigenvalue lies to the left. For simultaneous (Hurwitz-) stability, controller coefficients have to be determined such that all  $N$  performance criteria have value less than 1. The smaller a performance value below 1 the better is the stability margin w.r.t. the imaginary axis for the respective operating point.

An additional minimal damping requirement over all eigenvalues in each operating point  $\mathbf{q}^{(i)}, i = 1, \dots, N$  can be formulated by the criteria

$$c_{i+N}(\mathbf{k}) = 1 - \min_j \{D_j(\mathbf{A}_c^{(i)}(\mathbf{k}))\}, \quad i = 1, \dots, N \quad (12.1.4)$$

with

$$D_j(\mathbf{A}_c^{(i)}(\mathbf{k})) = \operatorname{Re} s_j(\mathbf{A}_c^{(i)}(\mathbf{k})) / |s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))|, \quad s_j \neq 0 \quad (12.1.5)$$

For a stable operating point  $\mathbf{q}^{(i)}$  the value of such a criterion is between 0 and 1, for an unstable operating point it is between 1 and 2. The lower the minimal damping constant  $D$  over all eigenvalues for a stable operating point the larger is the respective criterion value.

An additional requirement for the eigenvalues to lie all within a circle with the origin of the complex plane as its center can easily be formulated by the criteria

$$c_{i+2N}(\mathbf{k}) = \max_j |s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))|, \quad i = 1, \dots, N \quad (12.1.6)$$

Combining different types of eigenvalue criteria allows for the expression of requirements for eigenvalues to lie within special pole regions ( $\Gamma$ -stability specifications). If for instance simultaneous  $\Gamma$ -stability for  $N$  chosen discrete operating points  $q^{(i)} \in Q$  is required for a  $\Gamma$  as in Fig. 12.3 we must find controller coefficients  $k$  such that for each

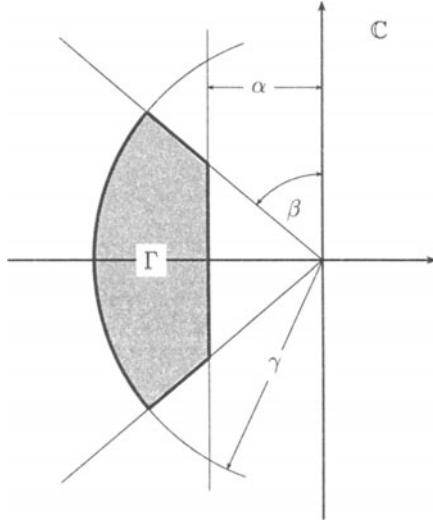


Fig. 12.3: Stability region  $\Gamma$

operating point  $q^{(i)}$  the following inequalities are satisfied,

$$\begin{aligned} c_i(k) &\leq \bar{d}_i := 1/(1 - \alpha) \\ c_{i+N}(k) &\leq \bar{d}_{i+N} := 1 - \sin(\beta) \\ c_{i+2N}(k) &\leq \bar{d}_{i+2N} := \gamma, \quad i = 1, \dots, N \end{aligned} \tag{12.1.7}$$

(The constants  $\alpha$ ,  $\beta$ , and  $\gamma$  could also depend on the operating point  $q^{(i)}$  that means different stability regions  $\Gamma^{(i)}$  could be chosen for different operating conditions. An example is the flight control problem of Section 11.7)

*Remark 12.1.* For sampled-data systems a family of circles is defined in (10.3.2) and illustrated in Fig. 10.5. For this case we choose

$$\begin{aligned} c_i(k) &= \max_j |z_j(A_c^{(i)}(k)) - \tau_0| \\ c_i(k) &\leq \bar{d}_i := r \end{aligned}$$

Note that the open-loop eigenvalues for all operating conditions satisfy  $c_i < 1 + \sqrt{2}$  by appropriate choice of the sampling interval, see Section 10.3.  $\square$

The vector

$$\bar{d} := [\bar{d}_1, \dots, \bar{d}_N, \bar{d}_{N+1}, \dots, \bar{d}_{2N}, \bar{d}_{2N+1}, \dots, \bar{d}_{3N}]^T \tag{12.1.8}$$

is called level of requirement. If we are satisfied with simultaneous  $\Gamma$ -stability for the  $N$  discrete operating points  $\mathbf{q}^{(1)}, \dots, \mathbf{q}^{(N)} \in Q$  then we have already completed the mathematical formulation of our design specifications by the above  $3N$  performance criteria. In general, however, further design specifications for the  $N$  operating points  $\mathbf{q}^{(i)} \in Q$  have to be satisfied. (We assume again that the continuum  $Q$  of operating conditions is sufficiently represented by the  $N$  operating points.)

Usually, design specifications can roughly be divided into specifications concerning control performance and into specifications concerning cost of control. Control performance specifications include requirements for the tracking- and disturbance behavior where the transient and the stationary behavior will be distinguished. Frequently time domain specifications are formulated by attributes of responses to specific test functions like a step input. Such an attribute can be the rise-time or overshoot or more generally the dynamic or the stationary accuracy. Cost of control specifications include for instance constraints on maximal values for actuator signals and their rate of change. A low feedback gain at high frequencies and a simple controller structure are required in most cases. Further specifications can be added, for instance, state variable constraints, tolerances for the gain of certain frequency transfer functions, etc..

#### *Time domain specifications*

For formulating specifications w.r.t. the dynamic time behavior, performance criteria of the type

$$\int_{t_1}^{t_2} [y(t, \mathbf{k}) - y_M(t)]^2 dt / (t_2 - t_1) \quad (12.1.9)$$

are of great importance, see Fig. 12.4. Caused by a certain system excitation (step function, ramp, initial values for the state vector) the squared control error between a plant output  $y$  and a desired model trajectory  $y_M$  is computed over a freely chosen time interval  $[t_1 ; t_2]$ . As a special case,  $y_M$  could be a constant  $y_M = y_{stat}$  when we require a

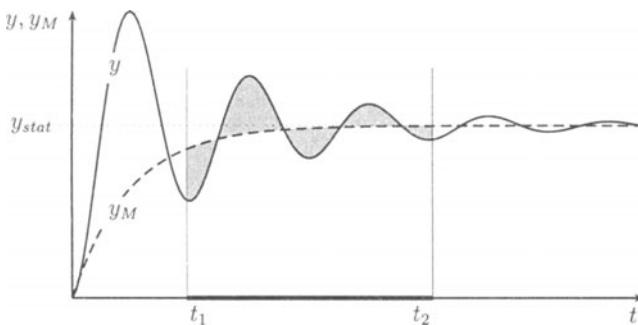


Fig. 12.4: Specifications on transient- and stationary behavior

small stationary deviation  $(y - y_{stat})^2$ . For that example  $t_1$  could be chosen as a desired settling time. But we stress that the time intervals can freely be changed between the later described design iterations. For a desired transient behavior, represented by a

chosen model trajectory  $y_M$ , the desired rise-time and the desired settling time could be a good initial choice for  $t_1$  and  $t_2$ .

A further important type of time domain criteria, namely

$$\max_{t_1 \leq t \leq t_2} |y(t, \mathbf{k}) - \bar{y}| \quad (12.1.10)$$

with

$$\bar{y} := (y_{\max} + y_{\min})/2 \quad (12.1.11)$$

reflects the requirement that the variable  $y$  may not exceed a given maximal value  $y_{\max}$  and may not fall below a minimal value  $y_{\min}$  in the time interval  $[t_1, t_2]$ ,

$$y_{\min} \leq y(t, \mathbf{k}) \leq y_{\max} \quad (12.1.12)$$

Especially for given constraints on control- or state variables this type of criterion can be used, see Fig. 12.5.

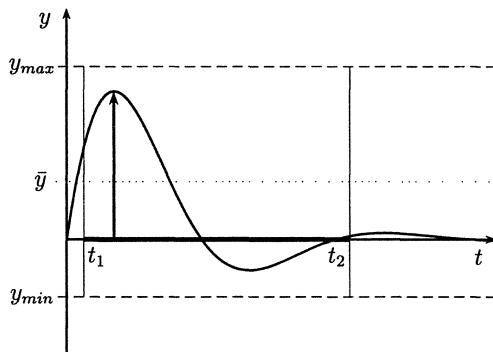


Fig. 12.5: Time domain constraints

#### *Frequency domain specifications*

Just as we can use the squared error for time functions as a performance criterion we can do it for frequency dependent functions. As an example the requirement for a small deviation of the gain of any frequency function  $|g(j\omega, \mathbf{k})|$  from a desired model function  $|g_M(j\omega)|$  over a freely chosen frequency interval  $[\omega_1, \omega_2]$  can be expressed by the following criterion

$$\int_{\omega_1}^{\omega_2} [|g(j\omega, \mathbf{k})| - |g_M(j\omega)|]^2 d\omega / (\omega_2 - \omega_1) \quad (12.1.13)$$

Also constraints can be formulated in the frequency domain where

$$\max_{\omega_1 \leq \omega \leq \omega_2} |g(j\omega, \mathbf{k})| - \bar{g} \quad (12.1.14)$$

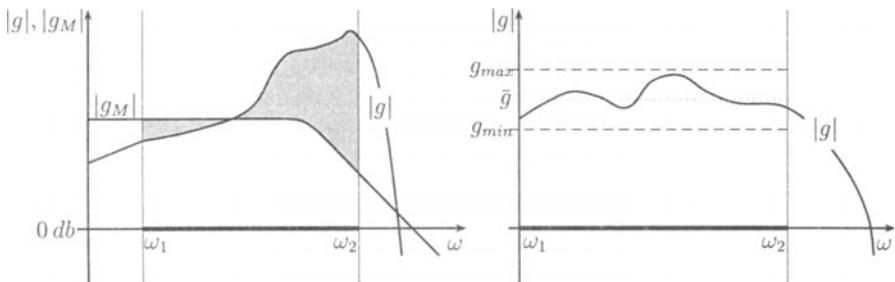


Fig. 12.6: Specifications in frequency domain

is an example with

$$\bar{g} := (g_{\max} + g_{\min})/2 \quad (12.1.15)$$

#### *Other specifications*

We can also include criteria of the type

$$k_i^2 \quad (12.1.16)$$

which directly take the magnitude of a controller coefficient into consideration. This type of criterion is also useful for investigations to simplify a given controller structure by reducing some of its gains to zero. Also the norm of the controller parameter vector

$$\|\mathbf{k}\| \quad (12.1.17)$$

can be applied directly in the design. A typical example for an application specific criterion from flight control is the expression

$$\left| \frac{q(0, \mathbf{k})}{n_z(0, \mathbf{k})} \right| \quad (12.1.18)$$

where after a step excitation of a normal acceleration  $n_z$ , the ratio of pitch rate  $q$  and normal acceleration at  $t = 0$  is considered as a relevant unit of measure for the pilot.

## 12.2 Concept of the Design by Optimizing a Vector Performance Index

Assume that we have formulated the design specifications by  $L$  performance indices  $c_i, i = 1, \dots, L$ . The  $L$  indices are now condensed in a performance vector

$$\mathbf{c} := [c_1, \dots, c_L]^T \quad (12.2.1)$$

It is obvious that the magnitude of this vector depends on the actual controller coefficients  $\mathbf{k}$ , i.e. we have

$$\mathbf{c} = \mathbf{c}(\mathbf{k}) \quad (12.2.2)$$

We start our design with initial controller coefficients  $\mathbf{k} = \mathbf{k}^0$  resulting in the initial performance vector

$$\mathbf{c}^0 := \mathbf{c}(\mathbf{k}^0) \quad (12.2.3)$$

Our design goal is to determine controller coefficients  $\mathbf{k}$  with a satisfying system behavior. That means the values of the performance criteria have to become sufficiently small, at least a certain level of requirement  $\bar{d}_i$  has to be satisfied,

$$c_i \leq \bar{d}_i, \quad i = 1, \dots, L \quad (12.2.4)$$

With the level of requirement vector  $\bar{\mathbf{d}} := [\bar{d}_1 \dots \bar{d}_L]^T$  we write

$$\mathbf{c} \leq \bar{\mathbf{d}} \quad (12.2.5)$$

In general the level of requirement  $\bar{d}_i$  is known in advance only for criteria which represent given physical or control theoretical limits as for instance criteria for maximal actuator signals, maximal allowed over-shoot, eigenvalue stability and the like. Otherwise the level of requirement, for instance for a squared control error, cannot be given in advance. The level of requirement  $\bar{\mathbf{d}}$  is a subjective decision of the designer which can be changed between different design iterations as we will see later. But for the moment we simply assume a fixed given level of requirement  $\bar{\mathbf{d}}$ .

For better demonstration of our considerations graphical presentations are restricted to the two dimensional case. But all statements are valid for a general  $L$ -dimensional performance vector. In Fig. 12.7, where  $C_a$  is the set of all attainable performance vectors,

$$C_a := \{\mathbf{c} \mid \exists \mathbf{k} : \mathbf{c} = \mathbf{c}(\mathbf{k})\} \quad (12.2.6)$$

we see for a constructed example the initial performance vector  $\mathbf{c}^0 = \mathbf{c}(\mathbf{k}^0)$  and the set of performance vectors satisfying a given level or requirement  $\bar{\mathbf{d}}$ . This set is denoted as the set of satisfactory performance vectors  $\bar{C}$ ,

$$\bar{C} := \{\mathbf{c} \mid \mathbf{c} \leq \bar{\mathbf{d}}\} \quad (12.2.7)$$

In general, however, there exists not necessarily a controller vector  $\mathbf{k}$  with the property  $\mathbf{c}(\mathbf{k}) < \bar{\mathbf{d}}$ . In that case the level of requirement is not attainable with the chosen controller structure. Otherwise the level of requirement is attainable as for our example of Fig. 12.7. The attainable performance vectors which are satisfactory are called the set of satisfactory solutions  $\bar{C}_a$ ,

$$\bar{C}_a := \bar{C} \cap C_a \quad (12.2.8)$$

It should be noted that in general neither the set of attainable performance vectors  $C_a$ , particularly its boundary  $\partial C_a$ , nor the set of satisfactory solutions  $\bar{C}_a$  is known. (In addition, we have to keep in mind that in general it is not known if a set with a special property is connected.)

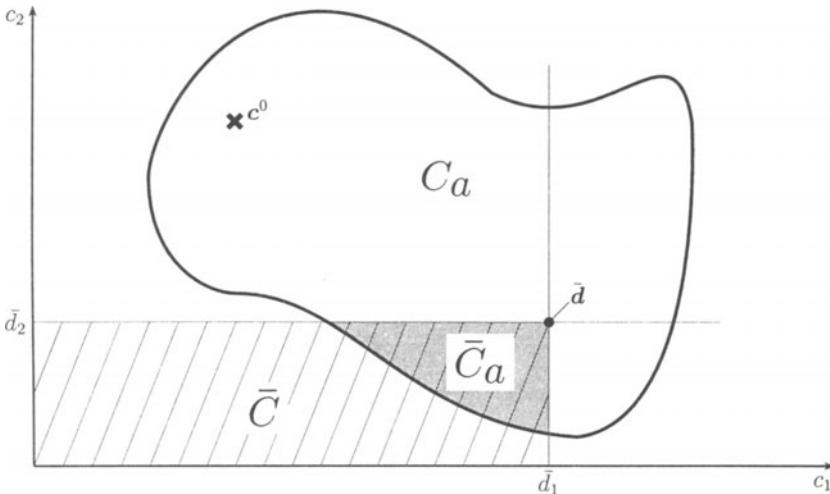


Fig. 12.7: Level of requirement  $\bar{d}$ , satisfactory performance vectors  $\bar{C}$ , attainable performance vectors  $C_a$  and satisfactory solutions  $\bar{C}_a$

Now we introduce a relation on the set of performance vectors. It is said that a performance vector  $\tilde{\mathbf{c}}$  is better than a performance vector  $\hat{\mathbf{c}}$  written as

$$\tilde{\mathbf{c}} \leq \hat{\mathbf{c}} \quad (12.2.9)$$

if each component  $\tilde{c}_i$  has a value less or equal  $\hat{c}_i$

$$\tilde{c}_i \leq \hat{c}_i, \quad i = 1, \dots, L \quad (12.2.10)$$

The set of better performance vectors of  $\hat{\mathbf{c}}$  is denoted by

$$B(\hat{\mathbf{c}}) := \{\mathbf{c} \mid \mathbf{c} \leq \hat{\mathbf{c}}\} \quad (12.2.11)$$

Unfortunately, it is not possible to compare two arbitrarily chosen performance vectors by the above relation. The relation ' $\leq$ ' represents only a partial but not a total structure of order on the set of performance vectors.

The better solutions of  $\mathbf{c}$  are the attainable performance vectors in the lower left quadrant of a coordinate system with axes parallel to the criteria axes and with its origin at  $\mathbf{c}$ . So in the constructed example the satisfactory solutions  $\bar{C}_a$  do not belong to the set of better performance vectors  $B(\mathbf{c}^0)$  of our initial performance vector  $\mathbf{c}^0$  as is shown in Fig. 12.8. Therefore, it is not sufficient to search for a satisfactory solution only in the set  $B(\mathbf{c}^0)$  of better performance vectors of  $\mathbf{c}^0$ .

With the set of better solutions belonging to a performance vector we can characterize a specific set of attainable performance vectors. It is the set of the so-called compromise solutions  $P$ . A compromise solution is characterized by the property that for it an open neighborhood  $U$  exists which does not contain any better point different from  $\mathbf{c}$ ,

$$P := \{\mathbf{c} \mid \exists U(\mathbf{c}) : B(\mathbf{c}) \cap U(\mathbf{c}) = \{\mathbf{c}\}\} \quad (12.2.12)$$

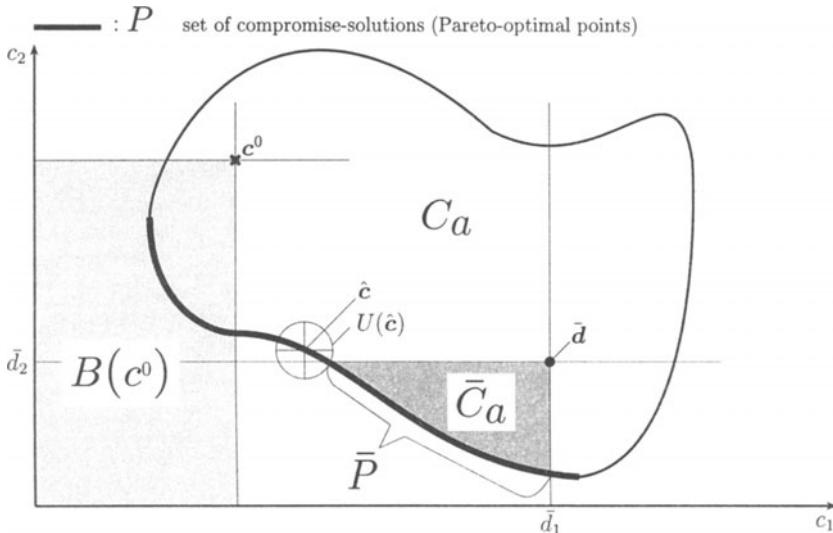


Fig. 12.8: Better performance vectors  $B(c^0)$ , compromise solutions  $P$ , satisfactory compromise solutions  $\bar{P}$ .

Hence, a compromise solution has a neighborhood in which starting from it no criterion can be decreased without increasing at least one other criterion. The compromise solutions are on the boundary  $\partial C_a$  of all attainable performance vectors  $C_a$ . For our example all points of the fat lined curve in Fig. 12.8 represent compromise solutions.

Starting from a point which is not in  $P$ , at least one criterion can be decreased without increasing any other criterion. Such a point is not a compromise solution because we can decrease all components of the performance vector, that is, for each neighborhood  $U$  of such a point better performance vectors exist which are attainable. Compromise solutions are in a certain sense optimal points. In the literature they are sometimes called Pareto-optimal points due to an Italian scientist of political economy who has studied such points already about 1890, [138].

Pareto-optimal points or compromise solutions  $c \in P$  are contained in each set of satisfactory solutions  $\bar{C}_a$ . They are called the set of satisfactory compromise solutions, see Fig. 12.8

$$\bar{P} := P \cap \bar{C}_a \quad (12.2.13)$$

Our goal now will be to find such a satisfactory compromise solution starting from  $c^0 = c(k^0)$ , that means we want to determine controller coefficients  $k$  such that  $c(k) \in \bar{P}$ . For that we introduce for each performance criterion  $c_i$  a design parameter  $d_i$  which has to be chosen greater than zero,

$$d_i > 0, \quad i = 1, \dots, L \quad (12.2.14)$$

Now the function

$$\alpha(k) := \max_i \left( \frac{c_i(k)}{d_i} \right) \quad (12.2.15)$$

is minimized w.r.t.  $\mathbf{k}$  by application of any parameter optimization procedure,

$$\min_{\mathbf{k}} \alpha(\mathbf{k}), \quad \text{initial value } \mathbf{k}^0 \quad (12.2.16)$$

We obtain a  $\mathbf{k}^*$  for which a neighborhood exists in which at least one criterion cannot further be decreased.

