Cooperativity and its Use in Robust Control and State Estimation for Uncertain Dynamic Systems with Engineering Applications

Dissertation

zur
Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Maschinenbau und Schiffstechnik
an der Universität Rostock

vorgelegt von
M. Sc. Julia Kersten
geb. am 03.05.1989 in Görlitz

Rostock, 2. September 2020

Gutachter: Prof. Dr.-Ing. Harald Aschemann
Lehrstuhl für Mechatronik, Universität Rostock

Prof. Tarek Raïssi
Conservatoire National des Arts et Métiers

Tag der Verteidigung: 3. August 2020

Lehrstuhl für Mechatronik der Universität Rostock
2020

https://doi.org/10.18453/rosdok_id00002788

Preface

This Dissertation Thesis was done as part of my scientific work as a Research Associate with the Chair of Mechatronics at the Faculty of Mechanical Engineering and Marine Technology at the University of Rostock, Germany.

Firstly, I want to thank Prof. Dr.-Ing. Harald Aschemann, Head of the Chair of Mechatronics, for giving me this opportunity as well as for his supervision, scientific comments and the overall experience I gained while with his Chair. I further want to give my special thanks to Prof. Tarek Raïssi (CNAM, Paris, France) for serving as another reviewer of this thesis and his insightful comments on the subject. Additionally, I like to thank all commission members for their contribution and especially for their willingness to conduct the procedure with such promptness. This holds especially for both reviewers. I thank Prof. Dr.-Ing. Harald Aschemann for his straightforward organization despite the challenging pandemic-related inconveniences.

I would like to further express my gratitude to all — currently and former — colleagues at the Chair of Mechatronics and the cooperating Chairs for their help, be it administrative or scientific nature, as well as the private interactions invoking a lot of fun. This holds especially for Dr.-Ing. Luise Senkel, M.Sc. Wiebke Frenkel, M.Sc. Carolin Wüstenhagen, M.Sc. Alexander Wache and M.Sc. Lukas Pröhl as well as Angela Frankenberg and Kathrin Hollatz. An additional special thanks goes to PD. Dr.-Ing. habil Andreas Rauh for numerous insightful discussions on a broad variety of scientific topics but especially regarding this thesis. Moreover, I would like to thank the proofreaders, PD. Dr.-Ing. habil Andreas Rauh and Dr.-Ing. Luise Senkel.

Moreover, I would like to thank my family and friends for their moral support. A special thanks goes to Julia Furche for her help in handling the double load of family and work in the pandemic-related lock down. Without her, this thesis would not have been finished in time.

The last thanks are dedicated to my partner Frank for his encouragement and patience as well as to my stepson Thore for reminding me that sometimes a lost Lego is the only uncertainty that needs to be researched.

Rostock, August 2020 Julia Kersten

Cooperativity and its Use in Robust Control and State Estimation for Uncertain Dynamic Systems with Engineering Applications

— Abstract —

In control engineering, model-based designs are applied to a broad variety of applications. Here, one aims to find a balance in the modeling approach depicting the reality as detailed as necessary while keeping the complexity as low as possible in terms of realizability. For this, simplifications are used, e.g. in the form of approximation of nonlinearities, parameter couplings or order reduction. Additionally, errors may occur in parameter identification due to physically motivated phenomena, measurement effects, or possible numerical discretization errors. As a result, basically all real-life systems are subject to uncertainties.

In the presented work, those uncertainties are regarded as intervals, where worst-case bounds are represented by the upper and lower limit of an uncertain parameter. Novel control designs are introduced, which are based on a linear matrix inequality approach suitable for uncertain systems. Two extensions to state-of-the-art designs were given; the first with a constant controller gain approach over the complete time horizon and a second using a gain scheduling design over temporal subslices. Here, both rely on iterative solutions in the terms that controller gains are adapted based on the reachability analysis of former simulations. This means, that an efficient application of such methods is only realized with a reliable computation of possible interval enclosures. State-of-the-art enclosure techniques are often subject to overestimation, a possible solution comes in form of so-called cooperative systems. The structure of these systems allows for a separately, point-valued evaluation of the worst-case bounds, while guaranteeing the real value to be insight said bounds. This property can be found in numerous systems, however, exceptions occur especially concerning models from the fields of electrical as well as mechanical applications. To widen the applicability of cooperativity into these fields, this work presents transformation methods to adapt the structure of the treated system in such a way that it becomes cooperative while keeping its original stability properties. Due to the nature of said transformations this is done for systems with purely real eigenvalues and systems including conjugate-complex ones. As a final theoretical contribution, a state estimation is added to the controlled system as a form of fault diagnosis. Here, two possible approaches are presented. The first aims at keeping the structure of the controlled (and transformed into a cooperative form) system and, hence, is called cooperative-preserving observer. A second design is oriented on the control design making use of the duality principle, meaning that the controlled system is transformed and a parallel model, including the observer is also transformed into a cooperative system. Both results are then compared for the fault diagnosis to detect actuator or/and sensor faults.

Overall, this work gives a generally applicable method combining robust control designs with the computation of verified interval enclosures, and estimators as fault diagnosis tools. Based on the theoretical findings, suitable application scenarios are given in the second part of this work. Here, a constant gain controller design is applied to an electrical circuit, which is then subject to a transformation approach with purely real eigenvalues and a cooperativity-preserving observer design. Next, two mechanical, oscillatory systems are used to show a transformation based on complex-conjugate eigenvalues. Fault diagnosis models are further implemented in parallel. As an extension, the theory is applied to a fractional-order system to show that it works equally well for such models and highlight necessary adaptions. Finally, limits of the presented methods are acknowledged and an alternative solution is demonstrated on the example of an inverted pendulum.

Contents

List of Figures

List of Tables

Abbreviations

CPE	Constant phase element
FOM/FOS	Fractional-order model/system
IVP	Initial value problem
LMI	Linear matrix inequalities
OCV	Open circuit voltage
ODE	Ordinary differential equation
SOC	State of charge
w.l.o.g.	without loss of generality
w.r.t.	with respect to

Definitions

List of Symbols

An overview on used notations in this work shall be given in the following. In general, small letters express scalars (e.g. $p \in \mathbb{R}$) or, if written in boldface, vectors (e.g. $\mathbf{x}$). Boldfaced, capital letters (e.g. $\mathbf{A}$) are used for matrices. Intervals are denoted by square brackets, where $[\mathbf{x}] = \left[[x_1] \quad \dots \quad [x_n]\right]^T$ is an interval vector with each entry including $[x_i] = [\underline{x}_i \,;\, \overline{x}_i]$, $\underline{x}_i \leq x_i \leq \overline{x}_i$, $i = 1, \dots, n$. Furthermore, the list is split into three parts according to the structure of the presented thesis; theory, applications, and extensions. Note that supplementary, intermediate, and auxiliary variables are not explicitly listed, but specified at the place of their first usage.

Theory

Fundamental Notation

$\mathbf{P} \succ 0$	Positive definiteness of a matrix $\mathbf{P}$
$\mathbf{P} \succeq 0$	Positive semi-definiteness of a matrix $\mathbf{P}$
$\mathbf{P} \prec 0$	Negative definiteness of a matrix $\mathbf{P}$
$\mathbf{P} \preceq 0$	Negative semi-definiteness of a matrix $\mathbf{P}$
$\mathcal{D}$	Polytopic (vertex-related) representation of an uncertain dynamic system
$\mathbb{R}$	Set of real numbers
$\mathbb{C}$	Set of complex numbers
$\Re\{.\} = \sigma$	Real part
$\Im\{.\} = \omega$	Imaginary part
$\mathrm{rad}\,\{[x]\}$	Radius of an interval $[x]$
$\mathrm{diam}\,\{[x]\}$	Diameter of an interval $[x]$
$\mathrm{mid}\,\{[x]\}$	Midpoint of an interval $[x]$
$\mathrm{vol}\,\{[x]\}$	Volume of an interval $[x]$
$\inf([x])$	Infimum of an interval $[x]$
$\sup([x])$	Supremum of an interval $[x]$
$\mathbf{A}$	System matrix of a continuous-time dynamic system
$\mathbf{A}\,(\mathbf{x}(t))$	System matrix in quasi-linear state-space representation
$\mathbf{A}_\mathrm{C}$	System matrix in closed-loop control of a quasi-linear state-space representation
$\mathbf{A}_\mathrm{O}$	System matrix of an observer in quasi-linear state-space representation
$\mathbf{B}$, $\mathbf{b}$	Input matrix (vector) of a continuous-time dynamic system (MIMO, SISO)
$\mathbf{B}\,(\mathbf{x}(t))$	Input matrix in quasi-linear state-space representation
$\boldsymbol{\mathcal{B}}$	Bounding box
$\mathbf{C}$, $\mathbf{c}^T$	Output matrix (vector) of a continuous-time or discrete-time dynamic system (MIMO, SISO)
$\mathbf{C}\,(\mathbf{x}(t))$	Output matrix in quasi-linear state-space representation

$\mathbf{C}_\mathrm{m}$, $\mathbf{c}_\mathrm{m}^T$	Measured output matrix (vector) of a continuous-time or discrete-time dynamic system (MIMO, SISO)
$\mathbf{D}$, d	Feed-through matrix (scalar) of a continuous-time dynamic system (MIMO, SISO)
$\mathbf{D}_0$, $\mathbf{D}_1$	Real-valued parameter matrices
$\mathbf{E}$	Matrix with all elements equal to 1
$\mathbf{e}(t)$	Error vector in observer design
$\dot{\mathbf{x}} = \mathbf{f}(\cdot)$	Function describing the dynamics of a continuous-time dynamic system
$\mathbf{G}_\mathbf{w}$	Reference transfer functions matrix
$\mathbf{H}$, $\mathbf{h}$	Feedback gain matrix (vector) of a continuous-time state observer (MIMO, SISO)
$\mathbf{I}$	Identity matrix
$\mathbf{J}$	Jacobian matrix
$\mathbf{K}$, $\mathbf{k}^T$	Feedback gain matrix (vector) of a continuous-time state controller (MIMO, SISO)
$\mathbf{N} = \tilde{\mathbf{A}}_\mathrm{C}$	Transformed system matrix (always Metzler)
$\mathbf{p}$	Vector of (uncertain) system parameters
s	Laplace variable
T	Sampling time/ Time constant
t_k	Discrete time steps
$\mathbf{T}$, $\mathbf{S}$	Transformation matrix (into Metzler structure)
$\tilde{\mathbf{T}}$, $\mathbf{V}$	Transformation matrix (into diagonal dominant form)
$\mathbf{u}$	Input vector of a continuous-time dynamic system
$V(.)$	Lyapunov function (candidate)
$\mathbf{v}_{\lambda i}$	i-th left eigenvector of a dynamic system
$\mathbf{w}_{\lambda i}$	i-th right eigenvector of a dynamic system
$\mathbf{v}$, $\mathbf{w}$	worst-case bounds of an uncertain state $\mathbf{v} < \mathbf{x} < \mathbf{w}$
$\mathbf{x}(t) \in \mathbb{R}^{n_\mathrm{x}}$	State vector of a continuous-time dynamic system
$\hat{\mathbf{x}}(t)$, $\hat{\mathbf{y}}(t)$	State/ output estimate determined by a continuous-time observer
$[\mathcal{X}]$	Domain of reachable states
$\mathbf{y}(t) \in \mathbb{R}^{n_\mathrm{y}}$	Output vector of a continuous-time dynamic system
$\mathbf{y}_\mathrm{m}(t) \in \mathbb{R}^{n_\mathrm{m}}$	Measured output vector of a continuous-time dynamic system (analogously the index m is used for all further occurrences in which measured and general system outputs have to be distinguished)
$\mathbf{z} = \tilde{\mathbf{x}}(t) \in \mathbb{R}^{n_\mathrm{x}}$	Transformed state vector of a continuous-time dynamic system
γ	Stability margin in control and observer design
λ_i	i-th eigenvalue of a dynamic system
$\mathbf{\Theta}$	Overall transformation matrix
σ_p	Sensitivity of the system to a parameter variation p
$\boldsymbol{\xi}$	Generalized parameter vector of a dynamic system

Application Scenarios

Electrical Circuit

C	Capacity
i_i	Currents with $i \in \{R_\mathrm{S}, R_\mathrm{L}, R_\mathrm{C}, C, L\}$
L	Inductivity
P_d	Desired power
R_i	Ohmic resistances with $i \in \{\mathrm{S}, \mathrm{L}, \mathrm{C}\}$
u_i	Voltages with $i \in \{R_\mathrm{S}, R_\mathrm{L}, R_\mathrm{C}, C, L\}$
u_0	Input voltage

High-Bay Rack Feeder

A	Cross sectional area of the beam
$\mathbf{D}$	Damping matrix
E	Young's modulus of the beam
$\mathbf{h}$	Input vector of generalized forces
I_{zB}	Second moment of area of the beam
k_d	damping factor
$\mathbf{K}$	Stiffness matrix
l	Length of the beam
m_i	Mass of the cage, carriage, and at the tip of the beam ($i \in \{K, S, E\}$)
$\mathbf{M}$	Mass matrix
$\mathbf{q}$	Vector of generalized coordinates
T_1	time constant of the underlying velocity control
v_S	Velocity of the carriage
x_K	Vertical position of the cage
y_i	Horizontal position of the cage, and carriage ($i \in \{K, S\}$)
$\boldsymbol{\delta}$	Disturbance vector
η	Confidence level
Θ_K	Mass moment of inertia of the beam
κ	Dimensionless system parameter
ρ	Density of the beam
ν	Bending deflection of the first eigenmode

Boom Crane

d	Damping coefficient normalized on m_L
D	Dissipation parameter
E_i	Kinetic and potential energy ($i \in \{\mathrm{kin}, \mathrm{pot}\}$)
F_i	Force on payload and carriage ($i \in \{L, HT\}$)
g	Gravitational acceleration
l	Rope length
m_i	Mass of crane, payload, and carriage ($i \in \{C, L, HT\}$)
$\mathbf{q}$	Vector of generalized coordinates
$\mathbf{Q}$	Vector of external forces[1]

[1] Note that this vector is written with a capital letter to distinguish it from the vector of generalized coordinates.

$\mathcal{Q}$, $\mathcal{R}$	Weighting matrices
r	Carriage postion
T_1	Time constant
x_L	Vertical position of the payload
y_L	Horizontal position of the payload
δ	Damping coefficient
ϕ	Rope angle

Extensions

Fractional-Order System

C	Capacity
i_0	Terminal current of the battery
Q_n	Nominal battery capacity
R	Ohmic resistance
u_0	Terminal voltage of the battery
u_{OCV}	Open circuit voltage
u_{CPE}	Voltage of the constant phase element
η	Charging/discharging efficiency
ν	Fractional order
σ	State of charge

Alternative Computation of Interval Enclosures–Inverted Pendulum on a Cart

a	Length of the pendulum
F	Accelerating force of the carriage controller
g	Gravitational acceleration
m	Point mass at the pendulum tip
M	Carriage mass
T	Final time of the exponential enclosure technique
T_1	Time constant of the underlying carriage velocity controller
u	Desired carriage velocity (system input)
x	Vertical position of the carriage
α	Pendulum angle
$\mathbf{\Lambda}$	Diagonal matrix of eigenvalues

1 Introduction

Most advanced control tasks, which rely on modeling the dynamic behavior of a system, are often described by ordinary differential equations (ODEs)

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}(t)) \ , \tag{1.1}$$

where $\mathbf{f}$ is a function describing the system dynamics depending on the state vector $\mathbf{x}$, the input $\mathbf{u}$, and $\mathbf{p}$ as a vector of parameters. Note that the input and the parameters can be either time-invariant or time-varying[1]. In order to generate a system model, one needs to understand the mathematical and physical properties of the dynamic system in question. However, to find a system model to suit specified control tasks, it has to be decided which of these properties define the problem at hand best and whether some can be approximated — either because information does not need to be considered for the required control goal or it is simply not possible to acquire — to balance model-accuracy and implementation effort. Since approximation is a common tool when working with model-based control design, basically all real-world systems are subject to uncertainties. Simplifications can be, e.g. approximation of nonlinearities, parameter couplings or order reduction. Furthermore, errors may occur in parameter identification due to physically motivated phenomena, measurement effects, or possible numerical discretization errors. All of these errors can be the source of uncertainty, which is divided into two main types, on the one hand as

- **aleatory**, which is classified by environmental stochasticity meaning that results are not repetitive. These uncertainties are quantified using probabilistic methods, such as Monte-Carlo [6] or polynomial chaos [9] as the realization of Wiener processes. On the other hand
- **epistemic** uncertainty is regarded as systematic due to lack of knowledge. Here, we assume that those can be seen as bounded uncertainties, which means that methods to work with are e.g. Taylor models or intervals [65].

In the presented thesis, the considered uncertainties either come from parameter variations or from an approximated overbounding of nonlinearities and are, hence, classified as epistemic, structured and, accordingly, bounded. In this case, the dynamic system can be treated by interval variables. Normally, simple shapes such as zonotopes [28, 32] or intervals [21] replace complex shaped regions in a multi-dimensional state-space to simplify the computation. In this thesis, we make use of intervals to express the worst-case bounds as shown in Fig. 1.1 for an example of a parameter vector $\mathbf{p} = [p_1 \quad p_2]^T$ of order $n = 2$.

Here, one can see that the parameter variations stretch out a whole box of possible combinations. To account for such a scenario including all parameter realizations, the control design needs to be robust. This describes the property of the system to tolerate such[2] variations without exceeding predetermined tolerance bounds in the vicinity of some nominal dynamic behavior [65]. However, if robustness increases, the nominal degree of performance decreases, which

[1] For simplicity, time arguments for both will be omitted in the notation while always including both cases until otherwise established.

[2] e.g. parameter

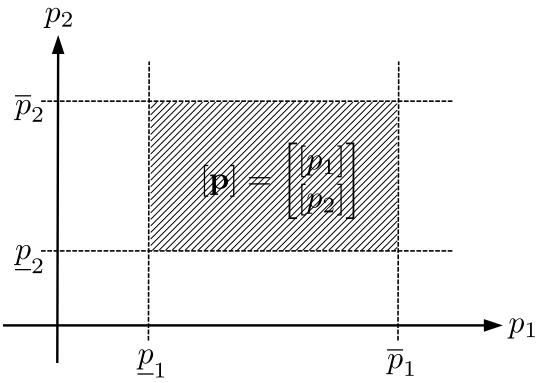


Figure 1.1: Interval box $\mathbf{p}$ resulting from two uncertain parameters p_1 and p_2.

means that another trade-off between robustness and performance has to be considered. Fig. 1.1 also shows, that if the parameters depend on one another arbitrary interactions are considered, increasing the conservativity of the robust controller further. To summarize, "the standard objective is to find a criterion for a controller as least conservative as possible" [65].
There are several possibilities to reduce conservativity and only a few will be subject of this thesis. However, a verified enclosure of all reachable states is often crucial not only to analyze the system's dynamics but also to find suitable control and observer designs. To achieve this, a lot of the presented approaches make use of the property of cooperativity.

1.1 Cooperativity of Continuous-Time Dynamic Systems

For most control purposes, if the dynamics are linear, Eq. (1.1) is reformulated into the so-called state-space model

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{p})\mathbf{x}(t) + \mathbf{B}(\mathbf{p})\mathbf{u}(t) \tag{1.2}$$

with the system matrix $\mathbf{A}$, the input matrix $\mathbf{B}$ and the output equation

$$\mathbf{y} = \mathbf{C}(\mathbf{p})\mathbf{x}(t) + \mathbf{D}(\mathbf{p})\mathbf{u}(t) \tag{1.3}$$

with the output matrix $\mathbf{C}$ and a feed-through matrix $\mathbf{D}$. Advantages of this representation include its efficiency when analyzing the system's structural properties — e.g. stability, controllability, and observability — and simplified implementation of control designs. For nonlinear systems, however, often a quasi-linear representation of the state-space model

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}(t)) \cdot \mathbf{x}(t) + \mathbf{B}(\mathbf{x}(t)) \cdot \mathbf{u}(t) \tag{1.4}$$

can be used, where the nonlinearities are shifted as variables into the describing matrices (system, input, output and feed-through) as done in this work. Note that both matrices $\mathbf{A}(\mathbf{x}(t))$ and $\mathbf{B}(\mathbf{x}(t))$ need to be without singularities and bounded for the whole operating horizon. This is realized by factoring out selected state variables. Note that for the transformation into cooperative representations, Eqs. (1.2) and (1.4) are treated equally, while the methods for control design differ in some aspects as described in Chapter 3.

The advantage of a cooperative system is that the worst-case bounds can be decoupled and then be treated as two known systems which simplifies the computation of worst-case enclosures of the system states with uncertain initial conditions to a large extent. Here,

$$\underline{\mathbf{A}} \cdot \mathbf{v}(t) + \underline{\mathbf{B}} \cdot \mathbf{u}(t) = \dot{\mathbf{v}}(t) \leq \dot{\mathbf{x}}(t) \leq \dot{\mathbf{w}}(t) = \overline{\mathbf{A}} \cdot \mathbf{w}(t) + \overline{\mathbf{B}} \cdot \mathbf{u}(t) \tag{1.5}$$

with

$$\begin{aligned} \underline{\mathbf{A}} &= \inf\left([\mathbf{A}(\mathbf{p})]\right) & \underline{\mathbf{B}} &= \inf\left([\mathbf{B}(\mathbf{p})]\right) \\ \overline{\mathbf{A}} &= \sup\left([\mathbf{A}(\mathbf{p})]\right) & \overline{\mathbf{B}} &= \sup\left([\mathbf{B}(\mathbf{p})]\right) \end{aligned} \tag{1.6}$$

holds if the terms $[\mathbf{B}] \cdot \mathbf{u}(t)$ as well as $\mathbf{v}(t)$ are always positive, otherwise,

$$\begin{aligned} \dot{\mathbf{v}}(t) &= \inf\left([\mathbf{A}_{\mathrm{d}}(\mathbf{p})] \cdot \mathbf{v}(t)\right) + \inf\left([\mathbf{A}_{\overline{\mathrm{d}}}(\mathbf{p})] \cdot [\mathbf{z}](t) + [\mathbf{B}(\mathbf{p})] \cdot \mathbf{u}(t)\right) \\ \dot{\mathbf{w}}(t) &= \sup\left([\mathbf{A}_{\mathrm{d}}(\mathbf{p})] \cdot \mathbf{w}(t)\right) + \sup\left([\mathbf{A}_{\overline{\mathrm{d}}}(\mathbf{p})] \cdot [\mathbf{z}](t) + [\mathbf{B}(\mathbf{p})] \cdot \mathbf{u}(t)\right) \end{aligned} \tag{1.7}$$

with $\mathbf{z}(t) \in [\mathbf{z}](t) = [\mathbf{v}(t)\,;\,\mathbf{w}(t)]$, $\mathbf{A}_{\mathrm{d}} = \mathrm{diag}(\mathbf{A}) = \mathbf{A} \circ \mathbf{I}$, and $\mathbf{A}_{\overline{\mathrm{d}}} = \mathbf{A} \circ (\mathbf{E} - \mathbf{I})$, must be applied. A sufficient criterion for cooperativity [61] of autonomous dynamic systems

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t)\right) \;\;,\;\; \mathbf{x} \in \mathbb{R}^n \tag{1.8}$$

is that all off-diagonal elements $J_{i,j}$, $i, j \in \{1, \ldots, n\}$, $i \neq j$, of the corresponding Jacobian

$$\mathbf{J} = \frac{\partial \mathbf{f}\left(\mathbf{x}\right)}{\partial \mathbf{x}} \tag{1.9}$$

are strictly non-negative according to

$$J_{i,j} \geq 0 \;\;,\;\; i, j \in \{1, \ldots, n\} \;\;,\;\; i \neq j \;\;. \tag{1.10}$$

This is commonly referred to as a Metzler structure [22, 33]. For such cases, it is guaranteed that state trajectories $\mathbf{x}(t)$ starting in the positive orthant

$$\mathbb{R}^n_+ = \{\mathbf{x} \in \mathbb{R}^n \;\mid\; x_i \geq 0, \quad \forall i \in \{1, \ldots, n\}\} \tag{1.11}$$

stay in this positive orthant for all $t \geq 0$ because $\dot{x}_i(t) = f_i\left(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots x_n\right) \geq 0$ holds for all components $i \in \{1, \ldots, n\}$ of the state vector as soon as the state x_i reaches the value $x_i = 0$. This property is commonly also denoted as positivity of the system model (1.8) [22, 33].