*Remark 12.2.* With a  $\mathbf{k}^*$  minimizing  $\alpha$  we have reached either a compromise solution  $\mathbf{c}(\mathbf{k}^*) \in P$  or a point which has a neighborhood in which less than  $L$  criteria can be decreased without increasing the remaining criteria (so the last are locally constant). Such a solution is denoted as a weak compromise solution, see Fig. 12.9). Since weak compromise solutions usually do not appear in applications we do not further consider such points.

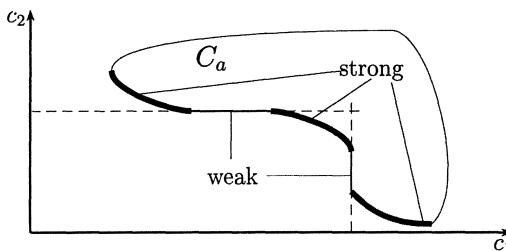


Fig. 12.9: Strong and weak compromise solutions

□

Hence, optimizing parameters as above yields a certain (strong) compromise solution  $\mathbf{c}^* = \mathbf{c}(\mathbf{k}^*) \in P$  with a corresponding vector of controller coefficients  $\mathbf{k}^*$ .

For each compromise solution  $\tilde{\mathbf{c}} = \mathbf{c}(\hat{\mathbf{k}}) \in P$  which can be connected with another compromise solution  $\check{\mathbf{c}} \in P$  by a path which lies totally in  $P$ , there exists at least one vector of design parameters  $\hat{\mathbf{d}} = [\hat{d}_1 \dots \hat{d}_N]^T$  which, put into the above problem (12.2.15), yields  $\check{\mathbf{c}}$  as the solution of the parameter optimization problem (12.2.16) with initial parameter values  $\mathbf{k}^0 = \hat{\mathbf{k}}$ . Hence, for our constructed example of Fig. 12.7 or of Fig. 12.8 we would be able to reach each compromise solution by choosing a suitable design vector  $\mathbf{d}$  for a parameter optimization step starting with  $\mathbf{k}^0$ . Unfortunately, we do not know the right design parameters  $\mathbf{d}$  resulting immediately in a certain  $\check{\mathbf{c}} \in P$  and especially, as desired in the set  $\bar{P}$  of satisfactory compromise solutions. However, we can try to reach a satisfactory compromise systematically during a sequence of design iterations.

For that, in the first design step each design parameter  $d_i$  is chosen greater than the respective criterion value,

$$c_i(\mathbf{k}^0) \leq d_i^1, \quad i = 1, \dots, L \quad (12.2.17)$$

(Upper indices characterize the iteration step.) Since  $c_i(\mathbf{k}^0)/d_i^1 \leq 1$  for each  $i$  we trivially have  $\alpha(\mathbf{k}^0) = \max_i(c_i(\mathbf{k}^0)/d_i^1) \leq 1$ . Since  $\alpha$  is minimized over  $\mathbf{k}$  it follows for  $\mathbf{k}^1$ , the result of the first optimization step,  $\alpha(\mathbf{k}^1) \leq 1$  and therefore,

$$c_i(\mathbf{k}^1) \leq d_i^1, \quad i = 1, \dots, L \quad (12.2.18)$$

Hence, with the choice of design parameters as in (12.2.17) no criterion  $c_i$  can exceed the corresponding design parameter  $d_i$  after an optimization step. Thus, we move always within the chosen level of design parameters  $\mathbf{d}^1$ .

In our example the level of requirement  $\bar{d}_1$  for criterion  $c_1$  is already satisfied with the assumed  $\mathbf{c}^0$ , see Fig. 12.10. It will not be exceeded by  $c_1(\mathbf{k})$  after the first optimization step if we put  $d_1^1 \leq \bar{d}_1$ . Since criterion  $c_2(\mathbf{k}^0)$  does not yet satisfy the corresponding level of requirement  $\bar{d}_2$  we appoint to it a higher priority than for  $c_1$ . This is expressed by a choice of design parameters which do not unnecessarily constrain the space of performance criteria w.r.t.  $c_1$  (which already satisfies its level of requirement  $\bar{d}_1$ ) and in addition by pulling up the design parameter for  $c_2$  as near as possible to its criterion value,

$$\begin{aligned} d_1^1 &= \bar{d}_1 \\ d_2^1 &= c_2^0 \end{aligned} \quad (12.2.19)$$

In the first optimization step the performance vector will be improved until a compromise solution is reached within that set of attainable performance vectors which is restricted by the above chosen design parameter level  $\mathbf{d}^1$ .

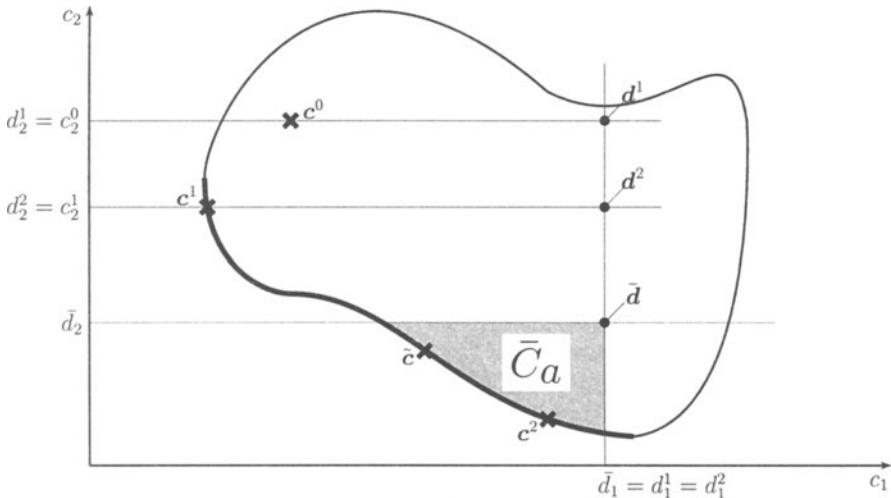


Fig. 12.10: Two design steps

Let the result of the first design step be  $\mathbf{c}^1 = \mathbf{c}(\mathbf{k}^1)$  of Fig. 12.10. It is seen that we have not yet reached a satisfactory compromise solution ( $c_2^1 > \bar{d}_2$ ). But if we now choose the

design parameters as

$$\begin{aligned} d_1^2 &= d_1^1 = \bar{d}_1 \\ d_2^2 &= c_2^1 \end{aligned} \quad (12.2.20)$$

a compromise solution below the level  $d_2^2$  is guaranteed after the second optimization step. If we assume a solution  $c^2 = c(\mathbf{k}^2)$  as in Fig. 12.10, then we have found a satisfactory compromise solution. Then the solution of all further design steps will remain a satisfactory compromise solution if we choose

$$\begin{aligned} c_1^{i-1} &\leq d_1^i \leq \bar{d}_1 \\ c_2^{i-1} &\leq d_2^i \leq \bar{d}_2, \quad i > 2 \end{aligned} \quad (12.2.21)$$

for the design parameters  $d_1, d_2$ . Should our goal be  $c_2 \leq \tilde{c}_2$  we would surely reach the compromise solution  $\tilde{c}$  of Fig. 12.10 after few optimization steps with the choice

$$\begin{aligned} d_1^i &= c_1^{i-1} \\ d_2^i &= \tilde{c}_2, \quad i > 2 \end{aligned} \quad (12.2.22)$$

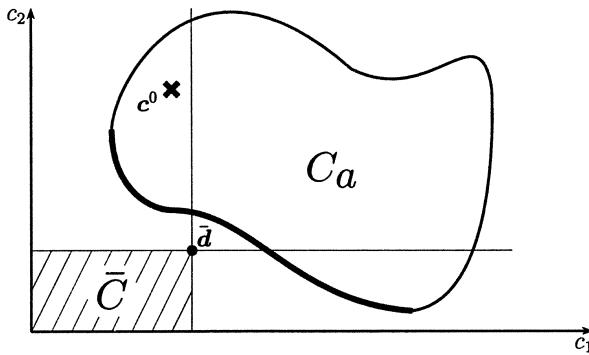
The choice of

$$c^\nu := c(\mathbf{k}^\nu) \leq \mathbf{d}^{\nu+1} \leq \mathbf{d}^\nu, \quad \mathbf{d}^{\nu+1} \neq \mathbf{d}^\nu \quad (12.2.23)$$

for the design parameter vector in the design step  $(\nu+1)$  makes it possible to approach a desired compromise solution. The new design vector is chosen lower than the preceding design vector but higher than the performance vector result of the  $\nu$ -th design step. The sequence of design parameters decreases monotonously. Optimizing the performance vector for a given  $\mathbf{d}^{\nu+1}$  results in controller parameters  $\mathbf{k}^{\nu+1}$  and closes the design step  $(\nu+1)$ .

For practical problems the above described iteration starting from a  $\mathbf{k}^0$  certainly leads to a compromise solution but usually not to a satisfactory one at the first attempt. In most cases this occurs for the following reasons:

- i) The design problem was not formulated adequately enough by the chosen performance criteria. Especially at the beginning of a design, criteria are often introduced which are, without knowing about it, unnecessarily contradicting or some undesired system behavior is not considered by the chosen criteria. But in general this can be recognized by the designer after some optimization trials. Then he has to change suitably the respective performance criteria which represent the original design specifications.
- ii) The desired level of requirement  $\bar{d}$  is not attainable, see Fig. 12.11, with the underlying controller structure  $(\bar{C}_a = \bar{C} \cap C_a = \emptyset)$ . If a higher level of design specifications cannot be accepted the controller structure has to be changed.
- iii) The initial values for the controller coefficients  $\mathbf{k}^0$  are chosen such that the optimization step results in a local minimum which is not satisfactory, see Fig. 12.12. We can then try to find initial values  $\mathbf{k}^0$  sufficiently near a region of satisfactory compromise solutions possibly by application of other design methods. But one

Fig. 12.11: Not attainable level of requirement  $\bar{d}$ 

can also try to come to a better starting point by first including only such performance criteria which do not yet satisfy their level of requirement and from which the parameter optimization can then be continued after inclusion of the remaining criteria.

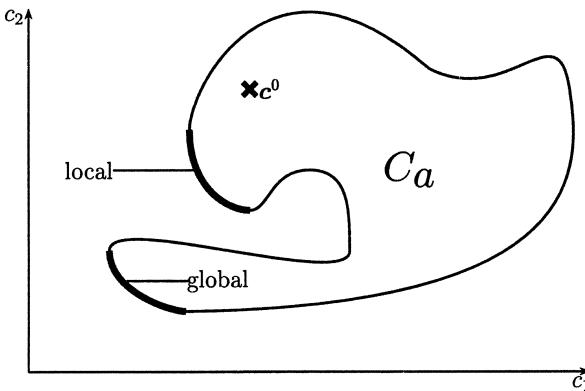


Fig. 12.12: Local and global compromise solutions

- iv) The choice of a design value  $d^{\nu+1}$  is unnecessarily restricting the possible solutions. So for the example of Fig. 12.13 with the described design systematic for choosing the design vector  $d$ ,  $c^\nu \leq d^{\nu+1} \leq d^\nu$  and the choice of  $d_1^{\nu+1} = \bar{d}_1$  we could not reach the compromise solutions satisfying the level of requirement  $\bar{d}$ , however, it would be possible with the choice of  $\tilde{d}_1^{\nu+1}$ .

Of course the designer first does not know why the design result has not been improved after an optimization step. We stress again that he really does not know the set of attainable performance vectors, especially he does not know the set of compromise solutions. But with the aid of the already generated different iteration step results,

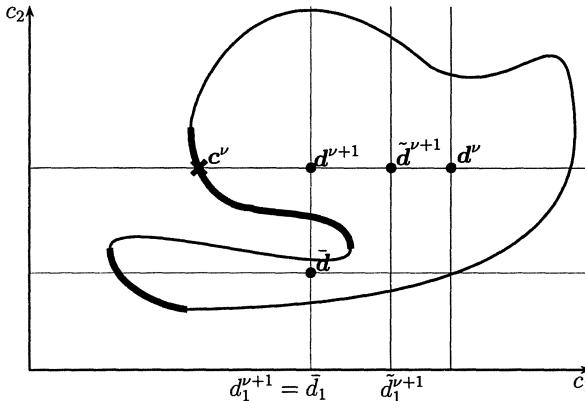


Fig. 12.13: Unnecessarily restricting the possible solution set with choice of  $d^{\nu+1}$

in combination with their physical and control theoretical interpretation, the designer comes very fast to insights which helps him to take suitable actions if he is not yet satisfied with the design result found hitherto. With the following iteration scheme we summarize the practical design procedure:

Starting with the mathematical model of the plant and the desired system behavior, one proceeds in the following steps (the additional upper index  $t$  characterizes the preparations made for the iteration loop 5-6-7-5-..., that means the choice of the controller structure, the initial values for the controller coefficients  $\mathbf{k}^0$ , the formulation of the performance criteria  $c_i$ , and the first design vector  $\mathbf{d}^1$ ):

- $t = 1$
- 1. choose a controller structure
- 2. assign initial values  ${}^t\mathbf{k}^0$  to the controller coefficients
- 3. formulate the desired system behavior by a performance vector  ${}^t\mathbf{c}$
- 4.  $\nu = 1$  choose the design vector  ${}^t\mathbf{d}^1$  for the first design ( $t$ -)iteration  
 ${}^t\mathbf{c}^0 := \mathbf{c}({}^t\mathbf{k}^0) \leq {}^t\mathbf{d}^1$   
go to 6.
- 5.  $\nu = \nu + 1$  choose the design vector  ${}^t\mathbf{d}^\nu$  for the ( $\nu$ )-th design step  
 ${}^t\mathbf{c}^{\nu-1} \leq {}^t\mathbf{d}^\nu \leq {}^t\mathbf{d}^{\nu-1}, \quad {}^t\mathbf{d}^\nu \neq {}^t\mathbf{d}^{\nu-1}$
- 6. minimize the goal function  $\alpha(\mathbf{k}) = \max_i({}^t c_i(\mathbf{k}) / {}^t d_i^\nu)$   
with result  ${}^t\mathbf{k}^\nu$  and  ${}^t\mathbf{c}^\nu = \mathbf{c}({}^t\mathbf{k}^\nu)$

7. if no (significant) improvement ( ${}^t\mathbf{c}^\nu \approx {}^t\mathbf{c}^{\nu-1}$ ) :  
 $t = t + 1, {}^t\mathbf{k}^0 = {}^{t-1}\mathbf{k}^\nu, \text{ goto 4. (or 3. or 2. or 1.)}$
8. evaluate the system behavior: if not satisfied goto 5.

If the resulting system behavior is not yet satisfactory a further iteration (from 5. until 7.) has to be performed. This process will be repeated until no significant improvement of the design result can be obtained. If then the design goal still has not yet be obtained the design can be continued by the following possibilities:

- go to 4. and choose a new initial design parameter vector  $\mathbf{d}^1$ ,
- go to 3. and formulate the desired system behavior by changing the vector  $\mathbf{c}$ ,
- go to 2. and assign new initial values to the controller coefficients,
- go to 1. and change the controller structure.

Leaving an inner iteration loop does not imply that we have to go back to the starting point of the design.

*Remark 12.3.* To realize the design procedure by optimizing a vector performance index, essentially a problem specific subprogram has to be written that first evaluates for a given controller parameter vector  $\mathbf{k}$  the performance vector  $\mathbf{c}(\mathbf{k})$  and then with the chosen design vector  $\mathbf{d}$  evaluates the target function  $\alpha$ . The minimum search can then be performed by a library optimization program.  $\square$

*Remark 12.4.* The design method is not an analytical one as for instance the design by pole placement, rather it is a numerical procedure with interaction by the designer. For the numerically based procedure no special structure of the plant or of the controller has to be assumed. So the linearity assumption for the plant and the controller which we have made for simplicity at the beginning is not necessary for application of such a procedure. Performance criteria can as well be evaluated for nonlinear systems. For such systems one could try to guarantee stability in an indirect manner only by inclusion of usual time domain criteria formulated for improving any simulation performance. It is also possible to supply any time domain criteria for the nonlinear system by eigenvalue criteria for the system linearized at different operating points. An important advantage of a design procedure based on numerical parameter optimization is the possibility to use realistic models of the total system directly in the design. For instance, is it also possible to combine a discrete controller algorithm (including such effects as limited word length, analog-digital devices, etc.) with the continuous, possibly nonlinear plant. A more realistic model in general requires higher computation time for evaluation of the performance criteria. Since for a minimum search usually a high number of target function evaluations are necessary the required computing time should be roughly estimated in advance.  $\square$

*Remark 12.5.* In the choice of a parameter optimization routine it is important to consider the assumptions on the target function (12.2.15). For the interactive data-base supplied software package at DLR [92] different subprograms are available optionally for the three standard methods of nonlinear programming,

- direct search optimization,
- unconstrained optimization,
- constrained optimization.

For instance the unconstrained optimization method of Fletcher and Powell [66] requires the target function  $\alpha$  at least to be (two times) continuously differentiable. Since the max-function (12.2.15) is not continuously differentiable with respect to the  $c_i$  on account of the product rule, this is also true with respect to  $k_i$ . Therefore, the target function has been replaced by the function

$$\bar{\alpha}(\mathbf{k}) := \frac{1}{\rho} \ln \left( \sum_{i=1}^L \exp\{\rho c_i(\mathbf{k})/d_i\} \right), \quad \rho > 0 \quad (12.2.24)$$

This function is continuously differentiable w.r.t.  $c_i$ . For numerical computations it can be transformed to

$$\bar{\alpha}(\mathbf{k}) := \alpha(\mathbf{k}) + \frac{1}{\rho} \ln \left( \sum_{i=1}^L \exp\{\rho c_i(\mathbf{k})/d_i - \alpha(\mathbf{k})\} \right). \quad (12.2.25)$$

Since the arguments of the exponential function are all negative,  $\bar{\alpha} - \alpha \leq (\ln L)/\rho$ . For a sufficiently high value  $\rho$  it follows  $\bar{\alpha}(\mathbf{k}) \geq \alpha(\mathbf{k})$  ( $\rho = 20$  turned out to be a good value with a difference between  $\alpha$  and  $\bar{\alpha}$  of typically 2 to 5% at  $\alpha \approx 1$ ), so the optimization of  $\alpha$  can be done sufficiently exact by optimizing  $\bar{\alpha}$ .

A further advantage of the modified optimization problem is the dependency of  $\bar{\alpha}$  on all components  $c_i$  of the performance vector  $\mathbf{c}$  whereas  $\alpha$  normally depends only on one component. Optimizing  $\bar{\alpha}$  excludes weak compromise solutions (see Remark 12.2) and therefore results in a Pareto-optimal point, that means a strong compromise solution. For further details concerning the computer realization of the design method we refer to [92, 93, 79, 65]. Numerous practical applications of the described design method have been performed, for robust controller design we mention [160, 112, 80, 27].  $\square$

## 12.3 Case Study: Car Steering

The controller design method by optimizing a vector performance index is demonstrated in this case study on active steering of a bus. We consider the linearized model introduced in Section 1.2,

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} a_{11}(\tilde{m}, v) & a_{12}(\tilde{m}, v) \\ a_{21}(\tilde{m}, v) & a_{22}(\tilde{m}, v) \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} b_1(\tilde{m}, v) \\ b_2(\tilde{m}, v) \end{bmatrix} \delta_f \quad (12.3.1)$$

which describes the car steering dynamics for front wheel steering, where the front wheel steering angle  $\delta_f$  is the only input variable. The two state variables are sideslip angle  $\beta$  and yaw rate  $r$ .