In case of a Metzler structure of the matrix $\mathbf{A}$, it is possible to simplify not only simulation tasks but also the application of suitably parameterized observers or state/output feedback controllers that can preserve cooperativity [15].

In general, dynamic systems can be described as cooperative, continuous-time models, if they are characterized by a finite number of non-negative state variables that can be interpreted as the nc[illegible] eneral systen

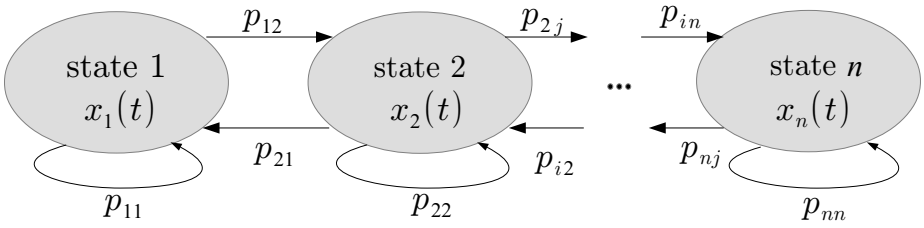


Figure 1.2: Graphical representation of an autonomous dynamic system.

Transport terms are, here, proportional to both, the node variable from which the edge starts and a non-negative (possibly state-dependent) weighting factor. Following this representation, the ODEs can be derived by

$$\dot{x}_i = -\sum_{j=1}^{n} p_{ij} x_i + \sum_{j=1, i \neq j}^{n} p_{ji} x_j \tag{1.12}$$

with $p_{ij} \in \mathbb{R}$, $p_{ij} \geq 0$ and $p_{ji} \geq 0$, $i \neq j$, resulting in

$$\dot{\mathbf{x}} = \begin{bmatrix} -\sum_{j=1}^{n} p_{1j} & p_{21} & \cdots & p_{n1} \\ p_{12} & -\sum_{j=1}^{n} p_{2j} & \cdots & p_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1n} & p_{2n} & \cdots & -\sum_{j=1}^{n} p_{nj} \end{bmatrix} \mathbf{x} \,, \tag{1.13}$$

where the Metzler structure [43] of the system matrix is obvious with non-negative off-diagonal entries. For a necessary condition of additional Hurwitz stability, see Sec. 2.5.1, the diagonal elements need to be strictly negative [17].
This structure naturally occurs in several applications in physics, biology, and (bio-)chemistry such as

- mass transfer in fluidic compartment models [12] and pressure dynamics in hydraulic and pneumatic networks with lumped storage elements,
- energy transfer between finite volume elements in thermal systems [47],
- compartment models of cell maturation, where the graph edges represent the growth of bacteria or microorganisms according to the Monod kinetics [19],
- enzymatic reactions according to the Michaelis-Menten kinetics [10], or
- continuous-time Markov chain models if the state variables denote probabilities for discrete model states with edge weights corresponding to the respective transition likelihoods.

However, ODEs derived by some first-principle modeling techniques like Kirchhoff's laws for general electric or magnetic networks with lumped parameters as well as modeling of mechanical multibody systems by the application of Lagrange's equations of second kind do not fulfill the sufficient properties of cooperativity given in (1.9) and (1.10). To apply cooperative treatment to these systems we, therefore, need to find general transformation procedures to use the advantages in a wider field of different applications.

1.2 Aim of this Work and its General Methodology

In this work, a generally applicable method shall be derived combining robust control designs with the computation of verified interval enclosures. Additionally, observers will be used as a form of fault diagnosis tool, further securing the overall system dynamics to follow the requirements set by the user by including this system diagnosis.

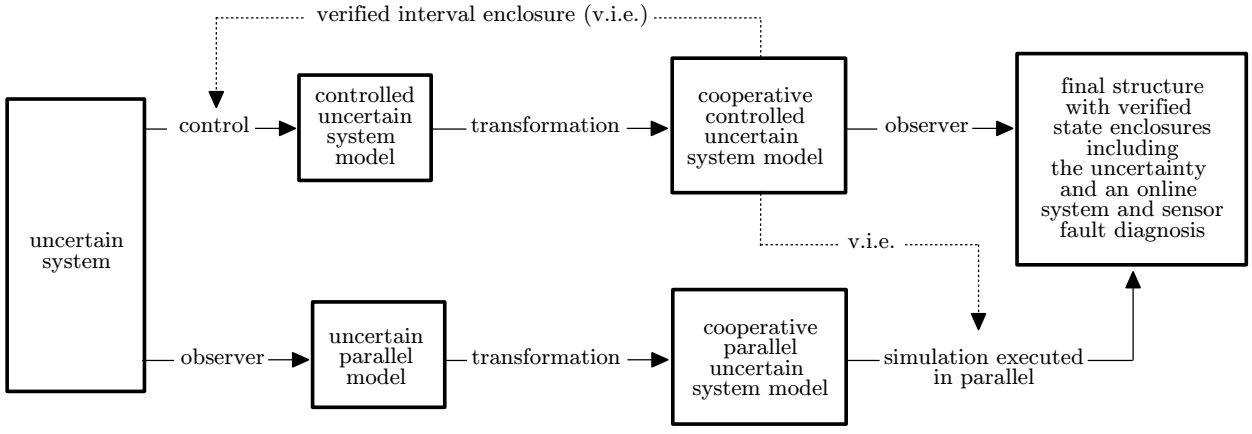


Figure 1.3: Outline of the theoretical aspects and contributions of this work.

For this, the overall theoretical aspects will be described in the first part, basically oriented on the structure of Fig. 1.3. Generally, a control design is firstly applied to the uncertain system. The resulting model is then transformed into a cooperative form with which verified interval enclosures (v.i.e.) can be computed. Those can either be used to tune the control design or as a comparison for the sensor diagnosis, when the observer gain is computed analogously to the controller gain following the duality principle. Here, a parallel model denotes a system representation simulated in parallel to the actual execution of the control task. Note that throughout this work, the reality will be called the "system" and its mathematically derived descriptions are named "model". The parallel model path of Fig. 1.3 can be omitted, if a cooperativity-ensuring observer design can be applied, so that the upper path will lead to the final structure.
After the theoretical explanations, suitable application scenarios to underline the general applicability and possible adaptations will be presented subsequently.

1.3 Outline

The presented thesis consists of three parts. The first part gives an overview on theoretical aspects used in this work. For this, Chapter 2 represents preliminaries as a summary of selected mathematical definitions regarding both main topics, interval arithmetics and control engineering. Subsequently, the next chapters follow the structure of Fig. 1.3 with a control synthesis in Chapter 3, the transformation procedures in Chapter 4 and observer designs in Chapter 5 concluding the theory part. The focus of Part II lies on giving application scenarios, where the theoretical aspects are put into a practical context. Following the findings of Chapter 4, the results are given separately for systems with purely real eigenvalues in Chapter 6 and systems including conjugate-complex eigenvalues in Chapter 7. Part III is dedicated to extensions of the presented theory. Here, Chapter 8 applies the deduced control design and transformation strategy to fractional-order systems and outlines necessary adaptations. In Chapter 9, limits of the presented transformation are presented and an alternative solution for computing interval enclosures is given. Chapter 10 concludes this thesis and gives an outlook on future work.

Part I

Theory

2 Preliminaries

Throughout this thesis, a number of different techniques from various fields including computer science, scientific computation, numerics, and control engineering will be applied. Therefore, this chapter contains an overview of mathematical basics and state-of-the-art findings as well as general clarifications of used technical terms. Thus, central topics of this chapter are fundamentals on interval arithmetic, covering basic operations, computation and arising problems, as well as state-of-the-art computations of interval enclosures namely verified forms of temporal Taylor series expansions. One can further find comments on sensitivity and basic information on linear matrix inequalities. Finally, stability of dynamic systems is discussed and different approaches to verify stability are given.

2.1 Fundamentals of Interval Arithmetic

Note that this section is limited to those aspects of interval arithmetic used in this thesis and hence, shall not be seen as a general overview. For more detailed information, the reader is referred to [21, 36].

A real-valued scalar interval is given by

$$[x] = [\underline{x}\ ;\ \overline{x}] = [\inf([x])\ ;\ \sup([x])]\,, \quad \underline{x} \leq \overline{x}, \quad \{\chi \in \mathbb{R} | \underline{x} \leq \chi \leq \overline{x}\}\ , \tag{2.1}$$

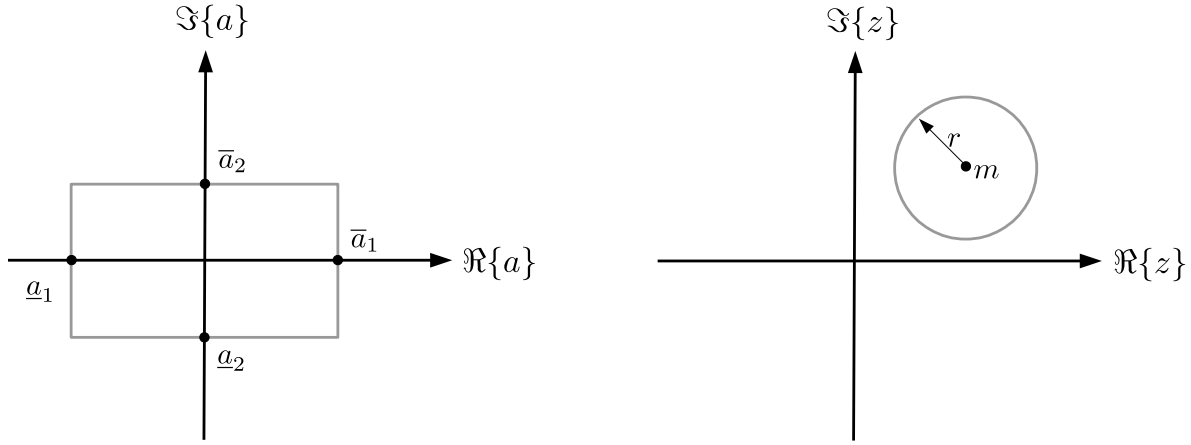
where the (largest) lower bound is defined as the infimum and the (smallest) upper bound as the supremum, respectively. The bounds are real finite values, that can be equal $\underline{x} = \overline{x}$, and hence, representing a point-valued interval. The interval vector

$$[\mathbf{x}] = \begin{bmatrix} [\underline{x}_1\ ;\ \overline{x}_1] \\ [\underline{x}_2\ ;\ \overline{x}_2] \\ \vdots \\ [\underline{x}_n\ ;\ \overline{x}_n] \end{bmatrix} \tag{2.2}$$

with n entries, assumed to be independent, is a special case of the interval matrix

$$[\mathbf{X}] = \begin{bmatrix} [\underline{x}_{11}\ ;\ \overline{x}_{11}] & [\underline{x}_{12}\ ;\ \overline{x}_{12}] & \dots & [\underline{x}_{1n}\ ;\ \overline{x}_{1n}] \\ [\underline{x}_{21}\ ;\ \overline{x}_{21}] & [\underline{x}_{22}\ ;\ \overline{x}_{22}] & \dots & [\underline{x}_{2n}\ ;\ \overline{x}_{2n}] \\ \vdots & \vdots & \ddots & \vdots \\ [\underline{x}_{m1}\ ;\ \overline{x}_{m1}] & [\underline{x}_{m2}\ ;\ \overline{x}_{m2}] & \dots & [\underline{x}_{mn}\ ;\ \overline{x}_{mn}] \end{bmatrix}\ . \tag{2.3}$$

Note that it is also possible to define complex-valued intervals which can be visualized and represented numerically as an interval box, see Fig. 2.1(a), or as a midpoint-radius form as in Fig. 2.1(b).