The matrix coefficients depend on two parameters, the virtual mass  $\tilde{m}$  and the car velocity  $v$  which are considered to be uncertain within the intervals

$$\begin{aligned}\tilde{m} &\in [\tilde{m}^-; \tilde{m}^+] \\ v &\in [v^-; v^+]\end{aligned}\quad (12.3.2)$$

The dependencies are given by (1.2.1). The other parameters appearing there are considered to be constant, namely the cornering stiffnesses  $c_f$  and  $c_r$  (the uncertainty in the road adhesion  $\mu \in [0.5; 1.0]$  is incorporated in the larger virtual mass uncertainty), the center of gravity location between the axis, characterized by the distances  $l_f$  and  $l_r$  (where  $\ell = l_r + l_f$  is the wheelbase), and the radius of inertia  $i$ . The operating domain is a two-dimensional box with the four vertex operating points numbered as,

$$\begin{aligned}\text{operating point 1 : } &(\tilde{m}^-, v^-) \\ \text{operating point 2 : } &(\tilde{m}^-, v^+) \\ \text{operating point 3 : } &(\tilde{m}^+, v^-) \\ \text{operating point 4 : } &(\tilde{m}^+, v^+)\end{aligned}\quad (12.3.3)$$

The lateral acceleration  $a_f$  at the front axle is given by the linear output equation (A.2.10) with  $d_2 = 0$ . Substituting  $\tilde{J} = i^2\tilde{m}$  into (A.2.10) and assuming for the moment  $i^2 = l_f l_r$  (see (2.5.9)) we get

$$a_f = d(-\beta - \frac{\ell_f}{v}r + \delta_f) = [-d \quad -d\frac{\ell_f}{v} \quad d] \begin{bmatrix} \beta \\ r \\ \delta_f \end{bmatrix} =: \mathbf{c}^T \begin{bmatrix} \beta \\ r \\ \delta_f \end{bmatrix} \quad (12.3.4)$$

where  $d = c_f(\ell/\ell_r)/\tilde{m}$ . In Section 2.5 it was found under the mass distribution assumption  $i^2 = l_f l_r$  that purely integrating the yaw rate error, see Fig. 12.14,

$$\dot{\delta}_f = k_I(w - r), \quad k_I = 1.0 \quad (12.3.5)$$

leads to a special structural property of the closed loop system:

The input-output behavior

$$a_f = g_f(s, \tilde{m}, v)w \quad (12.3.6)$$

is characterized by the simple first order transfer function,

$$g_f(s, \tilde{m}, v) = \frac{v}{1 + T(\tilde{m}, v)s} \quad (12.3.7)$$

with the virtual mass and velocity dependent time constant  $T(\tilde{m}, v) = \tilde{m}v\ell_r / (\ell c_f)$ . Since the closed loop system is described by three state variables  $(\beta, r, \delta_f)$ , this structural property implies two pole/zero cancellations in the rational expression

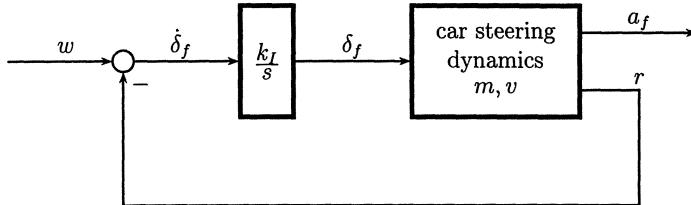


Fig. 12.14: Car steering with integrating control

$c^T[sI - \mathbf{A}_c]^{-1}[0 \ 0 \ 1]^T$  (where we have suppressed the dependency on  $\tilde{m}$  and  $v$ ) with  $\mathbf{A}_c$  the closed loop system matrix (2.5.3). Indeed, the purely integrating feedback loop (12.3.5) produces two closed loop poles cancelled by the two zeros of the open loop transfer function  $a_f/\delta_f$  which are invariant with respect to state feedback (especially w.r.t. yaw rate feedback). With the above purely integrating feedback structure these zeros have the additional property of being “output decoupling zeros”, [121], of the closed loop system.

In a state space formulation the output decoupling property is characterized by a two-dimensional subspace of the three-dimensional state space  $[\beta \ r \ \delta_f]^T$ , invariant with respect to the closed loop system matrix  $\mathbf{A}_c$  of (2.5.3) and contained in the kernel of the output matrix  $c^T$  of (12.3.4). All motions completely contained in this subspace are unobservable from the output variable  $a_f$ , that means all initial states in this subspace lead to a zero output,  $a_f \equiv 0$ .

From (12.3.4) the unobservable subspace is determined by the equation

$$-\beta - \frac{\ell_f}{v}r + \delta_f = 0 \quad (12.3.8)$$

Since it was found that the described decoupling property is independent of  $\tilde{m}$  and  $v$  we have called it robust decoupling (Section 2.5).

For high velocity  $v$  the location of the output decoupling (complex conjugate) zero pair has the disadvantageous consequence of the cancelled eigenvalue pair being badly damped. From now on we concentrate on high velocity and consider in the following only the most critical velocity value,  $v = v^+$ . There remains one single uncertainty parameter  $\tilde{m} \in [\tilde{m}^- ; \tilde{m}^+]$ .

Now we ask: For  $v = v^+$  and  $\tilde{m} \in [\tilde{m}^- ; \tilde{m}^+]$ , can we find yaw rate feedback control with well damped closed loop eigenvalues and with the (output) decoupling property in addition?

In Section 2.6 it was shown that this is possible if rear wheel steering can be applied in addition to front wheel steering. Then the damping of the complex conjugate eigenvalue pair can arbitrarily be increased without destroying the decoupling effect. By (gain scheduled) feedback of the yaw rate  $r$  to the rear wheel steering angle  $\delta_r$ , a zero pair of  $a_f/\delta_f$  can appropriately be assigned which is output decoupling for the closed loop system with the purely integrating feedback (12.3.5).

However, for the controller design of this section we consider only front wheel steering. Then, with the single input variable  $\delta_f$ , the zeros of  $a_f/\delta_f$  are invariant. Thus, for exact decoupling, the damping of the eigenvalue pair to be cancelled by these zeros is predetermined and cannot be changed. We have to relax our requirement and ask: Using only front wheel steering, can we design a yaw rate feedback controller resulting in well damped closed loop eigenvalues but with an additional almost decoupling effect? Almost decoupling means that the yaw rate influence on the lateral acceleration  $a_f$  of the front axle should be small in a sense that must be made precise later.

For that investigation we consider the city bus O 305 of Daimler-Benz with data given in Table 1.3 and  $v = v^+ = 20 \text{ [ms}^{-1}\text{]}$ . The condition for robust decoupling

$$i^2 = \ell_f \ell_r \quad (12.3.9)$$

is not satisfied and therefore, an exact pole/zero cancellation cannot be realized. However, with the integrating gain factor

$$k_I = 0.89 \quad (12.3.10)$$

proposed in [17] an almost decoupling effect is obtained. This can be seen in Fig. 12.15 where for operating point 2 ( $\tilde{m} = \tilde{m}^- = 9.95$ ) and operating point 4 ( $\tilde{m} = \tilde{m}^+ = 32.0$ ) the system response to the initial conditions

$$\beta(0) = 0, \quad r(0) = 1, \quad \delta_f(0) = \beta(0) + \frac{\ell_f}{v} r(0) \quad (12.3.11)$$

is shown. Three systems are compared. In system 1, the moment of inertia is reduced such that the ideal mass distribution assumption is satisfied, i.e.

$$S1 : \quad i^2 = 7.0831 = \ell_f \ell_r, \quad k_I = 1.0 \quad (12.3.12)$$

In system 2 the true moment of inertia is assumed and the best approximation to decoupling by (12.3.10) is used,

$$S2 : \quad i^2 = 10.85, \quad k_I = 0.89 \quad (12.3.13)$$

System 3 has a further decreased integrating gain, chosen to obtain good eigenvalue damping,

$$S3 : \quad i^2 = 10.85, \quad k_I = 0.1089 \quad (12.3.14)$$

Since the above chosen initial condition (12.3.11) is in the unobservable subspace of S1 (see (12.3.8)) we have  $a_f \equiv 0$  for S1 for all  $m \in [m^-; m^+]$ . For S2 and S3, exact decoupling is not possible, however, for S2 we have an almost decoupling effect in the sense of a well attenuated acceleration response due to the initial disturbance of the yaw rate. In any case for stationary constant disturbances the integrating controller guarantees  $a_f \rightarrow 0$  for  $t \rightarrow \infty$ .

The exact decoupling property of S1 and the almost decoupling property of S2 are also reflected in the eigenvalue/zero patterns of Fig. 12.15. For S1 we have an exact cancelling, for S2 the conjugate complex eigenvalue pair is near by the invariant zero

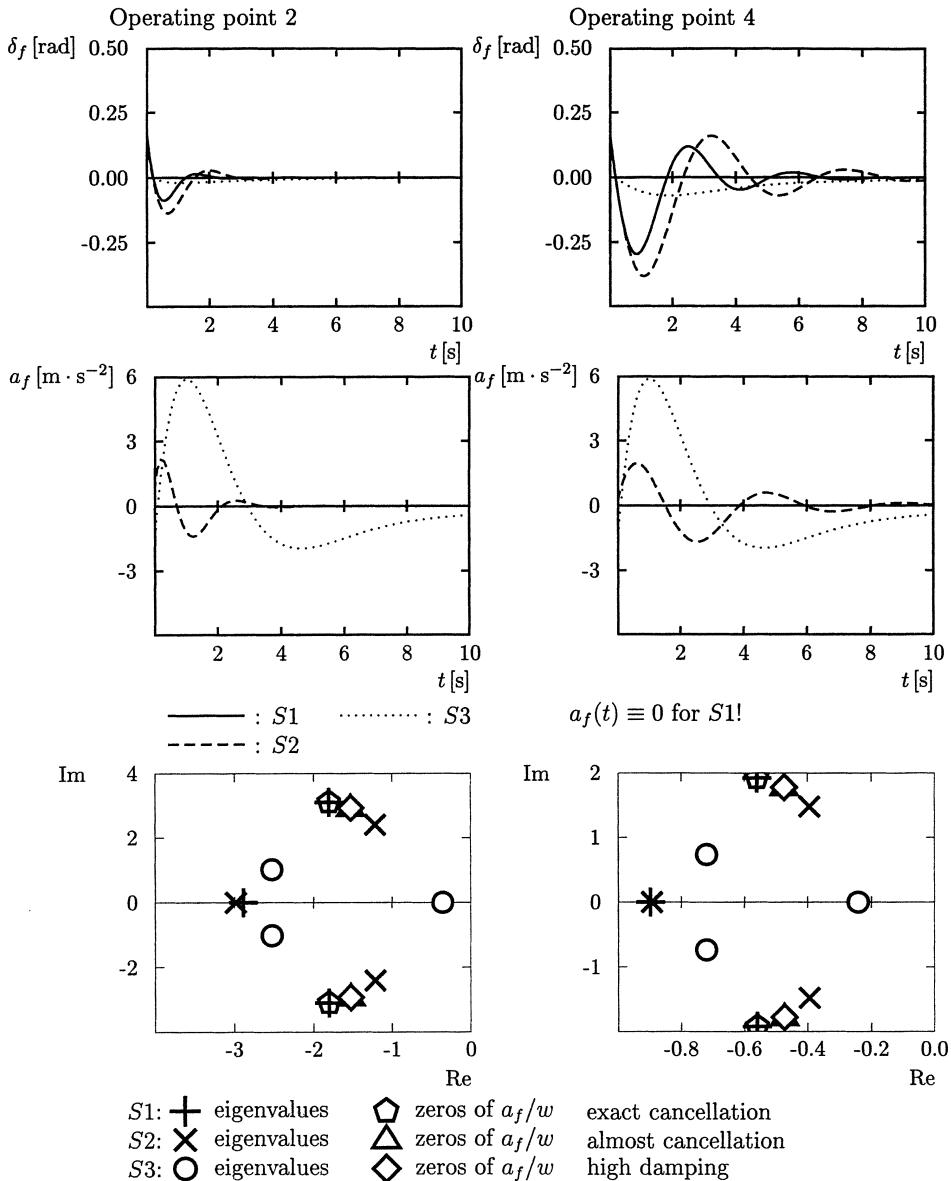


Fig. 12.15: Eigenvalue and simulation comparison for systems S1, S2 and S3

pair and the damping for both systems is very low. From a root locus plot it can be found that the gain  $k_I = 0.1086$  of S3 results in a satisfactory damping factor for operating point 2 and 4,

$$D^{(2)} = 0.928, \quad D^{(4)} = 0.7 \quad (12.3.15)$$

However, with decreasing  $k_I$  the real eigenvalue tends to zero (to the integrator pole of the open loop system) and the dynamic behavior of S3 becomes too sluggish. This effect is even worse for  $v < v^+$ . In addition, the yaw rate influence on lateral acceleration has been strongly amplified, see Fig. 12.15.

The characteristic properties for the three systems S1, S2, and S3, with different integrating gain  $k_I$  are compiled in Table 12.1:

closed loop system	controller	radius of inertia $r^2$	eigenvalue	damping $D^{(2)}$	$D^{(4)}$	decoupling property
S1	$k_I = 1.0000$	$7.0831 = \ell_f \ell_r$	bad:	0.500	0.280	exact
S2	$k_I = 0.8900$	10.850	bad:	0.450	0.257	almost
S3	$k_I = 0.1086$	10.850	good:	0.928	0.7	bad
design goal	?	10.850	good:	about 0.7		almost

Table 12.1: Integrating control with different gain and the design goal

System S2 will be our reference system w.r.t. the almost decoupling property and is used for comparison purposes during the controller design that follows. (Note that S1 assumes an ideal mass distribution and therefore is not suited as a basis of comparison.) As a starting point for the design we choose system S3 which already satisfies our main requirement for good eigenvalue damping. Now we start the design iteration scheme introduced in Section 12.2.

Step 1: choose a controller structure ( $t = 1$ )

Since we want to preserve the stationary rejection of constant disturbances (like cross-wind), we do not give up integration of the yaw rate error. An integrator can easily be implemented by an electric or hydraulic actuator without internal position feedback. We do not want complicate the forward path  $k_I/s$ , but with that simple controller of Fig. 12.14 it is not sufficient to reach our design goal. Thus, we extend it by a compensator in the feedback path as shown in Fig. 12.16, where the second order transfer function

$$g_c(s) = \tilde{g}_c(s) + d = \frac{\hat{k}(s + b_0)}{s^2 + a_1 s + a_0} + d = \frac{ds^2 + (da_1 + \hat{k})s + (da_0 + \hat{k})}{s^2 + a_1 s + a_0} \quad (12.3.16)$$

has been chosen. For yaw rate regulation in the stationary constant case we must have

$$\hat{r}_{stat} = r_{stat} \quad \text{for } r_{stat} = \text{const.} \quad (12.3.17)$$

which is satisfied with the controller coefficient constraint

$$d = 1 - \hat{k}b_0/a_0 \quad (12.3.18)$$

Hence, only the five controller coefficients

$$\mathbf{k} = [k_I \hat{k} a_0 a_1 b_0]^T \quad (12.3.19)$$

remain free for optimization.

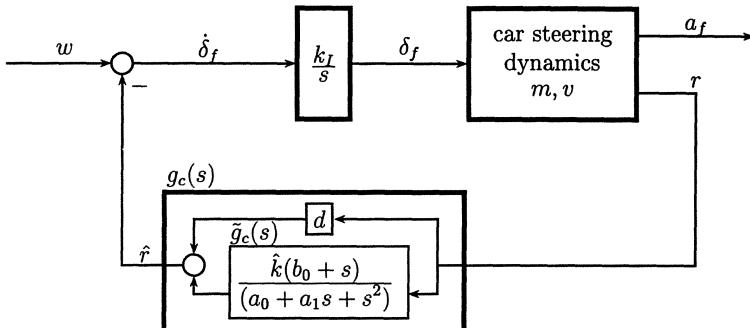


Fig. 12.16: Extension of the integrating controller structure by a feedback compensator

Step 2: give initial values  ${}^t\mathbf{k}^0$  to the controller coefficients ( $t=1, \nu = 0$ )

As initial values for the first optimization step

$${}^t\mathbf{k}^0 = \begin{bmatrix} k_I^0 \\ \hat{k}^0 \\ a_0^0 \\ a_1^0 \\ b_0^0 \end{bmatrix} = \begin{bmatrix} 0.1086 \\ 0.0000 \\ 9.0000 \\ 6.0000 \\ 4.0000 \end{bmatrix} \quad (12.3.20)$$

is chosen. This initial controller vector corresponds to the above integrating controller (with  $k_I = 0.1086$ ) because we have not closed the second order controller transfer function  $\tilde{g}_c(s)$  in the feedback loop of Fig. 12.16 ( $\hat{k} = 0, d = 1$ ). The transfer function  $\tilde{g}_c(s)$  in (12.3.16) is initially chosen to be stable and minimum-phase with two poles at  $-3.0$  and the zero at  $-4.0$ .

Step 3: formulate the desired system behavior by a performance vector  ${}^t\mathbf{c}$  ( $t=1, \nu = 0$ )  
The desired system behavior is primarily to obtain good damping and an almost decoupling property for the two vertex operating points 2 and 4. Furthermore, a certain stability margin should be satisfied and the actuator activity should not be too high. This desired system behavior has to be made precise by formulating the respective performance criteria:

Formulation of the damping specification (with  $D_j$  of (12.1.5)):

$$\tilde{c}_1^{(i)}(\mathbf{k}) = 1 - \min_j \{D_j(\mathbf{A}_c^{(i)}(\mathbf{k}))\}, \quad i = 2, 4 \quad (12.3.21)$$

Formulation of the almost decoupling specification:

$$\text{for response to the initial values (12.3.11)} : \begin{cases} \tilde{c}_2^{(i)}(\mathbf{k}) = \sqrt{\int_4^{10} [a_f^{(i)}(\mathbf{k}, t)]^2 dt}, & i = 2, 4 \\ \tilde{c}_3^{(i)}(\mathbf{k}) = \max_{0 \leq t \leq 4} |a_f^{(i)}(\mathbf{k}, t)|, & i = 2, 4 \end{cases} \quad (12.3.22)$$

Formulation of an actuator limit specification:

$$\text{for response to the initial values (12.3.11)} : \tilde{c}_4^{(i)}(\mathbf{k}) = \max_{0 \leq t \leq 4} |\delta_f^{(i)}(\mathbf{k}, t)|, \quad i = 2, 4 \quad (12.3.23)$$

Formulation of a stability margin specification:

$$\tilde{c}_5^{(i)}(\mathbf{k}) = \begin{cases} 1/[1 - \max_j [\operatorname{Re} s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))]] & \text{for } \max_j \operatorname{Re} s_j < 0 \\ 1 + \max_j [\operatorname{Re} s_j(\mathbf{A}_c^{(i)}(\mathbf{k}))] & \text{otherwise} \end{cases}, \quad i = 2, 4 \quad (12.3.24)$$

The  $\tilde{c}_1$ -criterion type is a measure for the minimal eigenvalue damping (see the discussion to (12.1.4) in Section 12.1). Its value decreases with increasing minimal damping.

The  $\tilde{c}_5$ -criterion type gets smaller for a smaller real part of the most right eigenvalue. Since the system response to the initial values (12.3.11) is considered as representative for the yaw rate influence on lateral acceleration, the meaning of the remaining criterion-types is evident. For the integral criteria  $\tilde{c}_2^{(2)}, \tilde{c}_2^{(4)}$  the time interval  $[4; 10]$  has been chosen since the corresponding lateral acceleration time response for system S2 (to which we want to compare our final results) is almost zero in that interval. The maximal values of  $a_f$  and  $\delta_f$  occur in the time interval  $[0; 4]$ , see Fig. 12.15. The performance vector now is brought into the following order:

$$\begin{aligned} \mathbf{c} &= [c_1 \dots c_5 \ c_6 \dots c_{10}]^T \\ &= [\tilde{c}_1^{(1)} \dots \tilde{c}_5^{(1)} \ \tilde{c}_1^{(2)} \dots \tilde{c}_5^{(2)}]^T \end{aligned} \quad (12.3.25)$$

Step 4: choose the design vector  ${}^t\mathbf{d}^1$  for the first design step  ${}^t\mathbf{c}^0 < {}^t\mathbf{d}^1 (t = 1, \nu = 1)$

We choose the design vector  ${}^1\mathbf{d}^1$  of column 2 in Table 12.2 for the first optimization run. Starting with an integrating gain factor of  $k_I = 0.1086 (\equiv S3)$ , which results in good damping our primary objective is to approximately keep this damping. Thus, we have chosen  ${}^1d_1^1 = {}^1d_6^1 = 0.3$ . That value represents the level of requirement  $\bar{d}_1 = \bar{d}_6 = 0.3$  for criterion  $c_1$  and  $c_6$ . By the argument of the last section it is guaranteed that herewith the minimal eigenvalue damping can not fall below  $1 - 0.3 = 0.7$  (see (12.3.21)) in both operating points.

At the same time the lateral acceleration should be better attenuated in the representative response due to a yaw rate initial value disturbance (for improving the instationary decoupling property). Therefore, we have put  ${}^1d_2^1 = {}^1c_2^0, {}^1d_3^1 = {}^1c_3^0$  and  ${}^1d_7^1 = {}^1c_7^0, {}^1d_8^1 = {}^1c_8^0$ . For these four criteria we can not specify the levels of requirement  $\bar{d}_2, \bar{d}_3, \bar{d}_7, \bar{d}_8$  in advance.

A better disturbance attenuation should not be gained at the cost of too much actuator activity and therefore the maximal steering angle value should not exceed the values

for system S2,  $\delta_{f,max}^{(2)}(S2) \approx 0.2$  and  $\delta_{f,max}^{(4)}(S2) \approx 0.45$ . This can be guaranteed with the choice of  ${}^1d_4^1 = 0.2 \doteq \bar{d}_4$ ,  ${}^1d_9^1 = 0.45 \doteq \bar{d}_9$

Furthermore for the stability margin constraint we are satisfied with a maximal real part of the closed loop eigenvalues near that at the beginning ( $k_I = 0.1086$ ) for operating point 4 and therefore choose  ${}^1d_5^1 = {}^1d_{10}^1 = 0.81 \approx \tilde{c}_5^{(4)}(\mathbf{k}^0) \doteq \bar{d}_5 = \bar{d}_{10}$ .

Step 6: minimization of the goal function  $\alpha(\mathbf{k})$  with result  ${}^t\mathbf{k}^\nu$  and  ${}^t\mathbf{c}^\nu$  ( $t = 1, \nu = 1$ )

The parameter optimization procedure is considered here as a black box. We do not give data about the optimization runs since they are dependent on the chosen optimization algorithm and the individual algorithm parameters that have to be given (e.g. convergence limits). As mentioned in the preceding section (Remark 12.5) we have several optimization algorithms available in our control design toolbox and usually do apply different ones during a design.

Here we only show the optimization result in Table 12.2 which is  ${}^1\mathbf{k}^1$  for the controller parameters and  ${}^1\mathbf{c}^1$  for the performance vector. In that table the result  ${}^1\mathbf{c}^1$  can be compared to the performance vector  ${}^1\mathbf{c}^0$  before optimization. We observe a significant improvement in the performance vector and therefore go to step 8. .

	${}^1\mathbf{c}^0$	${}^1\mathbf{d}^1$	${}^1\mathbf{c}^1$	${}^1\mathbf{d}^2$	${}^1\mathbf{c}^2$	
oper. point 2	7.22388E-02	3.00000E-01	2.93459E-01	3.00000E-01	3.00000E-01	$c_1$ : damping
	9.28356E-01	9.28356E-01	5.26327E-01	5.26327E-01	4.78094E-01	$c_2$ : integral $(a_f)^2$
	4.73051E+00	4.73051E+00	1.97781E+00	1.97781E+00	1.72517E+00	$c_3$ : max. $ a_f $
	3.37430E-02	2.00000E-01	9.33174E-02	2.00000E-01	1.00195E-01	$c_4$ : max. $ \delta_f $
	7.37423E-01	8.10000E-01	7.33652E-01	8.10000E-01	7.18450E-01	$c_5$ : stab. margin
oper. point 4	3.00000E-01	3.00000E-01	3.00000E-01	3.00000E-01	3.00000E-01	$c_6$ : damping
	3.32162E+00	3.32162E+00	2.17297E+00	2.17297E+00	2.07490E+00	$c_7$ : integral $(a_f^2)$
	5.48661E+00	5.48661E+00	3.77899E+00	3.77899E+00	3.59112E+00	$c_8$ : max. $ a_f $
	8.01315E-02	4.50000E-01	1.64699E-01	4.50000E-01	1.76627E-01	$c_9$ : max. $ \delta_f $
	8.05835E-01	8.10000E-01	7.80087E-01	8.10000E-01	7.66160E-01	$c_{10}$ : stab. margin
	${}^1\mathbf{k}^0$	${}^1\mathbf{k}^1$		${}^1\mathbf{k}^2$		
	1.08586E-01		1.72539E-01		1.89730E-01	$k_I$
	0.00000E+00		-6.77709E+00		-7.15832E+00	$\hat{k}$
	9.00000E+00		1.05993E+01		1.12409E+01	$a_0$
	6.00000E+00		6.28741E+00		6.50198E+00	$a_1$
	4.00000E+00		4.00142E+00		4.03874E+00	$b_0$

Table 12.2: Design iterations for  $t = 1$

Step 8: evaluate the system behavior - if not satisfied goto 5.

The system behavior can be judged by means of Table 12.2 in combination with representative system simulations and an analysis. For that we have compared the results of system S2 and of the system with the optimized controller coefficients  ${}^1\mathbf{k}^1$  in Fig. 12.17. Included is the simulation and eigenvalue result of the starting vector  ${}^1\mathbf{k}^0$ . We still can not be satisfied with the almost decoupling effect and want to further improve the acceleration attenuation. Therefore, we continue in the systematic manner described in our design iteration scheme and go to step 5.

Step 5: choose the design vector  ${}^t\mathbf{d}^\nu$  for the ( $\nu$ )-th design step,  ${}^t\mathbf{c}^{\nu-1} \leq {}^t\mathbf{d}^\nu \leq {}^t\mathbf{d}^{\nu-1}$ ,  ${}^t\mathbf{d}^\nu \neq {}^t\mathbf{d}^{\nu-1}$  ( $t = 1, \nu = 2$ )

The choice of the design parameter vector  ${}^1\mathbf{d}^2$  according to  ${}^1\mathbf{c}^1 \leq {}^1\mathbf{d}^2 \leq {}^1\mathbf{d}^1$  and the respective criteria values before and after the second optimization run can be found in Table 12.2. A small improvement in the criteria values to be decreased can be observed. However, Fig. 12.18 reflects no significant success for our design goal.

	${}^2\mathbf{c}^0 = {}^1\mathbf{c}^3$	${}^2\mathbf{d}^1$	${}^2\mathbf{c}^1$	
oper. point 4	3.00000E-01	4.00000E-01	3.89105E-01	$c_1$ : damping
	7.18450E-01	8.10000E-01	5.99422E-01	$c_2$ : integral $(a_f)^2$
	4.78094E-01	4.78094E-01	1.80816E-01	$c_3$ : max. $ a_f $
	1.72517E+00	1.72517E+00	9.13054E-01	$c_4$ : max. $ \delta_f $
	1.00195E-01	2.00000E-01	1.48668E-01	$c_5$ : stab. margin
oper. point 2	3.00000E-01	4.00000E-01	3.89420E-01	$c_6$ : damping
	7.66160E-01	8.10000E-01	6.63298E-01	$c_7$ : integral $(a_f)^2$
	2.07490E+00	2.07490E+00	1.18552E+00	$c_8$ : max. $ a_f $
	3.59112E+00	3.59112E+00	2.12207E+00	$c_9$ : max. $ \delta_f $
	1.76627E-01	4.50000E-01	2.72345E-01	$c_{10}$ : stab. margin
	${}^2\mathbf{k}^0$		${}^2\mathbf{k}^1$	
	1.89730E-01		3.59216E-01	$k_I$
	-7.15832E+00		-4.38347E+00	$\hat{k}$
	1.12409E+01		1.26831E+01	$a_0$
	6.50198E+00		6.78429E+00	$a_1$
	4.03874E+00		5.12270E+00	$b_0$

Table 12.3: Iteration step for  $t = 2$

In the next iteration step with  ${}^1\mathbf{c}^2 \leq {}^1\mathbf{d}^3 \leq {}^1\mathbf{d}^2$  the performance vector could not further be improved significantly ( ${}^1\mathbf{c}^3 \approx {}^1\mathbf{c}^2$ ). Therefore, we leave the  $t$ -iteration loop for  $t = 1$  with the monotonically decreasing sequence

$${}^1\mathbf{c}^3 \approx {}^1\mathbf{c}^2 \leq {}^1\mathbf{d}^3 \leq {}^1\mathbf{d}^2 \leq {}^1\mathbf{d}^1 \quad ({}^1\mathbf{d}^3 \neq {}^1\mathbf{d}^2 \neq {}^1\mathbf{d}^1) \quad (12.3.26)$$

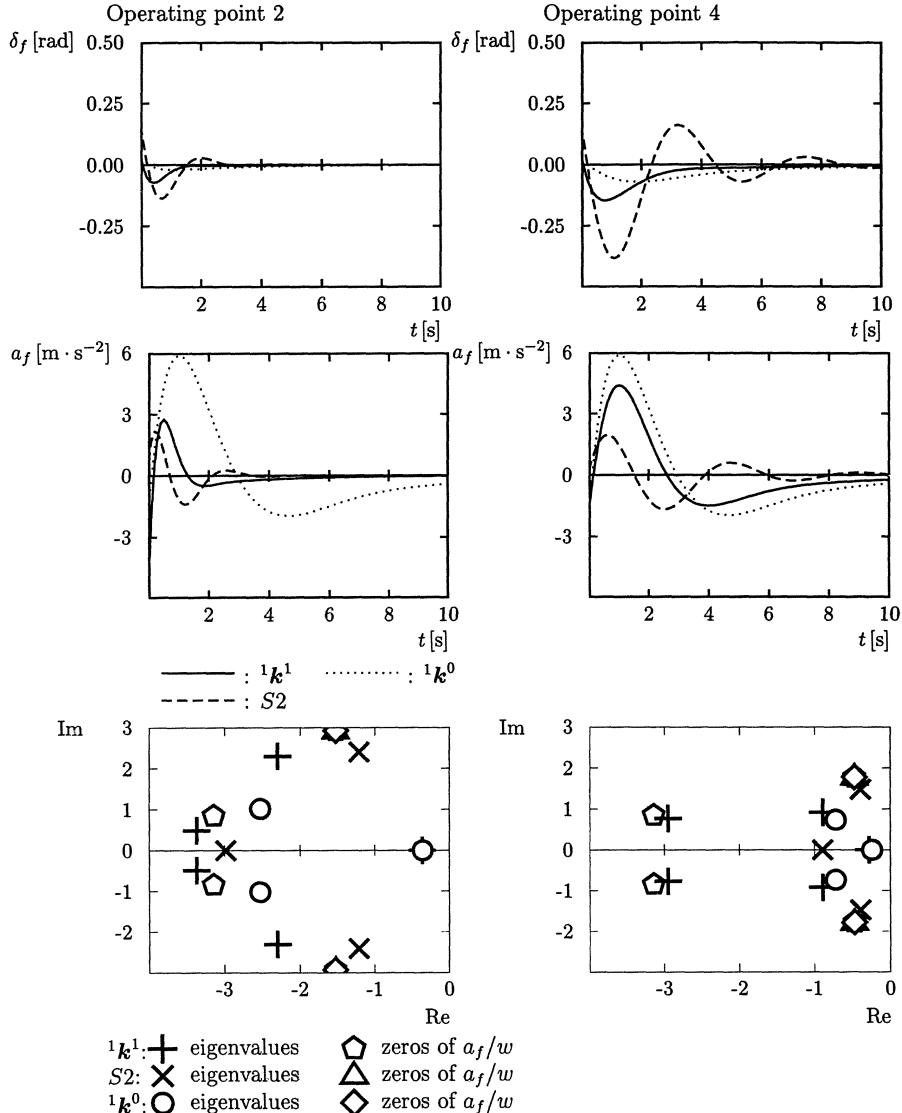
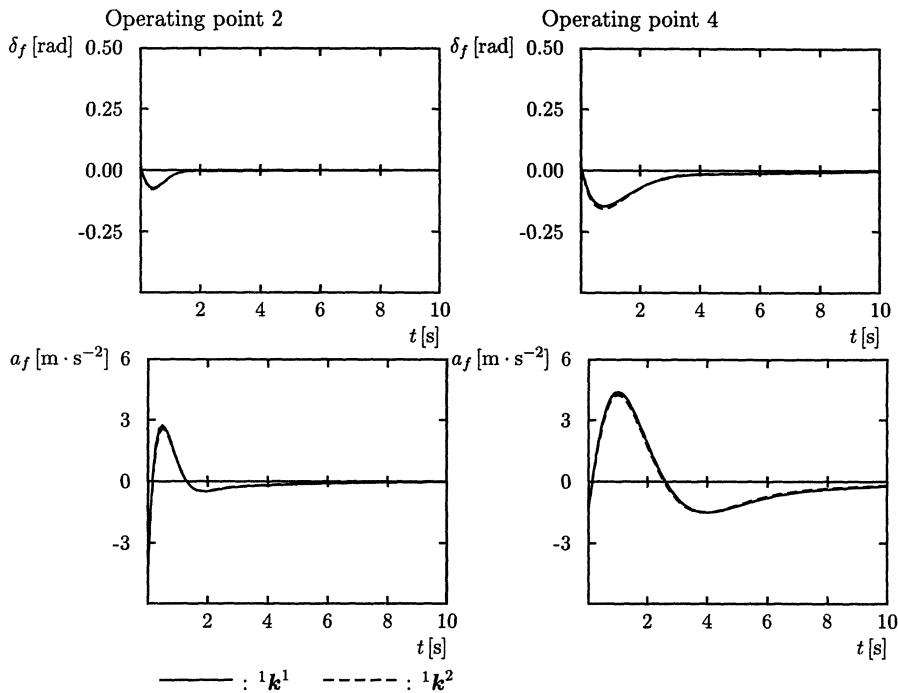
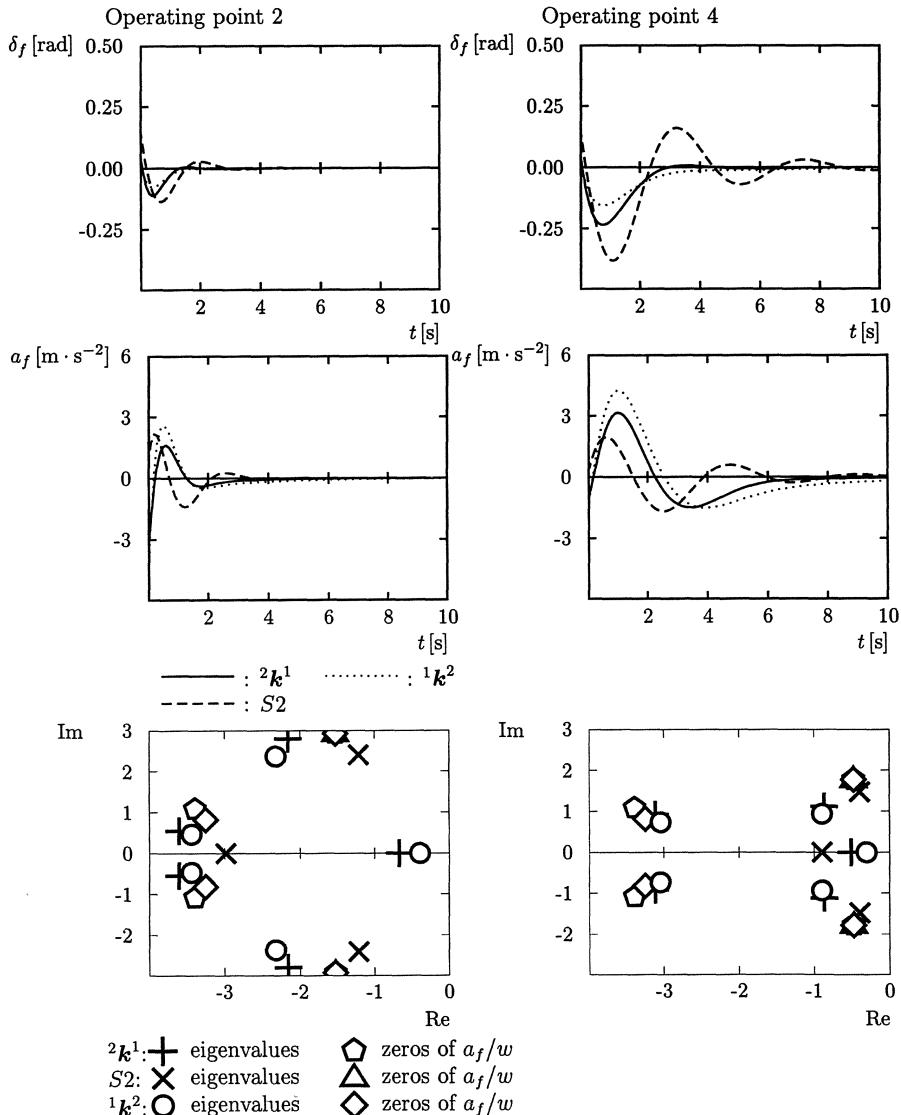


Fig. 12.17: Eigenvalue and simulation comparison after the first design iteration

Fig. 12.18: Convergence of the iteration loop for  $t = 1$ 

Step 7: if no (significant) improvement:  $t = t + 1$ ,  ${}^t\mathbf{k}^0 = {}^{t-1}\mathbf{k}^\nu$ , goto 4. (or 3. or 2. or 1.) ( $t = 1, \nu = 3$ )

We increase  $t$ ,  $t = 2$  and set  ${}^2\mathbf{k}^0 = {}^1\mathbf{k}^3$ . Observing the sequence of the eigenvalue damping criterion value  ${}^1c_6^\nu, \nu = 0, 1, 2$  in table 12.2, it becomes obvious that this damping constraint prevents a better result. From our earlier discussion that is not surprising since if we increase the damping we go further away from the zero pair responsible for the almost decoupling property. So we are interested in what happens if the design parameter values  $d_1$  and  $d_6$  for the damping constraints are relaxed. We are content with a minimal damping constraint of about 0.6 which is still better than the integrating control for S2. The iteration step is summarized in Table 12.3 and the relevant system behavior can be seen in Fig. 12.19. We observe that relaxing the eigenvalue damping constraint leads to a better almost decoupling property. Two poles tend to the invariant zero pair as could be expected from the exact decoupling control result from which our considerations have started.

Fig. 12.19: Eigenvalue and simulation comparison after iteration loop for  $t = 2$

# **Part V**

# **Appendix**

# A The Four-Wheel Car Steering Model

In this appendix some results from the automotive literature, e.g. [130, 183], are summarized from the robust control point of view.

## A.1 Single-Track Model

The essential features of car steering dynamics in a horizontal plane are described by the “single-track model” (or “two wheel model”) by Riekert and Schunck [145]. It is obtained by lumping the two front wheels into one wheel in the center line of the car, the same is done with the two rear wheels. Thereby, the car model of Fig. 1.3 is reduced to that of Fig. A.1, and the coupling with roll, pitch, and heave motions is not modelled.

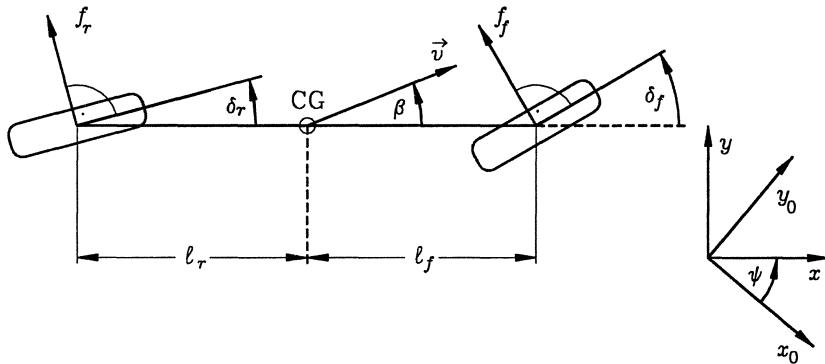


Fig. A.1: Single-track model for car steering

The angles  $\delta_f$  and  $\delta_r$  are the front and rear steering angles. The distance between the center of gravity (CG) and the front axle (resp. rear axle) is  $\ell_f$  (resp.  $\ell_r$ ) and together  $\ell = \ell_r + \ell_f$  is the wheelbase.

The velocity vector  $\vec{v}$  has the absolute value  $v = |\vec{v}|$ . The angle  $\beta$  between the vehicle center line and  $\vec{v}$  is called “vehicle sideslip angle”. In the horizontal plane of Fig. A.1

an inertially fixed coordinate system  $(x_0, y_0)$  is shown together with a vehicle fixed coordinate system  $(x, y)$  that is rotated by a “yaw angle”  $\psi$ . In the dynamic equations the yaw rate  $r := \dot{\psi}$  will appear as a state variable.