(a) Interval box: $[a] = [\underline{a}_1 \,;\, \overline{a}_1] + j\,[\underline{a}_2 \,;\, \overline{a}_2]$ **(b)** Midpoint-radius form $[z] = \langle m, r \rangle$

Figure 2.1: Visualization of complex-valued intervals.

2.1.1 Basic operations

Calculating with intervals is subject to a variety of considerations which must be obeyed in order to apply the techniques correctly. The basic operations addition

$$[x_1] + [x_2] = [\underline{x}_1 + \underline{x}_2 \,;\, \overline{x}_1 + \overline{x}_2] \tag{2.4}$$

and subtraction

$$[x_1] - [x_2] = [\underline{x}_1 - \overline{x}_2 \,;\, \overline{x}_1 - \underline{x}_2] \tag{2.5}$$

require no further restrictions if carried out on the set of real numbers. Multiplication

$$[x_1] \cdot [x_2] = [\min\{\underline{x}_1\ \underline{x}_2, \underline{x}_1\ \overline{x}_2, \overline{x}_1\ \underline{x}_2, \overline{x}_1\ \overline{x}_2\} \,;\, \max\{\underline{x}_1\ \underline{x}_2, \underline{x}_1\ \overline{x}_2, \overline{x}_1\ \underline{x}_2, \overline{x}_1\ \overline{x}_2\}] \tag{2.6}$$

and division

$$\frac{[x_1]}{[x_2]} = [x_1] \cdot \left[\frac{1}{\overline{x}_2} \,;\, \frac{1}{\underline{x}_2}\right] \quad \text{if} \quad 0 \notin [x_2] \tag{2.7}$$

however, are subject to the distinctions summarized in Tabs. 2.1 and 2.2.

Table 2.1: Case distinction for multiplication.

	$\overline{x}_2 \leq 0$	$\underline{x}_2 < 0 < \overline{x}_2$	$0 \leq \underline{x}_2$
$\overline{x}_1 \leq 0$	$[\overline{x}_1\ \overline{x}_2 \,;\, \underline{x}_1\ \underline{x}_2]$	$[\underline{x}_1\ \overline{x}_2 \,;\, \underline{x}_1\ \underline{x}_2]$	$[\underline{x}_1\ \overline{x}_2 \,;\, \overline{x}_1\ \underline{x}_2]$
$\underline{x}_1 < 0 < \overline{x}_1$	$[\overline{x}_1\ \underline{x}_2 \,;\, \underline{x}_1\ \underline{x}_2]$	$[\min\{\overline{x}_1\ \underline{x}_2, \underline{x}_1\ \overline{x}_2\} \,;\, \max\{\underline{x}_1\ \underline{x}_2, \overline{x}_1\ \overline{x}_2\}]$	$[\underline{x}_1\ \overline{x}_2 \,;\, \overline{x}_1\ \overline{x}_2]$
$0 \leq \underline{x}_1$	$[\overline{x}_1\ \underline{x}_2 \,;\, \underline{x}_1\ \overline{x}_2]$	$[\overline{x}_1\ \underline{x}_2 \,;\, \overline{x}_1\ \overline{x}_2]$	$[\underline{x}_1\ \underline{x}_2 \,;\, \overline{x}_1\ \overline{x}_2]$

Further important describing properties of intervals are given with the interval radius

$$\operatorname{rad}\{[x]\} = \frac{1}{2}\,(\overline{x} - \underline{x}) \quad , \tag{2.8}$$

the interval diameter

$$\operatorname{diam}\{[x]\} = \overline{x} - \underline{x} = 2 \cdot \operatorname{rad}\{[x]\} \quad , \tag{2.9}$$

Table 2.2: Case distinction for division.

	$\overline{x}_2 < 0$	$0 < \underline{x}_2$
$\overline{x}_1 \leq 0$	$[\overline{x}_1/\underline{x}_2 \ ; \ \underline{x}_1/\overline{x}_2]$	$[\underline{x}_1/\underline{x}_2 \ ; \ \overline{x}_1/\overline{x}_2]$
$\underline{x}_1 < 0 < \overline{x}_1$	$[\overline{x}_1/\overline{x}_2 \ ; \ \underline{x}_1/\overline{x}_2]$	$[\underline{x}_1/\underline{x}_2 \ ; \ \overline{x}_1/\underline{x}_2]$
$0 \leq \underline{x}_1$	$[\overline{x}_1/\overline{x}_2 \ ; \ \underline{x}_1/\underline{x}_2]$	$[\underline{x}_1/\overline{x}_2 \ ; \ \overline{x}_1/\underline{x}_2]$

as well as the interval midpoint

$$\operatorname{mid}\{[x]\} = \frac{1}{2}\left(\underline{x} + \overline{x}\right) \ , \tag{2.10}$$

and, finally, the volume of an interval box with n intervals

$$\operatorname{vol}\{[x]\} = \prod_{j=1}^{n} \operatorname{diam}\{[x_j]\} \ , \tag{2.11}$$

where all of the Eqs. (2.8)–(2.10) are defined component-wise.

2.1.2 Computing with intervals

Not only the evaluation of basic operations differs for intervals in comparison to point values. While some solutions may be obvious for the user, automatic computation suffers from properties inherited from the fundamental definition of outer enclosures. Two of these overestimation occurrences will be looked into detail in this thesis.

Dependency problem

On the occurrence of identical interval variables, the computation will replace the basic operation with the respective interval evaluation without acknowledging their dependency. This would lead to a too pessimistic solution as, e.g. in a diameter calculation of the enclosure for the interval difference $[x] - [x]$ as

$$\operatorname{diam}\{[x] - [x]\} = 2 \cdot \operatorname{diam}\{[x]\} \tag{2.12}$$

instead of $\operatorname{diam}\{[x] - [x]\} = 0$. Note that there are no inverse elements to addition and multiplication in naive interval evaluation, further complicating calculations used for control engineering because of irreversibility of the operations.[1]
However, algorithmic approaches to reduce the dependency problem include

- Symbolic reformulations, e.g. for univariate polynomials such as

$$[f]([x]) = 2 \cdot [x] - [x] \cdot [x] \quad \text{with} \quad [x] = [-1 \ ; \ 2] \tag{2.13}$$

1. $[f]([x]) = 2 \cdot [x] - [x] \cdot [x] = [-6 \ ; \ 6]$
2. $[f]([x]) = 2 \cdot [x] - [x]^2 = [-6 \ ; \ 4]$
3. $[f]([x]) = -([x] - 1) \cdot ([x] - 1) + 1 = [-3 \ ; \ 3]$
4. $[f]([x]) = -([x] - 1)^2 + 1 = [-3 \ ; \ 1]$,

[1] A standard for interval arithmetic is defined in 1788–2015 — IEEE Standard for Interval Arithmetic.

- Taylor series expansions

$$f(x) \in f(x_\mathrm{m}) + \left(\sum_{i=1}^{n-1} \left. \frac{\partial^i f(x)}{\partial x^i} \right|_{x_\mathrm{m}} \cdot \frac{([x] - x_\mathrm{m})^i}{i!} \right) + \left. \frac{\partial^n f(x)}{\partial x^n} \right|_{[x]} \cdot \frac{([x] - x_\mathrm{m})^n}{n!} \tag{2.14}$$

  with[2] $x_\mathrm{m} = \mathrm{mid}\,\{[x]\}$ accounting for interval-based enclosures of all possible truncation errors,
- Branch&Bound-Algorithms
  1. Splitting of the original interval into disjoint (except for the bounds) subintervals
  2. Using the fundamental enclosure property (monotony of enclosure)

$$[\tilde{x}] \subseteq [x] \quad \Rightarrow \quad f([\tilde{x}]) \subseteq f([x]) \tag{2.15}$$

  3. Evaluation of the function with general methods for the subintervals
  4. Building the hull over the whole range of values,
- Use of the property of monotony of the function $f(x)$

$$\left. \frac{\partial f}{\partial x} \right|_{x \in [x]} < 0 \quad \Rightarrow \quad f \in [f(\overline{x})\ ;\ f(\underline{x})] \tag{2.16}$$

$$\left. \frac{\partial f}{\partial x} \right|_{x \in [x]} > 0 \quad \Rightarrow \quad f \in [f(\underline{x})\ ;\ f(\overline{x})] \tag{2.17}$$

  – efficiently combined with Branch&Bound-Algorithms
  – component-wise applicable for multiple interval arguments as well as vector-valued functions, and
- Definition of contractors, see [21].

Wrapping effect

Typically, complex shaped regions in a multi-dimensional state-space enclosing the flow of a set of ODEs or of discrete-time difference equations are replaced by simple shapes such as zonotopes [28, 32] or intervals [21]. Interval boxes, as shown in Fig. 1.1 as an order $n = 2$ example for Eq. (2.2) are set to be axis-aligned for the computation. When computing with these interval boxes, they may rotate and distort as shown in Fig. 2.2, second step.

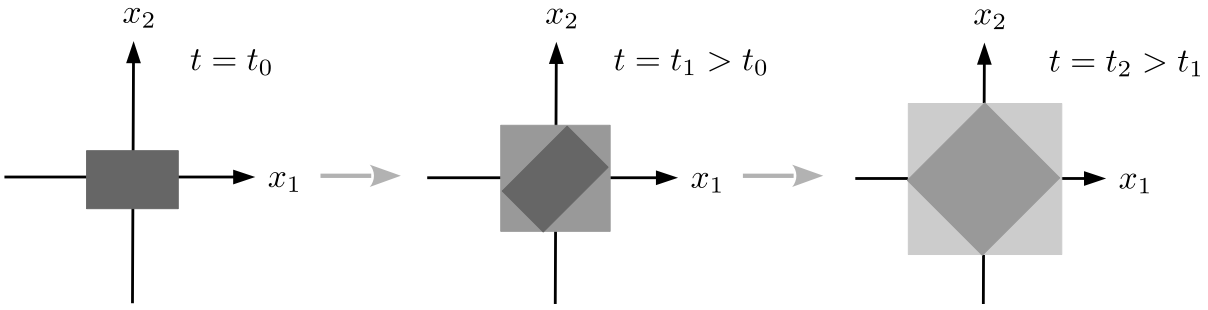


Figure 2.2: Wrapping effect: The initial interval box (left) is overestimated due to axis-alignment.

Since generally, computations are done with axis-aligned boxes, a wider box is built around the originally rotated box, leading to overestimation and hence, to (interval) bounds that are much wider than the actually reachable set.

[2] When expressed component-wise, this is also applicable trivially for vector-valued functions.

Simulation tool

Note that for a calculation on a CPU not all real values can be represented exactly as a finite-precision floating-point number of, e.g. data type `double`, which is why directed outward rounding of the interval bounds is used. All computations in this work are done in MATLAB using INTLAB (INTerval LABoratory toolbox) for reliable computing [57].

2.2 State-of-the-Art Computation of Interval Enclosures of all Reachable States: Taylor Series Expansion

A task for interval computation in control engineering is the computation of state enclosures, which can be used, e.g. for analyzing the system in general in terms of reachability, safety, and feasibility of trajectories as well as for optimizing controller gains based on the state evolution, or verifying controller as well as observer designs.

Stage 1

Consider the nonlinear, time-invariant dynamic system (1.8) to be described at some discrete-time sampling instants, so that $\mathbf{x}(t_k) \in [\mathbf{x}_k]$. Now, a *bounding box* $[\boldsymbol{\mathcal{B}}_k]$ can be determined which contains all possibly reachable states for $t \in [t_k \; ; \; t_{k+1}]$. This is done with the Picard iteration, see [14],

$$\Phi([\boldsymbol{\mathcal{B}}_k]) := [\mathbf{x}_k] + [0 \; ; \; T] \cdot \mathbf{f}([\boldsymbol{\mathcal{B}}_k]) \tag{2.18}$$

initialized with $[\boldsymbol{\mathcal{B}}_k] := [\mathbf{x}_k]$. On the one hand, if $\Phi([\boldsymbol{\mathcal{B}}_k]) \subset [\boldsymbol{\mathcal{B}}_k]$, the iteration is continued with $[\boldsymbol{\mathcal{B}}_k] := \Phi([\boldsymbol{\mathcal{B}}_k])$ until all interval diameters in the vector

$$\operatorname{diam}\{\Phi([\boldsymbol{\mathcal{B}}_k])\} = \sup(\Phi([\boldsymbol{\mathcal{B}}_k])) - \inf(\Phi([\boldsymbol{\mathcal{B}}_k])) \tag{2.19}$$

become practically identical to the diameters of the interval $[\boldsymbol{\mathcal{B}}_k]$. On the other hand, for $\Phi([\boldsymbol{\mathcal{B}}_k]) \not\subset [\boldsymbol{\mathcal{B}}_k]$, $[\boldsymbol{\mathcal{B}}_k]$ needs to be inflated before Eq. (2.18) is re-evaluated.

Stage 2

With this bounding box obtained by the Picard iteration, outer state enclosures at the time instant t_{k+1} are received by the temporal Taylor series

$$[\mathbf{x}_{k+1}] = [\mathbf{x}_k] + \sum_{i=1}^{\lambda} \frac{T^i}{i!} \mathbf{f}^{(i-1)}([\mathbf{x}_k]) + [\mathbf{E}_k] \;\; , \tag{2.20}$$

where typically $[\mathbf{x}_{k+1}] := [\mathbf{x}_{k+1}] \cap [\mathbf{B}_k] \subset [\mathbf{B}_k]$, of order λ with the guaranteed interval bounds $[\mathbf{x}_k]$ for $t = t_k$ and the enclosure

$$[\mathbf{E}_k] = \frac{T^{\lambda+1}}{(\lambda+1)!} \mathbf{f}^{(\lambda)}([\boldsymbol{\mathcal{B}}_k]) \tag{2.21}$$

of the temporal discretization error. Note that the expressions $\mathbf{f}^{(i)}$ in Eqs. (2.20) and (2.21) are the i-th order total derivatives of $\mathbf{f}$ with respect to time (proportional to the corresponding Taylor coefficients of the solution $\mathbf{x}_{k+1}$). The described interval enclosure (2.20) corresponds to the fundamental enclosure technique published in [40] and represents a common tool —

except for the applied QR preconditioning in VNODE, VNODE-LP[3] or AWA [31]— for the computation of interval enclosures of all reachable states. It will be used as a comparison method for a simulation-based gain-scheduled control design in this work.

2.3 Comments on Sensitivity

A common tool to handle intervals is their division into subintervals, c.f. the Branch&Bound-Algorithms of Sec. 2.1.2, which is then followed by building the hull over these subintervals to get to a tight solution. This is often done to counteract overestimation. The question now is, how to divide these intervals in an efficient manner. Efficiency — in this case — means to use the least necessary amount of subintervals to keep the effort as low as possible. This becomes even more important if the system has more than one uncertain parameter and we need to decide which to divide more often [18]. For this, a logical and simple approach is to consider the parameter with the most influence on the system. This is sometimes a question of physical knowledge of a real system, but can also be evaluated mathematically by making use of the so-called differential sensitivity [65]. In general, sensitivity describes the system's robustness against any kind of influence, whether it be external disturbances or parameter variations. As already mentioned, in this work, we will focus on the latter and furthermore restrict ourselves to sensitivity of eigenvalues. In [65], differential sensitivity, which is also called small-scale robustness, is described as a local characteristic defined by partial derivatives of first order with respect to a parameter p. Here, the derivative

$$s_\mathrm{p} = \left.\frac{\partial \mathbf{A}}{\partial p}\right|_{p=p_0} \tag{2.22}$$

characterizes the behavior of the system $\mathbf{A}$ in an infinitesimal vicinity of the nominal point p_0. If Eq. (2.22) is expanded to a parameter vector $\mathbf{p}$ with $\mathbf{p} = \begin{bmatrix} p_1 & p_2 & \dots & p_{n_\mathrm{p}} \end{bmatrix}^T$, a sensitivity vector $\mathbf{s}_\mathrm{p}$ is obtained, where its entries are a direct measure to judge the influence of the parameters on the system. This can also be transferred to deduce information on the parameter influence on eigenvalues, which will be required later in the presented thesis. In practice, it is sometimes difficult for real-life applications to derive the sensitivity symbolically, in analogy to Eq. (2.22) for

$$\mathbf{s}_\mathrm{p} = \frac{\partial \lambda(\mathbf{p})}{\partial \mathbf{p}} \ . \tag{2.23}$$

Hence, Eq. (2.23)[4] is expanded to

$$\mathbf{s}_\mathrm{p} = \frac{\partial \lambda(\mathbf{p})}{\partial \mathbf{p}} = (\mathbf{I} \otimes \mathbf{w}) \frac{\partial \mathbf{A}}{\partial \mathbf{p}} (\mathbf{I} \otimes \mathbf{v}) \ , \tag{2.24}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{w}$ the left eigenvector and $\mathbf{v}$ denotes the right eigenvector corresponding to the eigenvalue λ. Based on this information, obviously, parameters with higher sensitivity and large diameters will be subdivided more often than the others. Examples of this will be given in Chapters 4 and 7 of Part II.

[3] This QR preconditioning tries to find coordinate systems in which the boxes are axis-aligned. This may help in reducing the wrapping effect, however, if fitting coordinate systems can not be found, this may have the opposite effect.

[4] For the calculation rules see A.1.

2.4 Linear Matrix Inequalities and Their Use in Control Engineering

Consider a symmetric matrix $\mathbf{A}(\mathbf{p})$ that affinely[5] depends on certain variable parameters $\mathbf{p} \in \mathbb{R}$. The question is whether it is possible to choose those parameters in such a way that the eigenvalues' real parts

$$\Re\{\mathrm{eig}(\mathbf{A}(\mathbf{p}))\} < \mathbf{0} \tag{2.25}$$

become negative. Here, Linear Matrix Inequalities (LMIs) can be generally applied to a variety of problems considering the following [58]

- For positive definite LMIs $\mathbf{N}(\mathbf{p}) \succ \mathbf{0}$, $\mathbf{M}(\mathbf{p}) = -\mathbf{N}(\mathbf{p}) \prec \mathbf{0}$ holds.
- Matrix variables may appear as unknowns e.q. as $\mathbf{P}$ in the Lyapunov inequality

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} \prec \mathbf{0} \tag{2.26}$$

- If the maximum eigenvalue of the matrix $\mathbf{A}$ is $\lambda_{\max}(\mathbf{A}(\mathbf{p}))$, then $\Re\{\lambda_{\max}(\mathbf{A}(\mathbf{p}))\} < 0$ is equivalent to Eq. (2.26). Conditions on $\mathbf{p}$ are therefore generally defined by a non-smooth function. However, the solution set of Eq. (2.26) is always convex which allows for solving LMIs with adequate numerical algorithms.

A common tool for converting nonlinear (quadratic) inequalities into LMIs is the so-called Schur complement formula (A.3), where

$$\mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^T \succ \mathbf{0} \quad \text{with} \quad \mathbf{C} \succ \mathbf{0} \tag{2.27}$$

can be reformulated into

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} \succ \mathbf{0}, \quad \mathbf{A} = \mathbf{A}^T, \quad \mathbf{C} = \mathbf{C}^T \ . \tag{2.28}$$

This concept of LMIs is old and was firstly used for control engineering in 1940, where researchers tried to apply the Lyapunov theory (Section 2.5.2) for control applications [7]. However, LMIs were then solved by pen and paper and, hence, only applicable to low order problems. Recent developments in computer science opened LMIs to a broader field of applications, where nowadays different problems can be solved. In this work, LMIs are used either to solve a variety of (convex) optimization problems or to ensure robust stability of uncertain systems. While working with LMIs, the goal is to construct the inequalities describing the considered problem in such a way that there exists a feasible solution to them. The difficulty herein, therefore, lies in formulating suitable LMIs. All problems in this thesis were solved by the MATLAB toolbox SEDUMI [62] in combination with YALMIP [30].

2.5 Analyzing Stability of Dynamic Systems

Although there are many definitions of stability, the general idea is to determine if a system is "well behaved in some conceivable sense" [35]. To consider this, one needs a more mathematically defined way to decide on the system's behavior. For the sake of simplicity, we will refrain ourselves from the general stability theory and focus on the concept in control engineering. Here, mainly the stability of states corresponding to the system representation of Eq. (1.2)

[5] $\mathbf{A}(\mathbf{p})$ is affine in $\mathbf{p}$ if $\mathbf{A}(\mathbf{p}) = \mathbf{A}_0 + \sum_{i=1}^{n_\mathrm{p}} p_i \cdot \mathbf{A}_i$.