The forces transmitted from the road surface via the wheels to the car chassis are represented in Fig. A.1 by the side forces  $f_f$  and  $f_r$ . The forces in the longitudinal direction of the tires are assumed to be zero, i.e. the wheels are freely spinning. We do not model braking and the acceleration by the engine.

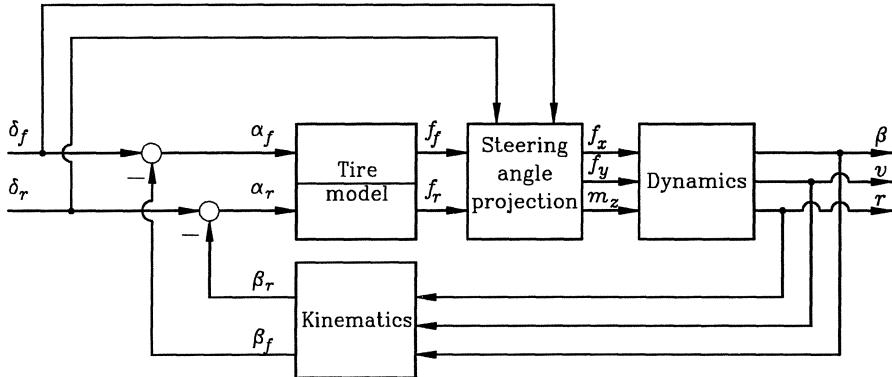


Fig. A.2: Block diagram of car steering

Fig. A.2 shows a block diagram of the model. The side forces  $f_f, f_r$  are projected through the steering angles into chassis coordinates  $(x, y)$ , where they appear as forces  $f_x, f_y$  and torque  $m_z$  around a  $z$ -axis which is pointing upward from the CG, i.e.

$$\begin{bmatrix} f_x \\ f_y \\ m_z \end{bmatrix} = \begin{bmatrix} -\sin \delta_f & -\sin \delta_r \\ \cos \delta_f & \cos \delta_r \\ \ell_f \cos \delta_f & -\ell_r \cos \delta_r \end{bmatrix} \begin{bmatrix} f_f \\ f_r \end{bmatrix} \quad (\text{A.1.1})$$

Via the dynamics model the forces cause state variables  $\beta, v, r$ . The equations of motions for three degrees of freedom in the horizontal plane are

a) longitudinal motion

$$-mv(\dot{\beta} + \dot{\psi}) \sin \beta + m\dot{v} \cos \beta = f_x \quad (\text{A.1.2})$$

b) lateral motion

$$mv(\dot{\beta} + \dot{\psi}) \cos \beta + m\dot{v} \sin \beta = f_y \quad (\text{A.1.3})$$

c) yaw motion

$$J\ddot{\psi} = m_z \quad (\text{A.1.4})$$

With  $r := \dot{\psi}$  we obtain from (A.1.2) to (A.1.4)

$$\begin{bmatrix} mv(\dot{\beta} + r) \\ m\dot{v} \\ J\dot{r} \end{bmatrix} = \begin{bmatrix} -\sin \beta & \cos \beta & 0 \\ \cos \beta & \sin \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ m_z \end{bmatrix} \quad (\text{A.1.5})$$

In the next step, the sideslip angles  $\alpha_f$  and  $\alpha_r$  at the front and rear tires are obtained by a “kinematic model” from the steering angles  $\delta_f, \delta_r$ , and from the state variables  $\beta, r$  and  $v$ . Fig. A.3 illustrates the vehicle motion around a “momentary pole” MP. The local velocity vectors in front ( $\vec{v}_f$ ) and rear ( $\vec{v}_r$ ) and at the CG ( $\vec{v}$ ) are oriented perpendicular to the connecting line to the momentary pole. The front and rear chassis sideslip angles are  $\beta_f$  and  $\beta_r$ .

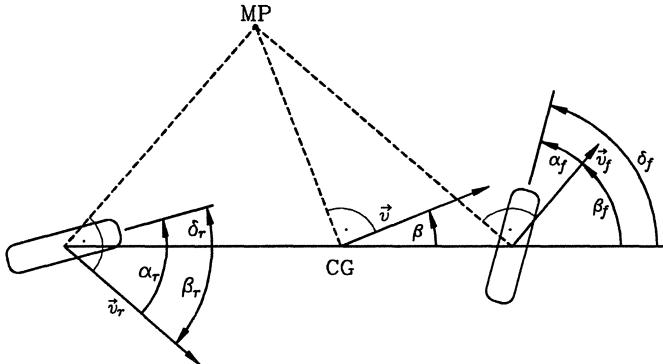


Fig. A.3: Kinematics variables

The velocity components in the direction of the longitudinal center line of the vehicle must be equal, i.e.

$$v_r \cos \beta_r = v_f \cos \beta_f = v \cos \beta \quad (\text{A.1.6})$$

The velocity components perpendicular to the center line depend on the yaw rate  $r$  as

$$\begin{aligned} v_f \sin \beta_f &= v \sin \beta + \ell_f r \\ v_r \sin \beta_r &= v \sin \beta - \ell_r r \end{aligned} \quad (\text{A.1.7})$$

The velocity terms  $v_f$  and  $v_r$  are eliminated by division by the corresponding terms from (A.1.6). Thus, the kinematic model is

$$\begin{aligned} \tan \beta_f &= \frac{v \sin \beta + \ell_f r}{v \cos \beta} = \tan \beta + \frac{\ell_f r}{v \cos \beta} \\ \tan \beta_r &= \frac{v \sin \beta - \ell_r r}{v \cos \beta} = \tan \beta - \frac{\ell_r r}{v \cos \beta} \end{aligned} \quad (\text{A.1.8})$$

and the tire sideslip angles are

$$\begin{aligned}\alpha_f &= \delta_f - \beta_f \\ \alpha_r &= \delta_r - \beta_r\end{aligned}\quad (\text{A.1.9})$$

The feedback-structured model of Fig. A.2 is now completed by the nonlinear tire model

$$\begin{aligned}f_f &= f_f(\alpha_f) \\ f_r &= f_r(\alpha_r)\end{aligned}\quad (\text{A.1.10})$$

For the rear wheel, all variables as they occur in a left turn, are illustrated by Fig. A.4. For the front wheel only the index  $r$  has to be exchanged by  $f$ .

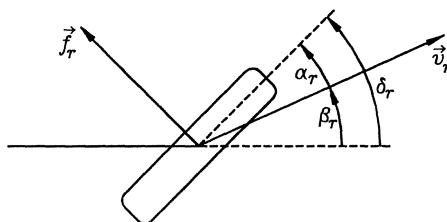


Fig. A.4: Variables of the tire model

The side force  $\vec{f}_r$  with absolute value  $f_r = |\vec{f}_r|$  is acting on the vehicle in direction of the wheel axle.

The separated rotating wheel can be studied experimentally. Under the effect of the reaction force  $-f_r$ , the wheel does not run into the direction of its longitudinal center line. It moves sideways such that a slip angle  $\alpha_r$  arises between the local velocity vector  $\vec{v}_r$  and the tire center line. The relationship between force  $f_r$  and slip angle  $\alpha_r$  can be measured at the separated wheel. A typical measured function  $f_r(\alpha_r)$  is shown in Fig. A.5.

Fig. A.5 also shows the influence of the parameter road adhesion that is lower on a wet or icy road than on dry road. For an icy road the available side force reaches a physical limit. If you turn your steering wheel quickly, then a large slip angle  $\alpha_r$  is produced. If it is larger than  $6^\circ$  to  $8^\circ$ , then a further increase of the steering command produces no additional side force. In the descending part of the curve of Fig. A.5, a further increase of the sideslip angle even has a negative effect: Less side force is generated. Of course, a control system cannot overcome such physical limits. Therefore, it is very important to design the controller such that only small tire sideslip angles occur.

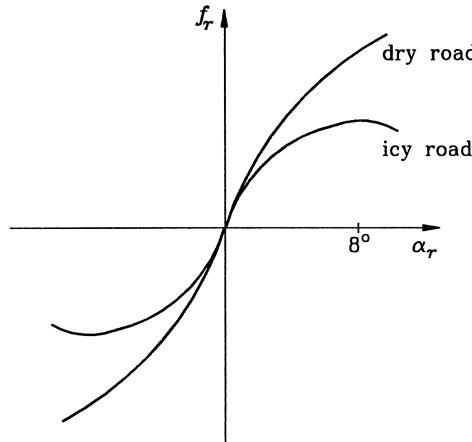


Fig. A.5: Measured dependency between side force  $f_r$  and wheel slip angle  $\alpha_r$  for dry and for icy road

## A.2 Linearized Single-Track Model

All four blocks in the diagram A.2 are nonlinear and can be linearized by additional assumptions A1) to A4).

A1) The sideslip angle  $\beta$  is assumed small. Then (A.1.5) becomes

$$\begin{bmatrix} mv(\dot{\beta} + r) \\ m\dot{v} \\ J\dot{r} \end{bmatrix} = \begin{bmatrix} -\beta & 1 & 0 \\ 1 & \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ m_z \end{bmatrix} \quad (\text{A.2.1})$$

A2) The velocity is constant,  $\dot{v} = 0$ . The second row of (A.2.1) yields  $f_x = -\beta f_y$  and with  $\beta^2 \ll 1$

$$\begin{bmatrix} mv(\dot{\beta} + r) \\ J\dot{r} \end{bmatrix} = \begin{bmatrix} f_y \\ m_z \end{bmatrix} \quad (\text{A.2.2})$$

The velocity  $v$  is now treated as an uncertain constant parameter.

A3) The steering angles  $\delta_f$  and  $\delta_r$  are small. Then from (A.1.1)

$$\begin{bmatrix} f_y \\ m_z \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \ell_f & -\ell_r \end{bmatrix} \begin{bmatrix} f_f(\alpha_f) \\ f_r(\alpha_r) \end{bmatrix} \quad (\text{A.2.3})$$

A4) The chassis sideslip angles  $\beta_f$  and  $\beta_r$  are small. Then (A.1.8) becomes

$$\begin{aligned} \beta_f &= \beta + \ell_f r/v \\ \beta_r &= \beta - \ell_r r/v \end{aligned} \quad (\text{A.2.4})$$

The nonlinear tire characteristic of Fig. A.5 is approximated by the tangent at  $\alpha_r = 0$ , i.e.

$$\begin{aligned} f_f(\alpha_f) &= c_f^* \alpha_f = c_f \mu \alpha_f, \quad \alpha_f = \delta_f - \beta_f \\ f_r(\alpha_r) &= c_r^* \alpha_r = c_r \mu \alpha_r, \quad \alpha_r = \delta_r - \beta_r \end{aligned} \quad (\text{A.2.5})$$

In the automotive literature  $c_f^*$  and  $c_r^*$  are called “cornering stiffnesses”, they depend on several uncertain parameters like normal force, longitudinal acceleration  $v$ , tire pressure, tire temperature and most importantly on the adhesion coefficient  $\mu$  between road surface and tire. Typical values of  $\mu$  are

$$\begin{aligned} \mu &= 1 && \text{dry road} \\ \mu &= 0.5 && \text{wet road} \\ \mu &= 0.15 && \text{ice} \end{aligned}$$

The same  $\mu$  is assumed for rear and front wheels. The linearized plant model follows from (A.2.2) to (A.2.5) as

$$\begin{bmatrix} mv(\dot{\beta} + r) \\ J\dot{r} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \ell_f & -\ell_r \end{bmatrix} \begin{bmatrix} c_f \mu (\delta_f - \beta - \ell_f r/v) \\ c_r \mu (\delta_r - \beta + \ell_r r/v) \end{bmatrix} \quad (\text{A.2.6})$$

The uncertain parameters in this model are mass  $m$ , moment of inertia  $J$ , velocity  $v$  and road friction coefficient  $\mu$ . (A.2.6) shows that slippery road (i.e. small  $\mu$ ) has the same effect on the dynamics as an increased  $m$  and  $J$ . Thus, we can introduce new parameters: the “virtual mass”  $\tilde{m} := m/\mu$  and the “virtual moment of inertia”  $\tilde{J} := J/\mu$ . Uncertainty in the cornering stiffness due to changing road conditions is captured by larger variations of  $\tilde{m}$  and  $\tilde{J}$ .

Solving (A.2.6) for  $\dot{\beta}$  and  $\dot{r}$  and rearranging terms yields the linear state space model

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (\text{A.2.7})$$

where

$$\begin{aligned} a_{11} &= -(c_r + c_f)/\tilde{m}v \\ a_{12} &= -1 + (c_r \ell_r - c_f \ell_f)/\tilde{m}v^2 \\ a_{21} &= (c_r \ell_r - c_f \ell_f)/\tilde{J} \\ a_{22} &= -(c_r \ell_r^2 + c_f \ell_f^2)/\tilde{J}v \\ b_{11} &= c_f/\tilde{m}v \\ b_{12} &= c_r/\tilde{m}v \\ b_{21} &= c_f \ell_f/\tilde{J} \\ b_{22} &= -c_r \ell_r/\tilde{J} \end{aligned}$$

A performance variable of particular interest is the lateral acceleration  $a_f$  of the front axle. First, we calculate the acceleration at the CG. Since we assume no acceleration in the  $\vec{v}$ -direction,  $\vec{a}_{CG}$  is perpendicular to  $\vec{v}$ , see Fig. A.6, and its absolute value is  $a_{CG} = |\vec{a}_{CG}| = v(\dot{\beta} + \psi)$ . Its component in the vehicle lateral direction  $y$  is  $a_y = v(\dot{\beta} + \psi) \cos \beta$ . For small sideslip angles  $\beta$  and with  $\dot{\psi} = r$  we have

$$a_y = a_{CG} = v(\dot{\beta} + r) \quad (\text{A.2.8})$$

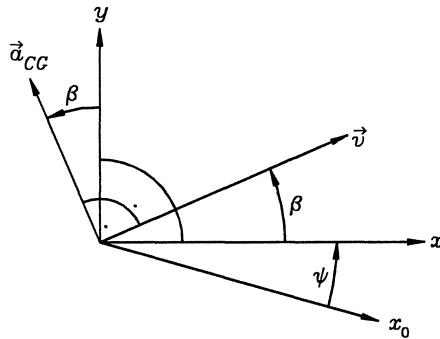


Fig. A.6:  $x_0$  = inertially fixed direction,  $x$  = vehicle center line

At the front axle a term  $\ell_f \dot{r}$  resulting from the yaw acceleration must be added, i.e.  $a_f = v(\dot{\beta} + r) + \ell_f \dot{r}$ . Substituting  $\dot{\beta}$  and  $\dot{r}$  from the differential equation (A.2.7) gives

$$a_f = (va_{11} + \ell_f a_{21})\beta + [v(1 + a_{12}) + \ell_f a_{22}]r + (vb_{11} + \ell_f b_{21})\delta_f + (vb_{12} + \ell_f b_{22})\delta_r \quad (\text{A.2.9})$$

We write (A.2.9) as an output equation for (A.2.7)

$$a_f = [c_1 \ c_2] \begin{bmatrix} \beta \\ r \end{bmatrix} + [d_1 \ d_2] \begin{bmatrix} \delta_f \\ \delta_r \end{bmatrix} \quad (\text{A.2.10})$$

where

$$\begin{aligned} c_1 &= va_{11} + \ell_f a_{21} &= -\frac{c_r + c_f}{\tilde{m}} + \frac{\ell_f(c_r \ell_r - c_f \ell_f)}{\tilde{J}} \\ c_2 &= v(1 + a_{12}) + \ell_f a_{22} &= \frac{c_r \ell_r - c_f \ell_f}{\tilde{m}v} - \frac{\ell_f(c_r \ell_r^2 + c_f \ell_f^2)}{\tilde{J}v} \\ d_1 &= vb_{11} + \ell_f b_{21} &= \frac{c_f}{\tilde{m}} + \frac{c_f \ell_f^2}{\tilde{J}} \\ d_2 &= vb_{12} + \ell_f b_{22} &= \frac{c_r}{\tilde{m}} - \frac{c_r \ell_r \ell_f}{\tilde{J}} \end{aligned}$$

## B Polynomials and Polynomial Equations

This appendix summarizes some results from classical algebra concerning polynomials in one or more variables. Only those properties of polynomials are mentioned which are important for understanding and applying the methods of robust control in this book. More details can be found in standard textbooks like [39] and [71].

### B.1 Polynomials in One Variable

A polynomial  $f(x)$  is an expression of the form

$$f(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n$$

with a nonnegative integer  $n$ . The coefficients  $a_i$ ,  $i = 1, 2, \dots, n$  are in general complex numbers. The last nonvanishing coefficient  $a_n$  determines the *degree*  $n$  of the polynomial. The term  $a_nx^n$  is called the *leading term*, and if  $a_n = 1$  the polynomial is called *monic*. The fundamental theorem of algebra (Gauss) says, that a polynomial  $f(x)$  with positive degree has a zero (or root), i.e. there exists a complex number  $x_0$  such that

$$f(x_0) = 0$$

The result of adding, subtracting and multiplying polynomials are again polynomials, whereas the division leads to rational functions. If  $g(x)$  is another polynomial then a division with remainder can be performed

$$f(x) = s(x)g(x) + r(x)$$

whereby the degree of the remainder polynomial  $r(x)$  is smaller than the degree of  $g(x)$ . If  $g(x) = x - x_0$ , a polynomial of degree one, then we may write

$$f(x) = s(x)(x - x_0) + r \tag{B.1.1}$$

with  $r$  a constant number. If  $x_0$  is a zero of  $f(x)$ , that is  $f(x_0) = 0$ , substituting  $x = x_0$  in (B.1.1) gives

$$f(x_0) = r = 0$$

that is  $f(x)$  can be divided by  $x - x_0$  without remainder

$$f(x) = s(x)(x - x_0)$$

The degree of the polynomial  $s(x)$  is  $n - 1$ . Applying the fundamental theorem to  $s(x)$  shows that  $s(x)$  has also a root. Repeating this process shows that a polynomial has exactly  $n$  zeros. Thus, if  $x_i$  are the  $n$  roots of  $f(x)$  there exists a unique factorization

$$f(x) = a_n(x - x_1)(x - x_2) \dots (x - x_n) \quad (\text{B.1.2})$$

Note that not necessarily all  $x_i$  are different, there may be *multiple* roots. Conditions for this property are given below.

If the coefficients  $a_i$  are real and if the nonreal number  $x_1 = u_1 + jv_1$  is a root of  $f(x)$  then the conjugate number  $x_2 = u_1 - jv_1$  is also a zero of  $f(x)$ . Two factors in (B.1.2) belonging to such a pair can be multiplied and the factorization (B.1.2) of  $f(x)$  has now the special form

$$f(x) = a_n(x) \prod(x - x_i) \prod(x^2 + b_jx + c_j)$$

that is the product contains only linear and quadratic terms and the factors have real coefficients. A real polynomial with odd degree has at least one real zero.

For polynomials up to degree four there exist formulas for computing the zeros. These formulas include elementary operations and root operations on the coefficients of the polynomials. For polynomials with degree five or higher, Abel and Galois have proven that it is not possible to express the zeros by the coefficients. But it is interesting to note that the zeros can be separated with arbitrary precision. This means that it is possible to determine the number of zeros in an interval, in a rectangle (or even a more general region) of the complex plane by rational operations on the coefficients. This basic property is the starting point for the classical methods of finding the zeros. Some of them can be found in [169].

Another important property of polynomials is that the zeros are continuous functions of the polynomial coefficients. Consider a polynomial  $f(x)$  where the leading coefficient  $a_n$  is varying, whereas all other coefficients are fixed. If  $a_n$  is changing the zeros are moving continuously in the complex plane. If  $a_n$  tends to zero than the modulus of one zero grows unlimited. A common practice is to say that this zero goes to infinity and if  $a_n$  reaches zero than the corresponding polynomial has a root at infinity. Besides the finite zeros of a polynomial there may exist zeros at infinity depending on how many coefficients  $a_n, a_{n-1}, \dots$  vanish.

Now consider two polynomials  $f(x)$  and  $g(x)$ . The question whether they have common zeros is answered by the *resultant* of two (fixed) polynomials.