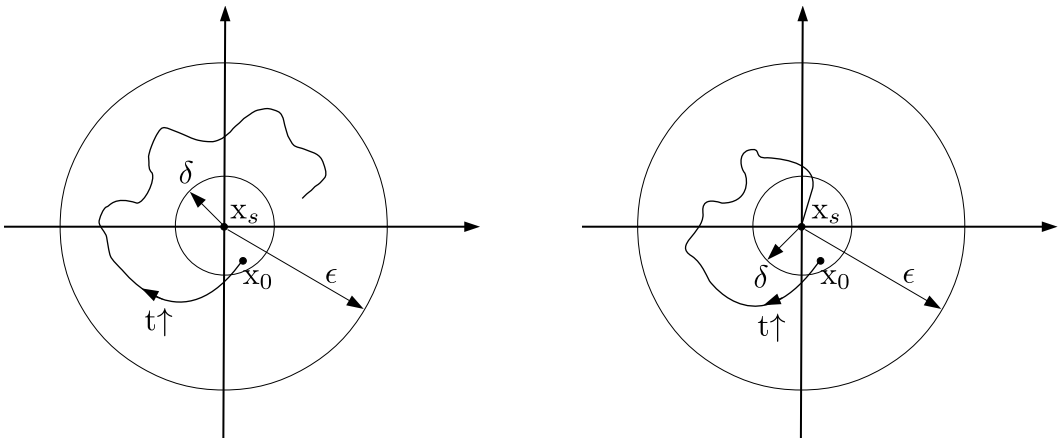
is considered. This idea is based on idle states or equilibrium points, where a stable system remains in its equilibrium state if not acted upon by an external disturbance with $\dot{\mathbf{x}} = \mathbf{0}$.
An idle state $\mathbf{x}_s$ of a dynamic system (1.8) is stable, if for an $\epsilon > 0$ there is a finite $\delta = \delta(\epsilon) > 0$, where

$$\|\mathbf{x}(0) - \mathbf{x}_s\| < \delta \quad \text{and} \quad \|\mathbf{x}(t) - \mathbf{x}_s\| < \epsilon \quad \text{for all} \quad t \geq t_0 \ . \tag{2.29}$$

Otherwise, the idle state $\mathbf{x}_s$ is unstable. Additionally, an idle state $\mathbf{x}_s$ is called attractive[6], if there is a finite $\delta_1 > 0$, where

$$\|\mathbf{x}(0) - \mathbf{x}_s\| < \delta_1 \quad \text{and} \quad \lim_{t \to \infty} \mathbf{x}(t) = \mathbf{x}_s \ . \tag{2.30}$$

If both Eqs. (2.29) and (2.30) hold and the idle state $\mathbf{x}_s$ is stable and attractive, it is called asymptotically stable. Figure 2.3 shows examples of the behavior for a stable and an asymptotically stable system, respectively. Here, one can clearly see that the trajectory starting in $\mathbf{x}(0) = \mathbf{x}_0$ remains inside the sphere with radius ϵ and in case of an asymptotically stable system additionally converges into the idle state $\mathbf{x}_s$.



(a) Idle state $\mathbf{x}_s$ is stable.

(b) Idle state $\mathbf{x}_s$ is asymptotically stable (stable and attractive).

Figure 2.3: Stability of the idle state on the basis of Eqs. (2.29) and (2.30).

The physical meaning of stability can be displayed in the resulting dynamics of a system, which should — in case of stable systems — not grow beyond their limits or even — in case of asymptotic stability — come to a rest. The total energy of a system provides a measure for this behavior. If the total energy does not increase, the system is stable. For asymptotic stability, the total energy will even decrease, e.g. due to an irreversible transformation of potential or kinetic into thermal energy in a mechanical system. Although there are several possibilities to prove stability of a dynamic system, the following summary will only cover a selection of them. All given theorems are used throughout this thesis for different kinds of stability as explained above.

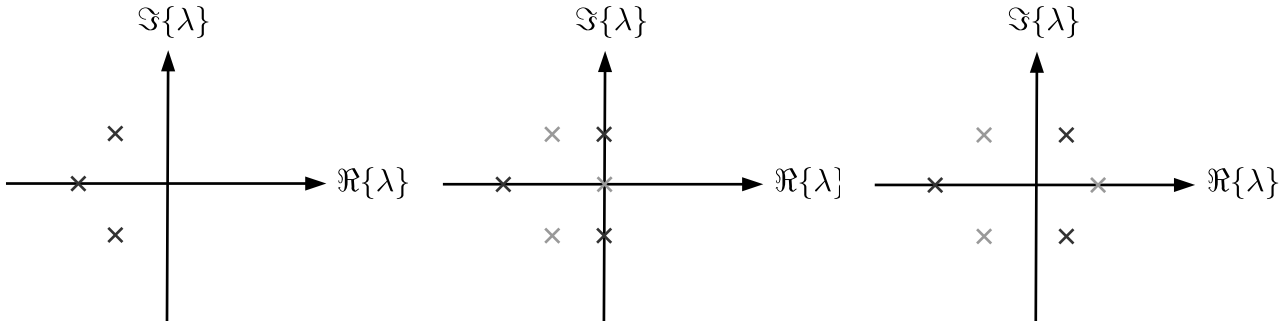
[6] convergent

2.5.1 Proofs of stability part I: linear systems

A simple stability statement for linear systems with crisp parameters[7] is to determine the location of all eigenvalues of the system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} \ . \tag{2.31}$$

If all eigenvalues lie in the open left half plane, meaning that all real parts are negative $\Re\{\lambda_i\} < 0$ with $i \in \{1, \ldots, n\}$, the system is asymptotically stable. For unstable systems, at least one real part of the eigenvalues is positive or eigenvalues of multiplicity two lie on the imaginary axis, and in case of stability one real eigenvalue or the real part of a conjugate-complex pair becomes zero (i.e. marginal stability), see Fig. 2.4.



(a) asymptotically stable **(b)** stable **(c)** unstable

Figure 2.4: Stability on the basis of eigenvalue locations for a system $n = 3$ with one real eigenvalue and a conjugate-complex pair.

Hurwitz criterion

This criterion makes use of the coefficients of the characteristic polynomial

$$p(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 \ , \tag{2.32}$$

which corresponds to $p(s) = \det(s\mathbf{I} - \mathbf{A})$ without pole-zero cancellation. Here, a so-called $n \times n$ Hurwitz-matrix is given by

$$\mathbf{H} = \begin{bmatrix} a_{n-1} & a_{n-3} & a_{n-5} & \ldots & a_1 & 0 & 0 & 0 & \ldots & 0 \\ a_n & a_{n-2} & a_{n-4} & \ldots & a_2 & a_0 & 0 & 0 & \ldots & 0 \\ 0 & a_{n-1} & a_{n-3} & \ldots & a_3 & a_1 & 0 & 0 & \ldots & 0 \\ 0 & a_n & a_{n-2} & \ldots & a_4 & a_2 & a_0 & 0 & \ldots & 0 \\ \vdots & & & & & \ddots & & & \vdots & \\ 0 & \ldots & 0 & a_{n-1} & a_{n-3} & a_{n-5} & \ldots & a_3 & a_1 & 0 \\ 0 & \ldots & 0 & a_n & a_{n-2} & a_{n-4} & \ldots & a_4 & a_2 & a_0 \end{bmatrix} \tag{2.33}$$

with its principal minors D_i, $i = 1, .., n$ according to

$$\mathbf{H} = \left[\begin{array}{c|c|c|c} \left| |a_{n-1}| \right| & a_{n-3} & a_{n-5} & \ldots \\ a_n & a_{n-2} & a_{n-4} & \ldots \\ 0 & a_{n-1} & a_{n-3} & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{array} \right] , \tag{2.34}$$

[7] Note that this is also valid for systems with uncertain parameters, if these parameters are unknown but otherwise time-invariant, i.e., constant.

which means

$$D_1 = a_{n-1} \ , \quad D_2 = \det \begin{pmatrix} a_{n-1} & a_{n-3} \\ a_n & a_{n-2} \end{pmatrix} \ , \quad \dots \quad . \tag{2.35}$$

Theorem 2.5.1 (Hurwitz criterion)
All zeros of the characteristic polynomial (2.32) have a negative real part, if and only if both of the following conditions are satisfied

1. *Necessary condition: All coefficients a_i are positive*

$$a_i > 0 \quad \textit{with} \quad i \in \{0, \dots, n\} \ .$$

2. *Necessary and sufficient condition: The n principal minors D_i of the matrix $\mathbf{H}$ are positive*

$$D_i > 0 \quad \textit{with} \quad i \in \{1, \dots, n\} \ .$$

Since the sufficient condition is also necessary, it is possible to only check the second condition. If the sufficient condition is not met, the regarded system is not asymptotically stable. However, this criterion will not give any information on which eigenvalue violates the conditions and to which extent. It therefore is a simple method to investigate whether all real parts lie in the open left half plane.

Kharitonov criterion

For a system of order n, where the interval coefficients a_i of the polynomial (2.32) vary independently

$$a_i \in [\underline{a}_i \ ; \ \overline{a}_i] \quad \text{with} \quad \underline{a}_i > 0$$

in a bounded interval, stability is given if all polynomials

$$\begin{aligned}
p_1(s) &= \overline{a}_0 + \underline{a}_1 s + \underline{a}_2 s^2 + \overline{a}_3 s^3 + \overline{a}_4 s^4 + \underline{a}_5 s^5 + \dots \\
p_2(s) &= \overline{a}_0 + \overline{a}_1 s + \underline{a}_2 s^2 + \underline{a}_3 s^3 + \overline{a}_4 s^4 + \overline{a}_5 s^5 + \dots \\
p_3(s) &= \underline{a}_0 + \overline{a}_1 s + \overline{a}_2 s^2 + \underline{a}_3 s^3 + \underline{a}_4 s^4 + \overline{a}_5 s^5 + \dots \\
p_4(s) &= \underline{a}_0 + \underline{a}_1 s + \overline{a}_2 s^2 + \overline{a}_3 s^3 + \underline{a}_4 s^4 + \underline{a}_5 s^5 + \dots
\end{aligned}$$

are independently proven to be Hurwitz stable. Besides the disadvantages of whatever stability proof used for the individual polynomials (e.g. Hurwitz), this approach involves a rather high conservatism for the regarded intervals if the coefficients a_i depend on each other and is therefore not used in this thesis.

Gershgorin circles

This approach estimates the placement of eigenvalues without knowing the exact value. It is therefore another interval representation directly applicable to uncertain systems. Here, the midpoint of a Gershgorin circle and its respective radius are

$$\Re\{\lambda_i\} \leq \underbrace{a_{ii}}_{\text{midpoint}} + \underbrace{\sum_{j=1, j\neq i}^{n} |a_{ij}|}_{\text{radius}} \tag{2.36}$$

where due to the constellation of the quadratic $n \times n$ matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1j} \\ a_{21} & a_{22} & \dots & a_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \dots & a_{ij} \end{bmatrix} \tag{2.37}$$

holds. If all complete circles lie in the open left half plane, it is proven, that the system is asymptotically stable. However, this may be a conservative approach since the worst-case scenarios are assumed for the eigenvalues, but overestimation may cause an asymptotically stable system to seem indistinguishable. Nevertheless, an exact knowledge of the eigenvalues is not needed.

2.5.2 Proofs of stability part II: nonlinear systems

The following methods generally allow for a consideration of nonlinear systems. This also includes possibilities to handle the stability of linear time-invariant (LTI) systems.

Lyapunov Criterion

On a former note, the total energy was considered as a measure for stability. Since it may be difficult to directly calculate the total energy of a system and its temporal variation, Lyapunov[8] extended his theory and replaced the positive definite energy function $E(\mathbf{x})$ with a more general function $V(\mathbf{x})$, also positive definite. This is called the direct method. Stability or asymptotic stability is given, if the time derivative $\dot{V}(\mathbf{x})$ is negative semi-definite [29] or definite, respectively, along the trajectories. A function $V(\mathbf{x})$ fulfilling this, is called a Lyapunov function with its time derivative

$$\dot{V}(\mathbf{x}) = \left(\frac{\partial V}{\partial \mathbf{x}}\right)^T \cdot \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \left(\frac{\partial V}{\partial \mathbf{x}}\right)^T \cdot \mathbf{f}(\mathbf{x}) \ . \tag{2.38}$$

In the following, the equilibrium point $\mathbf{x}_s = \mathbf{0}$ is assumed for simplicity; however, for an arbitrary equilibrium $\mathbf{x}_s \neq \mathbf{0}$, it can be shifted by transforming the respective coordinate system with

$$\mathbf{x} = \tilde{\mathbf{x}} + \mathbf{x}_s \tag{2.39}$$

resulting in

$$\dot{\tilde{\mathbf{x}}} = \mathbf{f}(\mathbf{x})\big|_{\mathbf{x}=\tilde{\mathbf{x}}+\mathbf{x}_s} \quad \text{with} \quad \tilde{\mathbf{x}}_s = \mathbf{0} \ . \tag{2.40}$$

Definition 2.5.1 (Local Lyapunov stability)
If $\mathbf{x}_s = \mathbf{0}$ is the equilibrium point[9] of a dynamic system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t))$, $\mathbf{f} : D \mapsto \mathbb{R}^n$ and if $V : D \mapsto \mathbb{R}$ is a continuously differentiable function, the equilibrium point $\mathbf{x}_s = \mathbf{0}$ is locally (asymptotically) stable, if

1. *$V(\mathbf{0}) = 0$*
2. *$V(\mathbf{x}) > 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$ and*

3a. *$\dot{V}(\mathbf{x}) < 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$ or*

3b. *$\dot{V}(\mathbf{x}) \leq 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$, respectively,*

is fulfilled.

[8] Aleksandr Lyapunov, 1857–1918

[9] Here, $\mathbf{x}_s = 0$ holds without loss of generality.

However, this does not indicate any information about the region of attraction of the equilibrium point and is only a sufficient criterion. For an equilibrium point to be globally asymptotically stable, it must be stable and all possible trajectories of the system must converge to it for $t \to \infty$. For this, the continuously differentiable function $V : D \mapsto \mathbb{R}$ also needs to be radially unbounded, which means $V(\mathbf{x}) \to \infty$ for $\|\mathbf{x}\| \to \infty$, resulting in the following definition.

Definition 2.5.2 (Global Lyapunov stability)
If $\mathbf{x} = \mathbf{0}$ *is the equilibrium point of a dynamic system* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t))$, $\mathbf{f} : \mathbb{R}^n \mapsto \mathbb{R}^n$ *and if* $V : \mathbb{R}^n \mapsto \mathbb{R}$ *is a continuously differentiable function, the equilibrium point* $\mathbf{x}_s = \mathbf{0}$ *is globally (asymptotically) stable, if*

1. $V(\mathbf{0}) = 0$
2. $V(\mathbf{x}) > 0$ *for all* $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$
3. $V(\mathbf{x}) \to \infty$ *for* $\|\mathbf{x}\| \to \infty$ *and*
4. $\dot{V}(\mathbf{x}) < 0$ *for all* $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$

is fulfilled.[10]

Note that an LTI system is stable if and only if it is quadratically stable. Often, a quadratic Lyapunov function according to

$$V(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} \tag{2.41}$$

for $\mathbf{x} \neq \mathbf{x}_\mathrm{s} = \mathbf{0}$ or, according to Krasowskij's approach, the following Lyapunov candidate may be suitable

$$V(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T\mathbf{f}(\mathbf{x}) \ . \tag{2.42}$$

For nonlinear time-variant systems, the reader is referred to [35]. For general applicability and in order to numerically determine Lyapunov function candidates $V(\mathbf{x})$, the so-called Lyapunov inequality on $\mathbf{P}$ may be used, which is a special form of an LMI [7]. A system (2.31) is quadratically stable if there exists a constant positive definite matrix $\mathbf{P} = \text{const.}$ such that

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} \prec 0 \tag{2.43}$$

holds. Here, for a quasi-linear model, global stability can be proven. However, a linearized system matrix reduces this to local stability. Note that the quadratic stability proof of Eq. (2.43) also holds for systems with time-varying parameters according to [7]. Concerning uncertain dynamic systems, a convex polytopic representation of parameter uncertainty according to

$$\mathcal{D} = \Big\{ [\mathbf{A}(\boldsymbol{\xi}), \mathbf{B}(\boldsymbol{\xi})] \Big| [\mathbf{A}(\boldsymbol{\xi}), \mathbf{B}(\boldsymbol{\xi})] = \sum_{\nu=1}^{n_\nu} \xi_v \cdot [\mathbf{A}_\nu, \mathbf{B}_\nu] \, ; \sum_{\nu=1}^{n_\nu} \xi_\nu = 1; \xi_\nu \geq 0 \Big\} \tag{2.44}$$

is used. To proof stability, a joint Lyapunov function V or — in case of quadratic stability — a positive matrix $\mathbf{P}$ has to be found for the union of all vertex systems $\nu \in \{1, \ldots, n_\nu\}$.

2.5.3 User-defined stability with Γ-regions

For a proof of stability of linear and quasi-linear systems, so-called Γ-regions are defined as subsets of the complex left half plane

$$\Gamma := \{ s \in \mathbb{C} \big| \mathbf{F}_\Gamma(s) \prec 0 \} \ , \tag{2.45}$$

[10] Note that for selected points, requirement 4. can be expanded to $\dot{V}(\mathbf{x}) \leq 0$ see [29].

see [1], with the linear matrix expression

$$\mathbf{F}_\Gamma(s) = \mathbf{D}_0 + s\mathbf{D}_1 + \bar{s}\mathbf{D}_1^T \ , \tag{2.46}$$

again, with s as the Laplace variable and $\bar{s}$ as its conjugate complex. Here, negative definiteness $\mathbf{F}_\Gamma \prec 0$ of the matrix in Eq. (2.46) needs to be fulfilled for all eigenvalues of the closed-loop system. The matrix inequality according to Eq. (2.45) and Eq. (2.46) is reformulated into another LMI in the following manner: If the set of all eigenvalues of a real-valued system matrix $\mathbf{A}$ is included in the interior of the region Γ, a positive definite matrix $\mathbf{P} = \mathbf{P}^T \succ 0$ can be determined that satisfies the matrix inequality

$$\mathbf{D}_0 \otimes \mathbf{P} + \mathbf{D}_1 \otimes (\mathbf{AP}) + \mathbf{D}_1^T \otimes (\mathbf{AP})^T \prec 0, \tag{2.47}$$

see [7, 59] and also Lyapunov stability [20] according to Eq. (2.43). Here, $\otimes$ is the Kronecker matrix product, see A.2, and $\mathbf{P}$ defines a Lyapunov function $V(\mathbf{x}) > 0$, see Eq. (2.41), with which stability of the autonomous dynamic system Eq. (2.31) can be proven, because (2.47) corresponds to $\dot{V}(\mathbf{x}) < 0$ along the trajectories outside the equilibrium.

User-defined Γ-stability regions can be composed of ellipses, hyperbolas, parabolas, cones, and strips in the complex plane, where for each of them the real-valued parameter matrices $\mathbf{D}_0 = \mathbf{D}_0^T$ and $\mathbf{D}_1$ have to be adjusted properly in Eq. (2.46). A selection of possible regions is shown in Fig. 2.5.

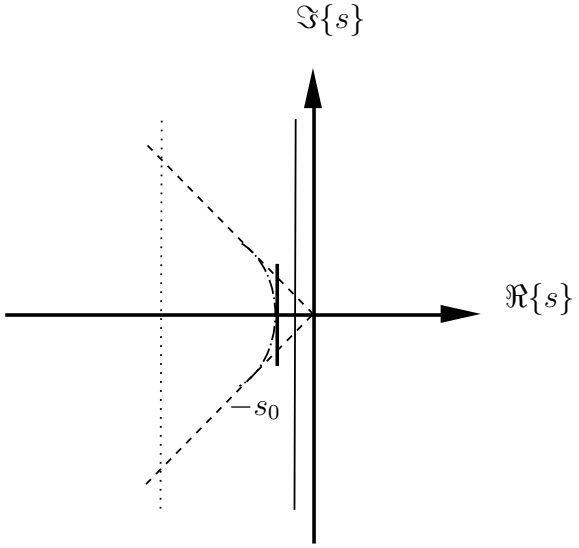


Figure 2.5: Possible stability regions. Absolute stability margin: vertical lines with the Γ-stability region on its left-hand side; limited bandwidth: dotted vertical line (or alternatively circle/ellipse) with stability region on the right-hand side of the boundary; minimum damping ratio: dashed lines; combination of minimum damping with absolute stability margin $\gamma = s_0$: hyperbolas (dashed-dotted) with opening to the left [24].

If the boundary of the stability domain Γ consists of multiple parts, all corresponding LMIs (2.47) need to be satisfied simultaneously. For example, to consider the absolute stability margin $\gamma = s_0 > 0$, corresponding to the design criterion that $\Re\{s\} < -s_0$ holds for all eigenvalues of the closed-loop control system, the Γ-region is chosen according to

$$F_\Gamma = 2\gamma + s + \bar{s} \prec 0 \tag{2.48}$$

and thus, corresponds to the setting $D_0 = 2\gamma$ and $D_1 = 1$. Note that pure Hurwitz stability, meaning $\Re\{s\} < 0$ in Eq. (2.45), is trivially included in this formulation by choosing $\gamma = 0$.

3 Control Synthesis

As stated in Chapter 1, a dynamic system (1.1) is given in a state-space model representation (1.2). Generally, a state feedback controller is designed according to

$$\mathbf{u} = -\mathbf{K}\mathbf{x} \tag{3.1}$$

with a constant gain $\mathbf{K}$, or as an adaptive controller by means of extended linearization techniques [4] for quasi-linear state-space representations of nonlinear systems (1.4) given by

$$\mathbf{u} = -\mathbf{K}(\mathbf{x}) \cdot \mathbf{x} \ . \tag{3.2}$$

This means that for a closed-loop system — omitting all arguments — and integrating (3.1) into (1.2)

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{u} = \mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{K}\mathbf{x} = \mathbf{A}_\mathrm{C}\mathbf{x} \tag{3.3}$$

with

$$\mathbf{A}_\mathrm{C} = (\mathbf{A} - \mathbf{B}\mathbf{K}) \tag{3.4}$$

holds. An easy way to calculate a suitable controller gain matrix $\mathbf{K}$ is based on Lyapunov stability, see Sec. 2.5.2. For this, the original LMI regarding Lyapunov stability (2.43) is interpreted for the closed-loop representation with Eq. (3.4) according to

$$(\mathbf{A} - \mathbf{B}\mathbf{K})^T \cdot \mathbf{P} + \mathbf{P} \cdot (\mathbf{A} - \mathbf{B}\mathbf{K}) \prec 0 \ . \tag{3.5}$$

Here, $\mathbf{P}$ is a matrix defining a Lyapunov function candidate

$$V(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} > 0 \tag{3.6}$$

for $\mathbf{x} \neq \mathbf{x}_\mathrm{s} = \mathbf{0}$, with which stability of the dynamic system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$ can be proven according to Section 2.5.2. Since Eq. (3.5) is a bilinear problem formulation with multiplicative coupling of the unknown $\mathbf{K}$ and $\mathbf{P}$, a reformulation into an LMI is necessary. For this, Eq. (3.5) is left and right multiplied by $\mathbf{P}^{-1}$ so that

$$\mathbf{P}^{-1} \cdot \Big[(\mathbf{A} - \mathbf{B}\mathbf{K})^T \cdot \mathbf{P} + \mathbf{P} \cdot (\mathbf{A} - \mathbf{B}\mathbf{K})\Big] \cdot \mathbf{P}^{-1} \prec 0 \ . \tag{3.7}$$