*Definition 2.1.* The *resultant* of two polynomials

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0, \quad a_n \neq 0$$

$$g(x) = b_m x^m + b_{m-1} x^{m-1} + \dots + b_1 x + b_0, \quad b_m \neq 0$$

is the determinant of the resultant matrix

$$\mathbf{R}(f, g) = \begin{bmatrix} a_n & a_{n-1} & a_{n-2} & \cdots & a_0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & a_n & a_{n-1} & \cdots & a_1 & a_0 & 0 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & a_n & a_{n-1} & a_{n-2} & \cdots & a_2 & a_1 & a_0 \\ b_m & b_{m-1} & b_{m-2} & \cdots & b_1 & b_0 & 0 & 0 & \cdots & 0 \\ 0 & b_m & b_{m-1} & \cdots & b_2 & b_1 & b_0 & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & & \vdots \\ 0 & \cdots & 0 & b_m & b_{m-1} & b_{m-2} & \cdots & b_2 & b_1 & b_0 \end{bmatrix}$$

□

*Theorem 2.1.*

Two polynomials  $f, g$  have a common zero if and only if

$$\text{Res}(f, g) := \det \mathbf{R}(f, g) = 0$$

□

The order of the resultant matrix is  $(m+n) \times (m+n)$ . The coefficients of the polynomial  $f$  appear in  $m$  (the degree of  $g$ ) rows and the coefficients of  $g$  in  $n$  (the degree of  $f$ ) rows. This theorem also answers the question whether a polynomial  $f$  has multiple roots. If  $f$ , for instance, has a double root  $x_0$ , then it can be factorized in the form  $f(x) = a_n(x - x_0)^2 g(x)$ . The first derivative  $f'(x) = a_n(2(x - x_0)g(x) + (x - x_0)^2 g'(x))$  vanishes also for  $x = x_0$ . Thus,  $x_0$  is also a zero of  $f'(x)$ . This is also true for a higher multiplicity of the zero. In general, if  $x_0$  is a zero of  $f$  of multiplicity  $k$ , then  $x_0$  is a zero of  $f'$  of multiplicity  $k-1$ , a zero of  $f''$  of multiplicity  $k-2$  and a zero of  $f^{(k-1)}$  of multiplicity 1, that is a single zero. Apply Theorem 2.1 with  $f$  and  $g = f'$ , the first derivative, the resultant  $R(f, f')$  determines whether  $f$  has a double root. This special determinant, more precisely

$$D(f) := (-1)^{\frac{n(n-1)}{2}} \text{Res}(f, f')/a_n$$

is referred as the *discriminant* of the polynomial  $f(x)$ . It coincides with the discriminant which is known from the quadratic equation. From this definition follows another representation

$$D(f) = a_n^{2n-2} \prod_{1 \leq i < j \leq n} (x_i - x_j)^2$$

The resultant matrix constructed in Chapter 4 helps to decide whether a polynomial  $p$  has a root on the imaginary axis. The polynomials  $f$  and  $g$  are constructed from the coefficients of the original polynomial  $p$ .

## B.2 Polynomials in Two Variables

A polynomial in two variables  $x$  and  $y$  is an expression of the form

$$f(x, y) = \sum a_{ij}x^i y^j$$

Important to note is that only a finite number of coefficients  $a_{ij}$  are different from zero. This polynomial can be written either as a polynomial in  $x$

$$f(x, y) = a_n(y)x^n + a_{n-1}(y)x^{n-1} + \dots + a_1(y)x + a_0(y)$$

or as polynomial in  $y$

$$f(x, y) = b_m(x)y^m + b_{m-1}(x)y^{m-1} + \dots + b_1(y)y + b_0(x)$$

The maximal sum  $i + j$  with nonvanishing coefficient  $a_{ij}$  is called the *degree* of the polynomial. A zero of  $f$  is a pair of numbers  $(x_0, y_0)$  with  $f(x_0, y_0) = 0$ .

## B.3 Some Properties of Two-dimensional Curves

A polynomial equation  $f(x, y) = 0$  may be viewed as a (real) curve in the  $(x, y)$ -plane. For plotting this curve, one variable, for example  $x$ , has to be gridded and the zeros of  $f(x^*, y) = 0$  yield the second component. Remember that some zeros may be nonreal. Thus, the number of real solution pairs  $(x, y)$  may vary. The tangent at the point  $P = (x_0, y_0)$  is with  $f_x := \frac{\partial f}{\partial x}$  and  $f_y := \frac{\partial f}{\partial y}$  defined by

$$f_x(x_0, y_0)(x - x_0) + f_y(x_0, y_0)(y - y_0) = 0$$

If both partial derivatives vanish for  $(x_0, y_0)$  this definition fails and such points are called *singular points*. With  $f_{xx} := \frac{\partial^2 f}{\partial x^2}$ ,  $f_{xy} := \frac{\partial^2 f}{\partial x \partial y}$  and  $f_{yy} := \frac{\partial^2 f}{\partial y^2}$  the slopes  $m$  of the tangents at  $P$  are given by the zeros of the quadratic equation  $f_{xx} + 2f_{xy}m + f_{yy}m^2 = 0$ . Depending on the sign of the discriminant there exist three types of singular points. If both zeros are real ( $D > 0$ ), there are two real slopes and  $P$  is called a *double point*, the curve intersects itself. For  $D = 0$  the slopes coincide and the curve has a *cusp*. If  $D < 0$ , the slopes are complex and  $P$  is called *isolated point*. In every neighborhood of  $P$  there are no further (real) points of the curve. It may happen that the second partial derivatives also vanish at  $P$ , the singularity of  $P$  is of higher order. The *curvature*  $\rho$  at  $P$  is determined by

$$\rho := \frac{-f_y^2 f_{xx} + 2f_x f_y f_{xy} - f_x^2 f_{yy}}{(f_x^2 + f_y^2)^{3/2}}$$

An *inflection point* occurs for  $\rho = 0$ . The curvature of the boundary of a convex region does not change its sign. The Mikhailov plot of a stable polynomial (see Chapter 4) is

a curve without inflection points. The asymptotes (nonparallel to the  $y$ -axis) of a curve can be determined as follows. Let  $y = mx + b$  be the equation of the asymptote. The equation  $f(x, mx + b) = 0$  is a polynomial in  $x$  with coefficients depending on  $m$  and  $b$  and let  $n$  the degree of the polynomial. The coefficient of the leading term is only dependent on  $m$ . Setting this term to zero yields the slope  $m$  of the asymptote. The second parameter  $b$  can be calculated by substituting  $m$  in the coefficient of  $x^{n-1}$  and the condition for  $b$  is that this coefficient also vanishes.

Sometimes a curve is defined by a parametric representation:

$$x = x(\alpha), \quad y = y(\alpha), \quad \alpha \in [\alpha^-; \alpha^+]$$

where  $x(\alpha)$  and  $y(\alpha)$  are rational functions. The maximal degree of the numerator polynomials resp. denominator polynomials determines the degree of the curve. It should be mentioned that a parametric representation of a curve can always be transformed into the parameterfree representation (see next section), but there exist curves for which a parameterfree representation does not exist. The parameter values for which the curve has asymptotes are given by the zeros of the denominator polynomials of  $x(\alpha)$  and  $y(\alpha)$ . Examples of parametrized curves mentioned in this book are the Mikhailov plot, the Nyquist plot, the Popov plot and the complex root boundary in the affine case.

## B.4 Two Polynomials in Two Variables

Consider now two equations in two variables

$$f(x, y) = 0, \quad g(x, y) = 0$$

where  $f$  and  $g$  are polynomials. The zeros of this system of two equations are those pairs  $(x^*, y^*)$  for which both  $f(x^*, y^*) = 0$  and  $g(x^*, y^*) = 0$ . If  $f$  and  $g$  are interpreted as (real) curves then the zeros are the intersection points of the two curves. It can be assumed that  $f$  and  $g$  have no common nonconstant factor  $h(x, y)$ . (If such a factor exists it can be found with the Euclidean algorithm [169] and the two equations can be divided by this factor). There arise two questions: Does there exist solutions and how these solutions can be found? The first question is answered by

*Theorem 2.2. (Bezout)*

Two polynomial equations  $f$  and  $g$  (with no common factor) with degree  $n$  resp.  $m$  have exactly  $nm$  solution pairs.

□

Again it should be pointed out that the solutions may be complex, the solutions may be multiple or some solutions may be at infinity.

Finding the solution is similar to solving a linear system of equations. In a first step one tries to eliminate one variable resulting in a polynomial which contains only one variable. By the Theorem of Bezout its degree is in general  $nm$ , if the degree has dropped than there must be one or more solutions at infinity. In a second step the zeros of the resulting polynomial are substituted in  $f$  and  $g$  and the zeros of the greatest common divisor of  $f$  and  $g$  (again a polynomial) determine the other components of the solution pairs. The elimination procedure can be carried out with the resultant method. The coefficients  $a_i$  and  $b_i$  are no longer fixed numbers. If, for example,  $y$  should be eliminated, then  $f$  and  $g$  are written as polynomials in  $y$  with coefficient polynomials in  $x$  and the corresponding resultant is written as

$$R(x) = \text{Res}_y(x, y)$$

Instead of finding the greatest common divisor of  $f$  and  $g$  for fixed  $x = x^*$  the classical method can be used where the second component  $y = y^*$  is expressed by two determinants, see [41].

The more elegant and modern method is using Gröbner bases [42]. This algorithm is not suited for paper and pencil calculations. But it is implemented in software packages for symbolic computations like REDUCE or MATHEMATICA. The algorithm manipulates the two polynomials  $f$  and  $g$  and yields in general two polynomials. The first one again is the resultant, i.e. a polynomial of degree  $mn$  in one variable, say  $x$ . The second polynomial has the form

$$h(y) + \sum_{i=0}^{nm-1} c_i x^i$$

where  $h(y)$  is a polynomial in  $y$ . Substituting the zeros  $x_i^*$  of the resultant in this polynomial yields the second component  $y^*$ . If  $h(y)$  has degree one then exactly one  $y_i^*$  belongs to each  $x_i^*$ . If the degree of  $h$  is greater than one then there are solutions with the same  $x$ -component but different  $y$ -components.

The transformation of the parameter representation of a curve to the parameterfree representation can also be performed with the resultant method. If  $x(\alpha) = n_x(\alpha)/d_x(\alpha)$  and  $y(\alpha) = n_y(\alpha)/d_y(\alpha)$  then the polynomials  $xd_x - n_x$  and  $yd_y - n_y$  are written as polynomials in  $\alpha$  with coefficients dependent on  $x$  and  $y$  and the parameterfree representation is

$$f(x, y) = \text{Res}_\alpha(xd_x - n_x, yd_y - n_y) = 0$$

This representation may comprise more points than the original one because the possible restriction on the parameter  $\alpha$  gets lost in the elimination procedure.

Another application of the resultant method is the determination of the envelope of a set of curves  $f(x, y, a) = 0$  dependent on the parameter  $a$ . If the envelope exists then its equation results from the elimination of  $a$  of the two polynomials

$$f(x, y, a) = 0, \quad \frac{\partial f(x, y, a)}{\partial a} = 0$$

## B.5 Several Polynomials in Several Variables

Only two remarks are made in this final section. The solution structure of a system of polynomial equations in more variables is much more complicated than in the case of a linear system of equations. For more details and examples see [42]. In general, it is not possible to determine the structure without actually solving the system.

Further, the method of Gröbner bases is practically limited to systems with at most five or six variables. A simple example will demonstrate this fact. The extension of Bezout's theorem for a system with three polynomials with degree  $m$ ,  $n$  resp.  $k$  in three variables shows that there exist in general  $m \times n \times k$  solution vectors. This means that the resulting polynomial containing only one variable has the degree  $m \times n \times k$ .

## Bibliography

- [1] "Flying qualities of piloted airplanes." MIL-F-8785 B (ASG), August 1969.
- [2] J. Ackermann, "Der Entwurf linearer Regelungssysteme im Zustandsraum," *Regelungstechnik*, vol. 20, pp. 297–300, 1972.
- [3] J. Ackermann, "Parameter space design of robust control systems," *IEEE Trans. on Automatic Control*, vol. AC-25, pp. 1058–1072, 1980.
- [4] J. Ackermann, "Robustness against sensor failures," *Automatica*, vol. 20, pp. 211–215, 1984.
- [5] J. Ackermann, *Sampled-data control systems: analysis and synthesis, robust system design*. Berlin: Springer, 1985.
- [6] J. Ackermann, "Robust car steering by yaw rate control," in *Proc. IEEE Conf. Decision and Control*, (Honolulu), pp. 2033–2034, 1990.
- [7] J. Ackermann, "Verfahren zum Lenken von Straßenfahrzeugen mit Vorder- und Hinterradlenkung." Patent No. P 4028 320.8 Deutsches Patentamt München, Sept. 1990.
- [8] J. Ackermann, "Uncertainty structures and robust stability analysis," in *Proc. First European Control Conference*, (Grenoble), pp. 2318–2327, 1991.
- [9] J. Ackermann, "Does it suffice to check a subset of multilinear parameters in robustness analysis?," *IEEE Trans. on Automatic Control*, vol. 37, pp. 487–488, 1992.
- [10] J. Ackermann, "Robust yaw damping of cars with front and rear wheel steering," in *Proc. IEEE Conf. Decision and Control*, (Tuscon, Arizona), 1992.
- [11] J. Ackermann, "Robust nonlinear decoupling and yaw stabilization of 4 WS cars," in *Proc. 12th IFAC World Congress*, (Sydney), 1993.
- [12] J. Ackermann, "Robust nonlinear decoupling and yaw stabilization of four-wheel steering cars," *Automatica*, 1993. accepted for publication.

- [13] J. Ackermann and B. Barmish, "Robust Schur stability of a polytope of polynomials," *IEEE Trans. on Automatic Control*, vol. 33, pp. 984–986, 1988.
- [14] J. Ackermann and H. Hu, "Robustness of sampled-data control systems with uncertain physical plant parameters," *Automatica*, vol. 27, pp. 705–710, 1991.
- [15] J. Ackermann, D. Kaesbauer, and R. Münch, "Robust  $\Gamma$ -stability analysis in a plant parameter space," *Automatica*, vol. 27, pp. 75–85, 1991.
- [16] J. Ackermann, D. Kaesbauer, and W. Sienel, "Design by search," in *Proc. First IFAC Symposium on Design Methods of Control Systems*, (Zürich), 1991.
- [17] J. Ackermann and W. Sienel, "Robust control for automatic steering," in *Proc. American Control Conference*, (San Diego), pp. 795–800, 1990.
- [18] J. Ackermann and W. Sienel, "What is a 'large' number of parameters in robust systems," in *Proc. IEEE Conf. Decision and Control*, (Honolulu), pp. 3496–3497, 1990.
- [19] J. Ackermann and W. Sienel, "On the computation of value sets for robust stability analysis," in *Proc. First European Control Conference*, (Grenoble), pp. 2318–2327, 1991.
- [20] J. Ackermann and S. Türk, "A common controller for a family of plant models," in *Proc. 21st IEEE Conf. Decision and Control*, (Orlando), pp. 240–244, 1982.
- [21] J. Anagnos, C. Desoer, and R. Minichelli, "Generalized Nyquist test for robust stability: Frequency domain generalizations of Kharitonov's theorem," in *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.), pp. 79–96, New York: Plenum Press, 1989.
- [22] B. Anderson, E. Jury, and M. Mansour, "On robust Hurwitz polynomials," *IEEE Trans. on Automatic Control*, vol. 32, pp. 909–912, 1987.
- [23] K. Åström and B. Wittenmark, *Computer controlled systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1984.
- [24] F. Bailey and C. Hui, "A fast algorithm for computing parametric rational functions," *IEEE Trans. on Automatic Control*, vol. 34, pp. 1209–1212, 1989.
- [25] G. Balas, J. Doyle, K. Glover, A. Packard, and R. Smith, *The  $\mu$  analysis and synthesis toolbox*. The MathWorks, 1991.
- [26] J. Bals, *Aktive Schwingungsdämpfung flexibler Strukturen*. PhD thesis, Universität Karlsruhe (TH), 1989.
- [27] J. Bals, "A hyperstability/multiobjective optimization approach for active flexible structures," in *IFAC Symposium on Design Methods for Control Systems*, vol. 2, (Zürich), pp. 621–626, 1991.

- [28] B. Barmish, "New tools for robustness analysis," in *Proc. IEEE Conf. Decision and Control*, (Austin), pp. 1–6, 1988.
- [29] B. Barmish, "A generalization of Kharitonov's four-polynomial concept for robust stability problems with linearly dependent coefficient perturbations," *IEEE Trans. on Automatic Control*, vol. 34, no. 2, pp. 157–165, 1989.
- [30] B. Barmish, *New tools for robustness of linear systems*. New York: Macmillan, to appear 1993.
- [31] B. Barmish, J. Ackermann, and H. Hu, "The tree structured decomposition: a new approach to robust stability analysis," in *Proc. Conf. on Information Sciences and Systems*, (Princeton), 1990.
- [32] B. Barmish, C. Hollot, F. Kraus, and R. Tempo, "Extreme point results for robust stabilization of interval plants with first order compensators," in *Proc. American Control Conference*, (San Diego), 1990.
- [33] B. Barmish and H. Kang, "A survey of extreme point results of robust control systems," *Automatica*, vol. 29, pp. 13–35, 1993.
- [34] B. Barmish and K. Wei, "Simultaneous stabilizability of single input-single output systems," in *Proc. Int. Symposium on Mathematical Theory of Networks and Systems*, (Stockholm), 1985.
- [35] A. Bartlett, C. Hollot, and Huang-Lin, "Root locations of an entire polytope of polynomials: it suffices to check the edges," *Mathematics of Control, Signals and Systems*, vol. 1, pp. 61–71, 1988.
- [36] R. Berger, J. Hess, and D. Anderson, "Compatibility of maneuver load control and relaxed static stability applied to military aircraft," AFFDL-TR-73-33, Air Force Flight Dynamics Laboratory, 1973.
- [37] S. Bialas, "A necessary and sufficient condition for the stability of convex combinations of stable polynomials or matrices," *Bulletin of the Polish Academy of Sciences*, vol. 33, pp. 473–480, 1985.
- [38] M. Biehler, "Sur une classe d'équations algébriques dont toutes les racines sont réelles," *J. Reine Angewandte Mathematik*, vol. 87, pp. 350–352, 1879.
- [39] G. Birkhoff and S. M. Lane, *A survey of modern algebra*. New York: The Macmillan Company, 1965.
- [40] N. Bose and E. Zeheb, "Kharitonov's theorem and stability test of multi-dimensional digital filters," in *IEE Proceedings*, vol. G-133, pp. 187–190, 1986.
- [41] A. Brill, *Vorlesungen über ebene algebraische Kurven und algebraische Funktionen*. Braunschweig: Vieweg, 1925.

- [42] B. Buchberger, “Gröbner bases: an algorithmic method in polynomial ideal theory,” in *Recent trends in multidimensional system theory* (N. Bose, ed.), Reidel, 1985.
- [43] F. Cellier, *Continuous system modelling*. New York: Springer, 1991.
- [44] H. Chapellat and S. Bhattacharyya, “A generalization of Kharitonov’s theorem: Robust stability of interval plants,” *IEEE Trans. on Automatic Control*, vol. 34, pp. 306–311, 1989.
- [45] H. Chapellat, M. Dahleh, and S. Bhattacharyya, “On robust nonlinear stability of interval control systems,” *IEEE Transaction on Automatic Control*, vol. 36, no. 1, pp. 59–67, 1991.
- [46] J. Cieslik, “On possibilities of the extension of Khartitonov’s stability test for interval polynomials to the discrete-time case,” in *IEEE Trans. on Automatic Control*, vol. AC-32, pp. 237–238, 1987.
- [47] A. Cohn, “Über die Anzahl der Wurzeln einer algebraischen Gleichung in einem Kreise,” *Mathematische Zeitschrift*, vol. 14, pp. 110–148, 1922.
- [48] L. Cremer, “Ein neues Verfahren zur Beurteilung der Stabilität linearer Regelungssysteme,” *ZAMM*, p. 161, 1947.
- [49] W. Darenberg, “Automatische Spurführung von Kraftfahrzeugen,” *Automobil-Industrie*, pp. 155–159, 1987.
- [50] S. Dasgupta, “Perspectives on Kharitonov’s theorem: a view from the imaginary axis,” tech. rep., Dept. Elect. Comp. Eng., University of Iowa, Iowa City, 1987.
- [51] S. Dasgupta and A. Bhagwat, “Conditions for designing spr transfer functions for adaptive output error identification,” *IEEE T-CAS*, vol. 34, 1987.
- [52] A. Debowski and A. Kurylowicz, “Simultaneous stabilization of linear single-input/single-putput plants,” *Int. Journal of Control*, vol. 44, no. 5, pp. 1257–1264, 1986.
- [53] R. deGaston and M. Safonov, “Exact calculation of the multiloop stability margin,” *IEEE Trans. on Automatic Control*, vol. 33, pp. 156–171, 1988.
- [54] C. Desoer, R.-W. Liu, J. Murray, and R. Saeks, “Feedback system design: the fractional representation approach to analysis a synthesis,” *IEEE Trans. on Automatic Control*, vol. 25, pp. 399–412, 1980.
- [55] E. Dickmanns and T. Christians, “Relative 3D state estimation for autonomous visual guidance of road vehicles,” *Intelligent Autonomous Systems*, vol. IAS-2, 1989.
- [56] T. Djaferis, “Stabilization of systems with real parameter variations,” in *Proc. IEEE Conf. Decision and Control*, (Tampa), pp. 1870–1871, 1989.

- [57] E. Donges, R. Aufhammer, P. Fehrer, and T. Seidenfuß, "Funktion und Sicherheitskonzept der Aktiven Hinterachs kinematik von BMW," *Automobiltechnische Zeitschrift*, vol. 10, pp. 580–587, 1990.
- [58] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solution to standard  $h_2$  and  $h_\infty$  control problems," *IEEE Trans. on Automatic Control*, vol. 34, pp. 831–847, 1989.
- [59] Y. El-Deen and A. Seireg, "Mechatronics for cars: integrating machines and electronics to prevent skidding on icy roads," *Computers in Mechanical Engineering*, pp. 10–22, 1987.
- [60] E. Emre, "Simultaneous stabilization with fixed closed-loop characteristic polynomial," *IEEE Trans. on Automatic Control*, vol. 28, pp. 103–104, 1983.
- [61] B. Etkin, *Dynamics of atmospheric flight*. New York: Wiley, 1972.
- [62] W. Evans, "Graphical analysis of control systems," *Trans. AIEE*, vol. 67, no. 2, pp. 547–551, 1948.
- [63] S. Faedo, "Un nuovo problema di stabilità per le equazioni algebriche a coefficienti reali," *Ann. Scuola Norm. Super Pisa, Sci. Fis. Mat.*, vol. 7, pp. 53–63, 1953.
- [64] A. Fam and J. Meditch, "A canonical parameter space for linear system design," *IEEE Trans. on Automatic Control*, vol. 23, pp. 454–458, 1978.
- [65] R. Finsterwalder, "A parallel coordinate editor as a visual decision aid in a multi-objective concurrent control engineering environment," in *Proc. 5th IFAC/IMACS symposium on Computer Aided Design in Control Systems, University of Wales, Swansea, UK*, pp. 118–122, Pergamon Press, Oxford, 1991.
- [66] R. Fletcher and M. Powell, "A rapidly convergent descent method for minimization," *Computer J.*, vol. 6, pp. 163–171, 1963.
- [67] G. Franklin and J. Powell, *Digital control*. Reading, Massachusetts: Addison-Wesley, 1980.
- [68] S. Franklin and J. Ackermann, "Robust flight control: a design example," *AIAA J. Guidance and Control*, vol. 4, pp. 597–605, 1981.
- [69] R. Frazer and W. Duncan, "On the criteria for the stability of small motions," in *Proc. Royal Society A*, vol. 124, pp. 642–654, 1929.
- [70] M. Fu, "Polytopes of polynomials with zeros in a prescribed region: New criteria and algorithms," *Systems and Control Letters*, vol. 15, pp. 125–145, 1990.
- [71] W. Fulton, *Algebraic Curves*. Reading, Massachusetts: Addison-Wesley, 1989.
- [72] F. Gantmacher, *The theory of matrices*. New York: Chelsea, 1959.

- [73] F. Gantmacher, *Matrizenrechnung*. Berlin: VEB Deutscher Verlag der Wissenschaften, 1965.
- [74] B. Ghosh, “Some new results on the simultaneous stabilizability of a family of single input, single output systems,” *Systems and Control Letters*, vol. 6, pp. 39–45, 1985.
- [75] B. Ghosh, “An approach to simultaneous system design. Part i: Semialgebraic geometric methods,” *SIAM Journal of Control and Optimization*, vol. 24, pp. 480–496, 1986.
- [76] B. Ghosh, “An approach to simultaneous system design. Part ii: Nonswitching gain and dynamic feedback by algebraic methods,” *SIAM Journal of Control and Optimization*, vol. 26, pp. 919–963, 1988.
- [77] B. Ghosh and C. Byrnes, “Simultaneous stabilization and simultaneous pole-placement by nonswitching dynamic compensators,” *IEEE Trans. on Automatic Control*, vol. 28, pp. 735–741, 1983.
- [78] E. Gilbert, “Controllability and observability in multivariable control systems,” *SIAM Journal of Control and Optimization*, pp. 128–151, 1963.
- [79] G. Grübel, “A coherent technology for optimization-based control system design,” in *9th IFAC Workshop Control Applications of Optimization*, (München), 1992.
- [80] G. Grübel, D. Joos, D. Kaesbauer, and R. Hillgren, “Robust back-up stabilization for artificial-stability aircraft,” in *Proc. 14th ICAS Congress*, (Toulouse), 1984.
- [81] P. Gutman, C. Baril, and L. Neumann, “An image processing approach for computing value sets of uncertain transfer functions,” in *Proc. IEEE Conf. Decision and Control*, (Honolulu), pp. 1224–1229, 1990.
- [82] C. Hermite, “Sur le nombre des racines d’une équation algébrique comprise entre des limites données,” *J. Reine Angewandte Mathematik*, vol. 52, pp. 39–51, 1852. English translation Int. Journal of Control 1977.
- [83] C. Hollot and A. Bartlett, “Some discrete time counterparts to Kharitonov’s stability criterion for uncertain systems,” *IEEE Trans. on Automatic Control*, vol. 31, no. 4, pp. 355–356, 1986.
- [84] C. Hollot, F. Kraus, R. Tempo, and B. Barmish, “Extreme point results for robust stabilization of interval plants with first order compensator,” in *Proc. American Control Conference*, (San Diego), pp. 2533–2538, 1990.
- [85] I. Horowitz, *Synthesis of feedback systems*. New York: Academic Press, 1963.
- [86] I. Horowitz, “Quantitative feedback theory,” in *IEE Proc.*, vol. 129 D, pp. 215–226, 1982.

- [87] I. Horowitz, "Survey of quantitative feedback theory (qft)," *Int. Journal of Control*, vol. 53, pp. 255–291, 1991.
- [88] G. Howitt and R. Luus, "Simultaneous stabilization of linear single-input systems by linear state feedback control," *Int. Journal of Control*, vol. 54, no. 4, pp. 1015–1030, 1991.
- [89] H. Hu, C. Hollot, R. Tempo, and J. Ackermann, "Absence of extreme-point results in sampled-data control systems," in *Proc. First European Control Conference*, (Grenoble), pp. 1356–1359, 1991.
- [90] A. Hurwitz, "Über die Bedingungen, unter welchen eine Gleichung nur Wurzeln mit negativen reellen Teilen besitzt," *Mathematische Annalen*, vol. 46, pp. 273–284, 1895.
- [91] H. Inoue, H. Harada, and Y. Yokoya, "Allradlenksystem im Toyota Soarer." Presented at Tagung Allradlenksysteme bei Personenwagen, Haus der Technik, Essen, Dec. 1991.
- [92] H.-D. Joos, "Automatic evolution of a decision-supporting design project database in concurrent control engineering," in *Proc. 5th IFAC/IMACS symposium on Computer Aided Design in Control Systems, University of Wales, Swansea, UK*, pp. 113–117, Pergamon Press, Oxford, 1991.
- [93] H.-D. Joos and M. Otter, "Control engineering data structures for concurrent engineering," in *Proc. 5th IFAC/IMACS symposium on Computer Aided Design in Control Systems, University of Wales, Swansea, UK*, pp. 107–112, Pergamon Press, Oxford, 1991.
- [94] E. Jury, *Theory and application of the z-transform method*. New York: John Wiley, 1964.
- [95] E. Jury and T. Pavlidis, "Stability and aperiodicity constraints for systems design," *IEEE Trans. on Circuit Theory*, pp. 137–141, 1963.
- [96] D. Kaesbauer, *Robuster Reglerentwurf durch Kontraktion eines Polgebiets*. PhD thesis, Institut für Elektrotechnik, Technische Universität Graz, 1986.
- [97] D. Kaesbauer, "On robust stability of polynomials with polynomial parameter dependency: two/three parameter case," *Automatica*, vol. 29, no. 1, pp. 215–217, 1993.
- [98] D. Kaesbauer and J. Ackermann, "The distance from stability or gamma-stability boundaries," in *Proc. 11th IFAC Congress*, (Tallinn), pp. 130–134, 1990.
- [99] T. Kailath, *Linear systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1980.
- [100] R. Kalman, "On the general theory of control systems," in *First IFAC Congress*, (Moscow), pp. 481–492, 1960.

- [101] R. Kalman, “Mathematical description of linear dynamical systems,” *SIAM Journal of Control and Optimization*, pp. 152–192, 1963.
- [102] J. Kasselmann and T. Keranen, “Adaptive steering,” *Bendix Technical Journal*, vol. 2, pp. 152–192, 1969.
- [103] P. Khargonekar, I. Petersen, and K. Zhou, “Robust stabilization of uncertain linear systems: quadratic stability and the  $h_\infty$  control theory,” *IEEE Trans. on Automatic Control*, vol. 34, pp. 831–847, 1989.
- [104] P. Khargonekar and A. Tannenbaum, “Non-Euklidian metrics and the robust stabilization of systems with parameter uncertainty,” *IEEE Trans. on Automatic Control*, vol. AC-30, pp. 1005–1013, 1985.
- [105] V. Kharitonov, “On a generalization of a stability criterion,” in *Seria fiziko-matematicheskaiia*, vol. 1, pp. 53–57, Izvestiia Akademii nauk Kazakhskoi SSR, 1978.
- [106] V. Kharitonov, “Asymptotic stability of an equilibrium position of a family of systems of linear differential equations,” *Differential Equations*, vol. 14, no. 3, pp. 26–35, 1979.
- [107] H. Kimura, “Robust stabilization for a class of transfer functions,” *IEEE Trans. on Automatic Control*, vol. 29, pp. 788–793, 1984.
- [108] F. Kraus, B. Anderson, E. Jury, and M. Mansour, “On robustness of low order Schur polynomials,” in *IEEE Trans. on Circuits and Systems*, pp. 570–577, 1988.
- [109] F. Kraus, B. Anderson, and M. Mansour, “Robust Schur polynomial stability and Khartitonov’s theorem,” *Int. Journal of Control*, vol. 47, pp. 1213–1275, 1988.
- [110] G. Kreisselmeier and R. Steinhauser, “Systematic control design by optimizing a vector performance index,” in *IFAC Symposium on Computer Aided Design of Control Systems*, (Zürich), pp. 113–117, 1979.
- [111] G. Kreisselmeier and R. Steinhauser, “Systematische Auslegung von Reglern durch Optimierung eines vektoriellen Gütekriteriums,” *Regelungstechnik*, vol. 27, pp. 76–79, 1979.
- [112] G. Kreisselmeier and R. Steinhauser, “Application of vector performance optimization to a robust control loop design for a fighter aircraft,” *Int. Journal of Control*, vol. 37, pp. 251–284, 1983.
- [113] H. Kwakernaak, “A condition for robust stabilizability,” *Systems and Control Letters*, vol. 3, pp. 1–5, 1982.
- [114] D. Lazard, “System of algebraic equations (algorithms and complexity),” in *Symposium on symbolic and algebraic computation*, (Bonn), 1991.

- [115] G. Leitmann, "Guaranteed asymptotic stability for some linear systems with bounded uncertainties," *J. of Dynamic Systems, Measurement, and Control*, vol. 1, pp. 212–216, 1979.
- [116] A. Leonhard, "Neues Verfahren zur Stabilitätsuntersuchung," *Archiv für Elektrotechnik*, vol. 38, pp. 17–28, 1944.
- [117] A. Liénard and A. Chipart, "On the sign of the real part of the roots of an algebraic equation," *J. Math. Pures et Appl.*, vol. 10, pp. 291–346, 1914.
- [118] W. Linvill, "Sampled-data control systems studied through comparison of sampling with amplitude modulation," *AIEE Transactions*, vol. 70, pp. 1779–1788, 1951.
- [119] L. MacColl, *Fundamental theory of servomechanisms*, pp. 88–101. New York: Van Nostrand, 1945.
- [120] A. MacFarlane, G. Grübel, and J. Ackermann, "Future design environments for control engineering," *Automatica*, vol. 25, No. 2, pp. 165–176, 1989.
- [121] A. MacFarlane and N. Karcianas, "Poles and zeros of linear multivariable systems: a survey of the algebraic, geometric and complex-variable theory," *Int. Journal of Control*, vol. 24, pp. 33–74, 1976.
- [122] M. Mansour, "On Robust Stability of Linear Systems," in *Systems and Control Letters*, (North Holland), 1993.
- [123] M. Mansour, F. Kraus, and B. Anderson, "Strong Kharitonov theorem for discrete systems," in *Proc. IEEE Conf. Decision and Control*, (Austin, Texas), 1988.
- [124] M. Marden, *The geometry of the zeros of a polynomial in a complex variable*. Math Surveys, no. 3, Providence, RI: American Math. Society, 1949.
- [125] A. Markazie and N. Hori, "A new method with guaranteed stability for discretization of continuous-time control systems," in *Proc. American Control Conference*, (Chicago), pp. 1397–1402, 1992.
- [126] D. McRuer, I. Ashkenas, and D. Graham, *Aircraft dynamics and automatic control*. Princeton: Princeton University Press, 1973.
- [127] L. Meirovitch, *Computational methods in structural dynamics*. Alphen van den Rijn, The Netherlands: Sijthoff & Noordhoff, 1980.
- [128] R. Middleton and G. Goodwin, *Digital control and estimation*. Englewood Cliffs, N.J.: Prentice-Hall, 1990.
- [129] A. Mikhailov, "Method of harmonic analysis in control theory," *Avtomatika i Telemekhanika*, vol. 3, pp. 27–81, 1938.
- [130] M. Mitschke, *Dynamik der Kraftfahrzeuge*, vol. C. Berlin: Springer, 1990.

- [131] R. Münch, “Reglerauslegung und Robustheitsanalyse für einen spurgeführten Bus,” Tech. Rep. IB 515-86-14, DLR, Oberpfaffenhofen, Germany, 1986.
- [132] T. Murdock and W. Schmittendorf, “Use of a genetic algorithm to analyze robust stability problems,” in *preprint*, 1991.
- [133] K. Narendra and A. Annaswamy, *Stable Adaptive Systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1989.
- [134] Y. Neimark, “On the root distribution of polynomials,” *Dokl. Akad. Nauk.*, vol. 58, pp. 357–360, 1947.
- [135] H. Nyquist, “Regeneration theory,” *Bell Systems Technical Journal*, pp. 126–?, 1932.
- [136] L. Orlando, “Sul problema di Hurwitz relativo alle parti reali delle radici di un’equazione algebraica,” *Mathematische Annalen*, vol. 71, pp. 233–245, 1911.
- [137] E. Panier, M. Fan, and A. Tits, “On the robust stability of polynomials with no cross-coupling between the perturbations in the coefficients of even and odd powers,” *Systems and Control Letters*, vol. 12, pp. 291–299, 1989.
- [138] V. Pareto, *Cours d’Economie Politique*. Lausanne: Rouge, 1896.
- [139] K. Poolla, J. Shamma, and K. Wise, “Linear and nonlinear controllers for robust stabilization problems: A survey,” in *Proc. IFAC Congress*, (Tallinn), pp. 176–183, 1990.
- [140] V. Popov, “Absolute stability of nonlinear systems of automatic control,” *Autom. & Rem. Control*, vol. 22, pp. 857–875, 1962.
- [141] F. Preparata and M. Shamos, *Computational geometry*. New York: Springer, 1985.
- [142] P. Putz and M. Wozny, “A new computer graphics approach to parameter space design of control systems,” *IEEE Trans. on Automatic Control*, vol. 32, pp. 294–302, 1987.
- [143] A. Rantzer, “Kharitonov’s weak theorem holds if and only if the stability region and its reciprocal are convex,” *Int. Journal of Nonlinear and Robust Control*, 1992.
- [144] A. Rantzer, “Stability conditions for polytopes of polynomials,” *IEEE Trans. on Automatic Control*, vol. 37, pp. 79–89, 1992.
- [145] P. Riekert and T. Schunck, “Zur Fahrmechanik des gummitbereiften Kraftfahrzeugs,” *Ingenieur Archiv*, vol. 11, pp. 210–224, 1940.
- [146] E. Routh, *A treatise on the stability of a given state of motion*. London: Macmillan, 1877.

- [147] R. Saeks and J. Murray, "Fractional representation, algebraic geometry, and the simultaneous stabilization problem," *IEEE Trans. on Automatic Control*, vol. 27, pp. 895–903, 1982.
- [148] I. Schur, "Über Potenzreihen, die im Inneren des Einheitskreises beschränkt sind," *Journal für Mathematik*, vol. 147, pp. 205–232, 1917.
- [149] I. Schur, "Über Potenzreihen, die im Inneren des Einheitskreises beschränkt sind," *Journal für Mathematik*, vol. 148, pp. 122–145, 1918.
- [150] K. Senger, "Dynamik und Regelung allradgelenkter Fahrzeuge," in *Fortschrittsberichte VDI*, no. 126 in Reihe 12: Verkehrstechnik/Fahrzeugtechnik, Düsseldorf: VDI-Verlag, 1989.
- [151] A. Sideris and R. deGaston, "Multivariable stability margin calculation with uncertain correlated parameters," in *Proc. IEEE Conf. Decision and Control*, (Athens), pp. 766–771, 1986.
- [152] A. Sideris and R. S. Peña, "Fast computation of the multivariable stability margin for real interrelated uncertain parameters," *IEEE Trans. on Automatic Control*, vol. 34, pp. 1272–1276, 1989.
- [153] W. Sienel, "Algorithms for tree structured decomposition," in *Proc. IEEE Conf. Decision and Control*, (Tuscon), pp. 739–740, 1992.
- [154] D. Siljak, *Nonlinear systems: the parameter analysis and design*. New York: Wiley, 1969.
- [155] C. Soh, "Robust stability of discrete-time systems using delta operators," in *IEEE Trans. on Automatic Control*, vol. AC-36, pp. 377–380, 1991.
- [156] C. Soh, C. Berger, and K. Dabke, "On the stability properties of polynomials with perturbed coefficients," *IEEE Trans. on Automatic Control*, vol. 30, pp. 1033–1036, 1985.
- [157] C. Soh and Y. Foo, "On the existence of strong Kharitonov theorems for interval polynomials," in *Proc. IEEE Conf. Decision and Control*, vol. 3, pp. 1882–1887, 1989.
- [158] K. Sondergeld, "A generalization of the Routh-Hurwitz stability criteria and an application to a problem in robust controller design," *IEEE Trans. on Automatic Control*, vol. 28, pp. 965–970, 1983.
- [159] G. Stein and J. Doyle, "Beyond singular values and loop shapes," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 14, pp. 5–16, 1991.
- [160] R. Steinhäuser, "Design of a feedback controller for a cryogenic windtunnel," in *Proc. 10th IFAC World Congr. on Automatic and Control*, vol. 3, (Munich), pp. 39–44, 1987.

- [161] B. Stevens and F. Lewis, *Aircraft control and simulation*. New York: Wiley, 1992.
- [162] M. Stieber, *Das Konzept des Tandemreglers zur automatischen Spurführung nicht schienengebundener Fahrzeuge*. MAN, Internal Report B 094002-EDS-005, 1980.
- [163] R. Tempo, “A dual result to Khartitonov’s theorem,” in *IEEE Trans. on Automatic Control*, vol. AC-35, pp. 195–197, 1990.
- [164] J. Truxal, “Adaptive control systems,” in *Control systems - some unusual design problems* (E. Mishkin and L. Braun, eds.), ch. 4, New York: McGraw, 1961.
- [165] T. Tsijino, T. Fujii, and K. Wei, “On the connection between controllability and stabilizability of linear systems with structural uncertain parameters,” *Automatica*, vol. 29, pp. 7–12, 1993.
- [166] Y. Tsypkin and B. Polyak, “Frequency domain criteria for  $l^p$ -robust stability of continuous linear systems,” *IEEE Trans. on Automatic Control*, vol. 36, no. 12, pp. 1464–1469, 1991.
- [167] Y. Tsypkin and B. Polyak, “Frequency domain criteria for robust stability of polytope of polynomials,” in *Control of uncertain dynamic systems* (S. Bhattacharyya and L. Keel, eds.), pp. 491–499, CRC Press, 1991.
- [168] Y. Tsypkin and B. Polyak, “Robust absolute stability of continuous systems,” in *Robustness of dynamic systems with parameter uncertainties* (M. M. et al., ed.), Basel: Birkhäuser, 1992.
- [169] B. van der Waerden, *Algebra*. Berlin: Springer, 1966.
- [170] P. Varaiya, “Smart cars on smart roads: design and evaluation,” *IEEE Trans. on Automatic Control*, vol. 38, pp. 195–207, 1993.
- [171] A. Vicino, A. Tesi, and M. Milanese, “An algorithm for nonconservative stability bounds computation for systems with nonlinearly correlated parametric uncertainties,” *IEEE Trans. on Automatic Control*, vol. 35, pp. 835–841, 1990.
- [172] M. Vidyasagar, *Nonlinear systems analysis*. Englewood Cliffs, N.J.: Prentice-Hall, 1978.
- [173] M. Vidyasagar, H. Schneider, and B. Francis, “Algebraic and topological aspects of feedback stabilization,” *IEEE Trans. on Automatic Control*, vol. 27, pp. 880–894, 1982.
- [174] M. Vidyasagar and N. Viswanadham, “Algebraic design techniques for reliable stabilization,” *IEEE Trans. on Automatic Control*, vol. 27, pp. 1085–1095, 1982.
- [175] I. Vishnegradsky, “Sur la théorie générale des régulateurs,” *Compt. Rend. Acad. Sci.*, vol. 83, pp. 318–321, 1876.

- [176] E. Walach and E. Zeheb, "Generalized zero sets of multiparameter polynomials and feedback stabilization," *IEEE Trans. on Circuits and Systems*, vol. 29, pp. 15–23, 1982.
- [177] K. Wei, "Asymptotic stabilization of single-input linear systems having unknown but bounded parameters," in *Proc. American Control Conference*, (Chicago), pp. 1004–1008, 1992.
- [178] K. Wei and B. Barmish, "An iterative procedure for simultaneous stabilization of mimo systems," *Automatica*, vol. 24, pp. 643–652, 1988.
- [179] D. Wu, W. Gao, and M. Chen, "Algorithm for simultaneous stabilization of single-input systems via dynamic feedback," *Int. Journal of Control*, vol. 51, pp. 631–642, 1990.
- [180] D. Youla, J. Bongiorno, and C. Lu, "Single-loop feedback-stabilization of linear multivariable dynamical plants," *Automatica*, vol. 10, pp. 159–173, 1974.
- [181] L. Zadeh and C. Desoer, *Linear system theory: the state space approach*. New York: MacGraw-Hill, 1963.
- [182] M. Zedek, "Continuity and location of zeros of linear combinations of polynomials," *Proceedings of the American Math. Society*, vol. 16, pp. 78–84, 1965.
- [183] A. Zomotor, *Fahrwerktechnik: Fahrverhalten*. Würzburg: Vogel-Verlag, 1987.

# Index

- accelerometer, 40, 332
- Ackermann's formula, 74, 299
- actuator, 261
- aircraft, 40
- Aizerman Conjecture, 229
- algebraic method, 90
- analysis and design, 42, 57
- Anderson, 129, 276
- anti-aliasing filter, 258
- approximate differentiation, 38
- automatic car steering, 15, 36, 318
- bandwidth, 327
- Barmish, 112, 276
- Bartlett, 136, 276
- barycentric coordinates, 133, 287
- basic rules of robust control, 76
- Bhattacharyya, 207, 229
- Bialas, 95, 120, 139, 211
- bilinear transformation, 266, 286
- Bode, 99
- boundary
  - approximate, 283
  - circle, 261
  - complex root, 239, 245, 282
  - grid, 174
  - hyperbola, 261
  - interior, 160
  - pixel approximation, 164
  - polygon approximation, 162
  - real root, 235, 282
  - validity, 80
- boundary crossing, 87, 93, 187, 238, 264
- boundary crossing theorem, 99, 117, 265, 295
- boundary mapping theorem, 240
- boundary representation, 234
- boundary representation theorem, 272
- box theorem, 207
- Butterworth, 64, 325
- canonical form, 8, 48
- car steering, 9, 44, 356
  - automatic, 283
- cascade structures, 41
- center of gravity, 45
- centrum of eigenvalues, 58
- Chapellat, 207, 229
- characteristic polynomial, 29, 42
- Chipart, 98
- coefficients
  - affine, 54, 135, 179, 182, 186
  - exponential, 257, 259
  - independent, 69
  - interval, 54, 179, 204, 209
  - linear, 256
  - multilinear, 54, 140
  - polynomial, 55, 151, 192, 244
- Cohn, 265, 275
- compensator, 205
  - low order, 35
  - positive, 211
- complex set
  - addition, 157
  - boundary, 219
  - inversion, 219
  - multiplication, 158
- compromise solution, 348
- computer algebra, 170
- computer graphics, 73, 164
- conservative estimate, 68, 69, 72
- continuous path, 89
- contour algorithm, 163
- controllability, 258

- robust, 26
- controllability matrix, 27
- controller
  - candidate, 73
  - deadbeat, 259
  - discrete-time, 258
  - integral, 324
  - integrating, 209
  - stabilizing simultaneously, 73, 293
- controller parameter space, 304
- convex hull, 147
  - of stable region, 267
  - stability region, 287
- cornering stiffness, 11
- crane, 4, 31, 83, 87, 92, 109, 115, 218, 296, 300
- Cremer, 112
- critical speed, 12
- critical stability conditions, 267
- crossover frequency, 103
- Curtis, 112
- curvature, 381
- cusp, 381
- D-decomposition, 106
- Dahleh, 229
- damping, 52, 59, 235, 259, 261, 330
- deadbeat solution, 263
- decentralized controller structure, 43
- degree, 381
- degrees of freedom, 40
- $\delta$ -transform, 277
- design, 294, 310
  - by search, 310
  - simultaneous Gamma-stabilization, 293
  - trial and error, 294
- design procedure, 354
- design specifications, 342
- design strategies, 73
- Desoer, 112, 147, 293
- differential equations, 5
- digital controller, 255
- dilation factor, 178
- discretization, 68
  - of controller and plant, 255
- discriminant, 239, 380
- Djaferis, 293
- domain splitting algorithm, 150
- dominant behavior, 59
- double point, 381
- Doyle, 293
- Duncan, 88, 93, 106, 107, 238
- edge, 80, 136, 138
- edge test, 278
- edge theorem, 135, 206, 278
- eigenvalues, 29
- elasticity, 41
- engineering art, 294
- engineering science, 294
- extremal value results, 68
- extreme point results, 276, 277
- failure detection, 316
- Fam, 268
- family
  - of Kharitonov plants, 205, 213, 230
  - of transfer functions, 205
- feedback
  - minimum-norm, 298
  - single loop, 203
- flight conditions, 20
- flight control, 19, 310, 330
- four-wheel car steering, 9, 44, 371
- Frazer, 88, 93, 106, 107, 238
- frequency
  - crossover, 103
  - domain methods, 70
  - generalized, 234
  - isolated, 101, 102, 104, 108, 183, 185, 190, 296
  - natural, 330
  - response, 70
  - singular, 99, 102, 103
  - transition, 137
  - upper bound, 115
- frequency band, 150
- frequency plot, 112, 120
- gain and phase margin, 70
- gain reduction margins, 311
- gain scheduling, 49

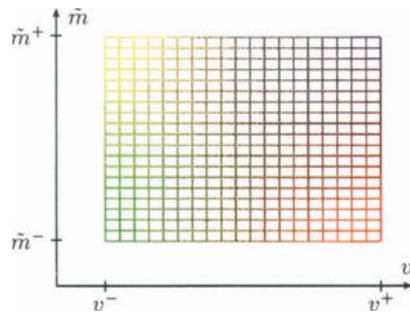
- Gamma-boundary  
circle, 236  
hyperbola, 235, 244, 296, 320  
logarithmic spiral, 242  
real axis segment, 235
- Gamma-stability, 234
- Ghosh, 293
- graphical solution, 308
- grid, 85, 87, 101, 104, 105, 118, 120, 156, 173, 264
- Gröbner bases, 383
- gyro, 37, 40, 332
- hand computations, 90
- Hermite, 90
- Hermite-Biehler theorem, 112
- hidden affine case, 140
- hold element, 255, 283
- Hollot, 136, 276
- Horowitz, 294
- Hurwitz, 33, 34, 69, 90
- hyperplane, 104, 187, 268
- hyperrectangle, 79
- hypersphere, 186
- index set, 166
- inflection point, 381
- integral term, 41
- interior point, 141, 143, 157, 174
- interlacing property, 115
- interval matrices, 68
- interval plant, 204, 205, 294  
positive, 211
- interval transfer function, 205
- isolated point, 141, 381
- Jacobi condition, 142
- Jacobi line, 144
- Jacobian, 101, 105, 107, 143, 163, 247
- Jury, 129, 264
- Kalman filter, 35
- Khargonekar, 293
- Kharitonov, 124, 126, 204
- Kharitonov polynomial, 206
- Kharitonov region, 276  
strong, 250
- weak, 250
- Kharitonov segment, 208
- Kharitonov's theorem, 277
- Kimura, 293
- Kraus, 276
- Kwakernaak, 293
- lateral mode, 44
- Leitmann, 293
- Leonhard, 112
- Leverrier algorithm, 30, 31
- Lienard, 98
- manipulated variables, 9
- Mansour, 129, 276
- mapping theorem, 147, 159
- Mason's formula, 222
- mass distribution, 45
- Matlab, 72
- Meditch, 268
- Mikhailov, 99, 112, 179
- Mikhailov plot, 128
- minimum-feedback norm, 311
- model  
state space, 21  
transfer function, 21  
uncertain plant, 21
- modelling  
analytical, 3  
input-output, 3
- moment of inertia, 47
- mu-optimization, 293
- multi-model problem, 73, 256, 295, 310
- natural frequency, 51, 59, 259
- necessary conditions, 267, 269
- Neimark, 106, 107
- nonconvex, 71
- nonlinear programming, 356
- nonlinear state space model, 6
- nonlinear tire characteristics, 49
- nonlinearity  
in a sector, 226  
of the plant model, 310
- nonminimum phase, 61
- norm bounded perturbation, 68
- normal vector, 189

- Nyquist, 99, 103, 112
- Nyquist plot, 211
- Nyquist value set, 280
- observability, 40, 258
  - robust, 26
- observability matrix, 28
- observable, 8, 37
- observer, 35
- octahedron, 275
- operating point
  - representative, 73, 256, 294
  - stable, 80
- optimization, 73
- Orlando's formula, 94, 265
- output feedback, 29, 40, 42
- overbounding, 14, 68, 70, 132, 151, 277
- overshoot, 58
- parameter
  - controller, 269
  - independent, 13
  - interval, 22
  - real, 79
- parameter box, 22
- parameter space, 110, 112, 120, 186, 239, 269, 293
  - cross sections, 75
- Pareto, 73, 349
- parpolygon, 137
- partition, 167
- partition of vectors, 43
- Pavlidis, 264
- penetration lines, 308
- performance, 295, 305
- performance criteria, 342
- phase, 114, 283
- PI controller, 35
- pixel, 164
- plane
  - cross section, 298, 309
  - distance from, 187
  - invariance, 298, 300, 302, 308
  - orthogonal, 187
- plant family, 21
  - finite, 22, 341
- point-in-polygon algorithm, 175
- Poisson sum, 257, 281
- pole assignment, 74
- pole placement matrix, 299, 302
- pole region, 295
- pole region assignment, 64, 73, 296
- pole-zero cancellations, 59
- Polyak, 179
- polynomial
  - $k$ -decomposition, 167
  - degree, 80
  - family, 79, 119
  - Hurwitz, 267
  - interval, 124, 126
  - product decomposition, 167
  - Schur, 267
  - sum decomposition, 167
  - total decomposition, 167, 171
  - uncertain, 54, 79, 83
- polynomial family, 67
- polynomials
  - classification, 54
  - polytope, 138, 268
  - Poolla, 293
- Popov criterion, 226, 312
- Popov plot, 227
- Popov sector, 228
- prefilter, 9, 40, 49, 53
- Quantitative Feedback Theory, 294
- quantization error, 311
- radius of stability, 280
- Rantzer, 250, 276, 278
- relative degree, 41, 258
- responses
  - frequency, 62
  - step, 62
- resultant, 239, 245, 379
- resultant matrix, 94
- robust controller design, 71
- robust stability analysis, 256
- robustly controllable, 27
- robustly Gamma-stabilizing, 71
- robustness analysis, 67, 72
- root locus, 13, 58, 84, 103

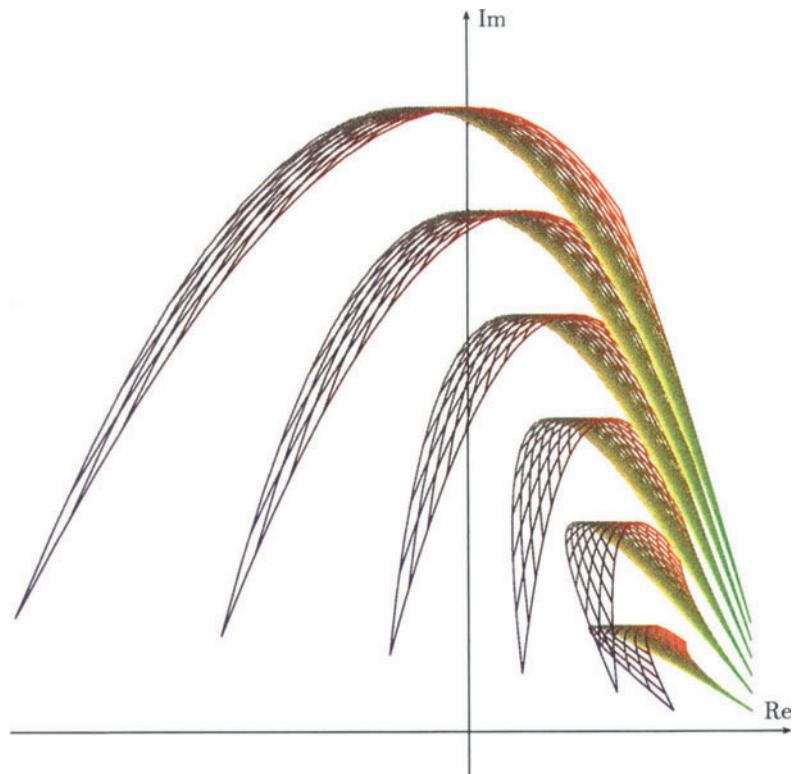
- root set, 80, 120  
Routh, 90  
sampling interval, 255, 258, 259, 261, 262, 284  
Schur, 264, 265  
sensor failures, 313  
sensor noise, 41  
sensors, 35  
    parallel, 315  
sets  
    intersection, 308  
Šiljak, 106  
similarity transformation, 42  
simultaneous Gamma-stabilization, 71, 295  
singular frequency, 101, 120  
singular perturbations, 80  
singular points, 381  
smallest destabilizing perturbation, 178  
soft control, 64  
specifications  
    eigenvalue, 57, 342  
    frequency domain, 345  
    time domain, 344  
stability, 81  
    absolute, 226  
    asymptotic, 81  
    boundaries, 188  
    circle, 284  
    critical condition, 99  
    Gamma, 34, 234, 261  
    margin, 311  
    necessary condition, 310  
    radius, 178, 248, 294  
    Schur, 263, 281  
    sufficient condition, 97, 147  
stability margin, 115  
stability region, 268  
stabilizability  
    robust, 27  
stabilization, 34, 40  
    simultaneous, 25, 26, 304, 305  
state feedback, 29, 295  
state space model, 3, 5  
steering transfer function, 10, 51  
step responses, 60  
strictly positive real, 211  
structural vibrations, 20, 70  
subsystem, 43  
subvector, 167  
sufficient condition, 275  
symbolic calculations, 67  
symbolic manipulation, 29  
system  
    mechanical, 172, 220  
    MIMO, 42, 214  
    minimum-phase, 211, 317  
    sampled-data, 255  
    scalable, 258  
testing sets, 276  
time responses, 59  
track guided bus, 197, 244, 247  
transfer function  
    tree structured, 216  
transfer function matrix, 3  
tree structured decomposition  
    of polynomials, 166  
    of transfer functions, 216  
triangularization, 44  
Tsyplkin, 179  
tuning parameter, 51  
uncertain parameters, 85  
uncertain polynomials, 26  
uncertainty  
    complex, 68  
    mathematically motivated, 70  
    norm bounded, 70  
    physically motivated, 70  
    structured, 70  
    unstructured, 70  
undershoot, 61  
unit ball, 178  
unit box, 178  
unit circle, 270  
unmodelled dynamics, 70  
upper bound, 115  
validity boundary, 107  
value set, 112, 118, 136, 142, 157, 218, 245, 280

- animation, 175
- color coding, 164, 247
- construction, 155
- Nyquist, 218, 221
  - sequential operations, 155
- vector performance index, 340
- velocity-invariant, 51
- velocity-scheduled gain, 9
- vertex, 80, 136, 138, 294, 304, 305
- Vidyasagar, 293
- Vishnegradsky, 106
- Wei, 293
- worst case perturbation, 72
- worst eigenvalues, 298
- yaw eigenvalues, 51
- yaw mode, 44
- Youla, 293
- z*-transfer function, 257
- Zadeh, 112
- zero exclusion, 119, 207
- zero sideslip angle, 53

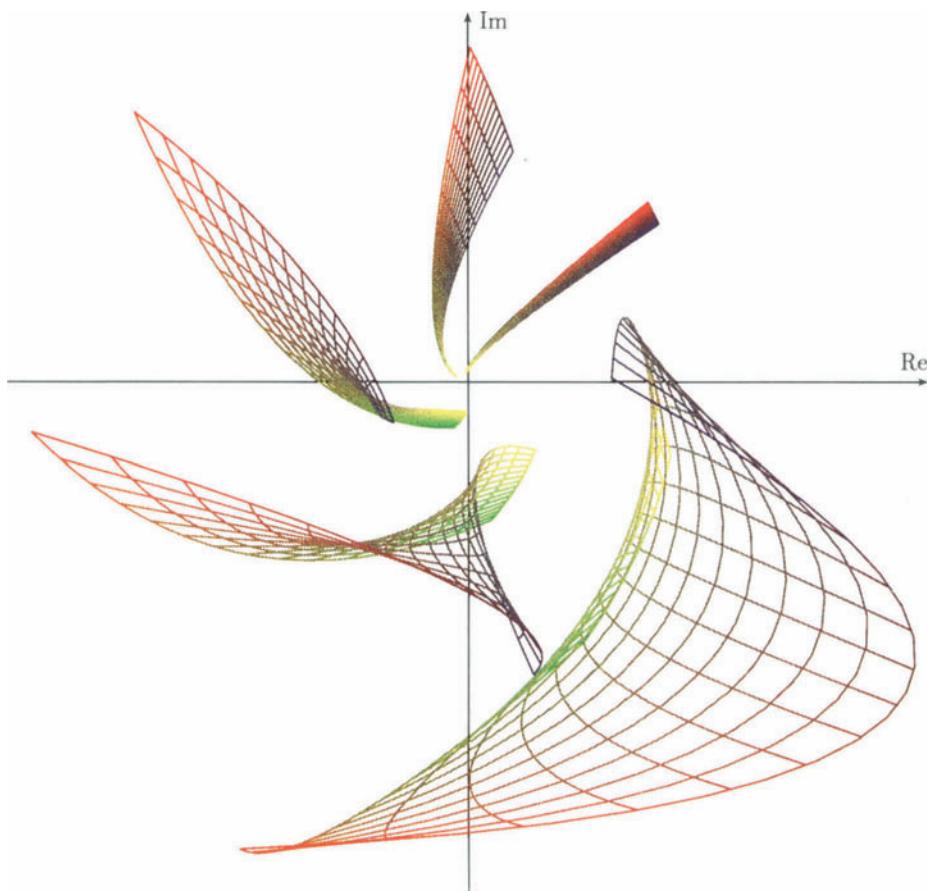
## Color plates



Color plate 1: Color coded uncertainty domain of the bus



Color plate 2: Color coded value sets of the yaw dynamics of the bus for  $s = -0.8 + i \cdot j0.2$ ,  $i = 1, \dots, 6$



Color plate 3: Color coded value sets of the track guided bus for  $\alpha = -0.36, -0.4, -0.5, -0.6$ , and  $\alpha = -0.7$