This can be expanded into

$$\mathbf{P}^{-1} \cdot (\mathbf{A} - \mathbf{B}\mathbf{K})^T + (\mathbf{A} - \mathbf{B}\mathbf{K}) \cdot \mathbf{P}^{-1} \prec 0 \tag{3.8}$$

or rather

$$\mathbf{P}^{-1}\mathbf{A}^T - \mathbf{P}^{-1}\mathbf{K}^T\mathbf{B}^T + \mathbf{A}\mathbf{P}^{-1} - \mathbf{B}\mathbf{K}\mathbf{P}^{-1} \prec 0 \ . \tag{3.9}$$

With a linearizing change of variables according to

$$\begin{aligned} &\mathbf{Q} := \mathbf{P}^{-1} \ , \quad \mathbf{Y} := \mathbf{K}\mathbf{P}^{-1} \ , \\ &\mathbf{P} = \mathbf{Q}^{-1} \ , \quad \mathbf{K} = \mathbf{Y}\mathbf{P} \quad \text{as well as} \quad \mathbf{Y}^T = \mathbf{P}^{-1}\mathbf{K}^T \ , \end{aligned} \tag{3.10}$$

the respective LMI is expressed as

$$\mathbf{Q}\mathbf{A}^T - \mathbf{Y}^T\mathbf{B}^T + \mathbf{A}\mathbf{Q} - \mathbf{B}\mathbf{Y} \prec 0 \; . \tag{3.11}$$

In this chapter, two basic designs of feedback controllers will be presented. The first will search for a constant controller gain $\mathbf{K}$ in terms of Eq. (3.1) accounting for all uncertainties by means of either uncertain parameters composed in the parameter vector $\mathbf{p}$ or uncertain states $\mathbf{x}$ as a joint solution for all possible scenarios over the whole time horizon. For this, Eq. (3.11) is reformulated for $\mathbf{p} \in [\mathbf{p}]$ into

$$\tilde{\mathbf{Q}}\left(\mathbf{A}(\mathbf{p}) - \mathbf{B}(\mathbf{p})\mathbf{K}\right)^T + \left(\mathbf{A}(\mathbf{p}) - \mathbf{B}(\mathbf{p})\mathbf{K}\right)\tilde{\mathbf{Q}} \prec 0 \tag{3.12}$$

with $\tilde{\mathbf{Q}} = \tilde{\mathbf{Q}}^T \succ 0$ as a stability requirement for the closed-loop dynamics. To replace the continuum of parameter dependencies $\mathbf{p}$, a polytopic model including the worst-cases of all vertex matrices is evaluated with the help of suitable LMIs, see Section 3.1. Since this means a quite large overestimation, adjustments in the modeling are introduced to reduce this in Section 3.2. Novel control designs extending the method of Sec. 3.1 are given in Section 3.3.

3.1 Robust LMI-Based Control Synthesis

Robustness is achieved in terms of an overapproximation of the (quasi-)linear system model by a polytopic uncertainty representation, see [59]. Generally, all uncertainties are treated as bounded parameter intervals, resulting in parameter-dependent system and input matrices $\mathbf{A}(\mathbf{p})$ and $\mathbf{B}(\mathbf{p})$. By introducing the vector $\boldsymbol{\xi} = [\xi_1 \; \ldots \; \xi_\nu]^T$, the polytopic model can be represented by

$$\mathcal{D} = \left\{ [\mathbf{A}(\boldsymbol{\xi}), \mathbf{B}(\boldsymbol{\xi})] \middle| [\mathbf{A}(\boldsymbol{\xi}), \mathbf{B}(\boldsymbol{\xi})] = \sum_{\nu=1}^{n_\nu} \xi_v \cdot [\mathbf{A}_\nu, \mathbf{B}_\nu] \, ; \sum_{\nu=1}^{n_\nu} \xi_\nu = 1; \xi_\nu \geq 0 \right\} \tag{3.13}$$

as a convex combination of extremal system models in (3.12) and suitably chosen vertex matrices $\mathbf{A}_\nu = \mathbf{A}_\nu(\mathbf{p})$ and $\mathbf{B}_\nu = \mathbf{B}_\nu(\mathbf{p})$, where each of them depends on the vector of independent parameters $\mathbf{p} \in \mathbb{R}^{n_\mathrm{p}}$ which influence the matrices $\mathbf{A}$ and $\mathbf{B}$ in an affine way. Those are contained in the interval box

$$[\mathbf{p}] = \left[\underline{\mathbf{p}} \; ; \; \overline{\mathbf{p}} \right] \tag{3.14}$$

with the component-wise defined bounds $\underline{p}_i \leq p_i \leq \overline{p}_i$, $i \in \{1, \ldots, n_\mathrm{p}\}$. An evaluation of $\mathbf{A}(\mathbf{p})$ and $\mathbf{B}(\mathbf{p})$ for each of the vertices

$$\mathcal{P} = \left\{ \begin{bmatrix} \underline{p}_1 \\ \underline{p}_2 \\ \vdots \\ \underline{p}_{n_\mathrm{p}} \end{bmatrix}, \begin{bmatrix} \overline{p}_1 \\ \underline{p}_2 \\ \vdots \\ \underline{p}_{n_\mathrm{p}} \end{bmatrix}, \ldots, \begin{bmatrix} \overline{p}_1 \\ \overline{p}_2 \\ \vdots \\ \overline{p}_{n_\mathrm{p}} \end{bmatrix} \right\} = \left\{ \mathbf{p}^{\langle 1 \rangle}, \ldots, \mathbf{p}^{\langle n_\nu \rangle} \right\} \tag{3.15}$$

results in vertex systems, which number $n_\nu = 2^{n_\mathrm{p}}$ is determined by the collection of independent parameters. All n_ν vertex systems need to be taken into account for the robust control design. To formulate a generally applicable method, we make use of the so-called Γ-region as explained in Section 2.5.3. Here, user-defined stability regions[1] are given in the form of (2.46). To achieve stability, the negative definiteness of $\mathbf{F}_\Gamma \prec 0$ needs to be satisfied for all eigenvalues of the system in consideration. To apply LMI design techniques, the inequality $\mathbf{F}_\Gamma \prec 0$ needs to be reformulated.

[1] such as ellipses, hyperbolas, parabolas, cones, and strips in the complex plane, see Sec. 2.5.3 and [1]

Theorem 3.1.1
If all eigenvalues of a real-valued system matrix $\mathbf{A}$ *lie within the interior of the region (2.46), a positive definite matrix* $\mathbf{P} = \mathbf{P}^T \succ 0$ *exists that fulfills the matrix inequality [59]*

$$\mathbf{D}_0 \otimes \mathbf{P} + \mathbf{D}_1 \otimes (\mathbf{AP}) + \mathbf{D}_1^T \otimes (\mathbf{AP})^T \prec 0 \ , \tag{3.16}$$

which is based on Eq. (2.43).

Again $\mathbf{P}$ is a matrix defining a Lyapunov function (3.6). The inequality (3.16) can be used for a direct control design. Analogously to Eqs. (3.7)–(3.11),

$$\mathbf{D}_0 \otimes \mathbf{P} + \mathbf{D}_1 \otimes (\mathbf{A}_\mathrm{C}\mathbf{P}) + \mathbf{D}_1^T \otimes (\mathbf{A}_\mathrm{C}\mathbf{P})^T \prec 0 \tag{3.17}$$

with (3.4), is multiplied left and right by $\mathbf{I} \otimes \mathbf{P}^{-1}$ resulting in

$$\mathbf{I} \otimes \mathbf{P}^{-1} \cdot \left(\mathbf{D}_0 \otimes \mathbf{P} + \mathbf{D}_1 \otimes (\mathbf{A}_\mathrm{C}\mathbf{P}) + \mathbf{D}_1^T \otimes (\mathbf{A}_\mathrm{C}\mathbf{P})^T\right) \cdot \mathbf{I} \otimes \mathbf{P}^{-1} \prec 0 \ . \tag{3.18}$$

Taking into account the calculation rules given in A.2, the left side product is

$$\left(\mathbf{D}_0 \otimes \mathbf{I} + \mathbf{D}_1 \otimes (\mathbf{P}^{-1}\mathbf{A}_\mathrm{C}\mathbf{P}) + \mathbf{D}_1^T \otimes \mathbf{A}_\mathrm{C}^T\right) \cdot \mathbf{I} \otimes \mathbf{P}^{-1} \prec 0 \ , \tag{3.19}$$

while the right side product leads to

$$\mathbf{D}_0 \otimes \mathbf{P}^{-1} + \mathbf{D}_1 \otimes (\mathbf{P}^{-1}\mathbf{A}_\mathrm{C}) + \mathbf{D}_1^T \otimes (\mathbf{A}_\mathrm{C}^T\mathbf{P}^{-1}) \prec 0 \ . \tag{3.20}$$

Since $\mathrm{eig}(\mathbf{A}_\mathrm{C}) = \mathrm{eig}(\mathbf{A}_\mathrm{C}^T)$ holds, we use

$$\mathbf{D}_0 \otimes \mathbf{P}^{-1} + \mathbf{D}_1 \otimes (\mathbf{P}^{-1}\mathbf{A}_\mathrm{C}^T) + \mathbf{D}_1^T \otimes (\mathbf{A}_\mathrm{C}\mathbf{P}^{-1}) \prec 0 \tag{3.21}$$

instead, resulting in

$$\mathbf{D}_0 \otimes \mathbf{P}^{-1} + \mathbf{D}_1 \otimes \left(\mathbf{P}^{-1}(\mathbf{A}^T - \mathbf{K}^T\mathbf{B}^T)\right) + \mathbf{D}_1^T \otimes \left((\mathbf{A} - \mathbf{BK})\mathbf{P}^{-1}\right) \prec 0 \ , \tag{3.22}$$

which is then expanded to

$$\mathbf{D}_0 \otimes \mathbf{P}^{-1} + \mathbf{D}_1 \otimes \left(\mathbf{P}^{-1}\mathbf{A}^T - \mathbf{P}^{-1}\mathbf{K}^T\mathbf{B}^T\right) + \mathbf{D}_1^T \otimes \left(\mathbf{A}\mathbf{P}^{-1} - \mathbf{BKP}^{-1}\right) \prec 0 \ . \tag{3.23}$$

With, again, a linearizing change of variables, see Eq. (3.10), this leads to

$$\mathbf{D}_0 \otimes \mathbf{Q} + \mathbf{D}_1 \otimes (\mathbf{Q}\mathbf{A}_\nu^T - \mathbf{Y}^T\mathbf{B}_\nu^T) + \mathbf{D}_1^T \otimes (\mathbf{A}_\nu\mathbf{Q} - \mathbf{B}_\nu\mathbf{Y}) \prec 0 \tag{3.24}$$

as the set of LMIs used to calculate the controller gain considering Lyapunov stability for the user-defined Γ-region. Here, the index $\nu \in \{1, \ldots, 2^{n_\mathrm{P}}\}$ denotes all possible vertices. For a robust stability of the uncertainty representation (3.13)–(3.15) with eigenvalues that are compatible with the domain $\mathbf{F}_\Gamma \prec 0$ defined in (2.46), a joint solution $\mathbf{Q} \succ 0$, $\mathbf{Y}$ of the LMI (3.24) for each of said vertices needs to be found.

3.1.1 Adding optimality criteria

Additionally to this, other optimality criteria such as robust H_2 and H_∞ tasks [65] can be taken into account by the same LMI-based design framework as shown in the following. For this, we consider the extended system to be

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_1\mathbf{w}(t) + \mathbf{B}_2\mathbf{u}(t) \ . \tag{3.25}$$

Here, $\mathbf{A}$, $\mathbf{x}$ and $\mathbf{u}$ are the already known system matrix, state vector, and input vector, respectively. The matrix $\mathbf{B}_2$ is the input matrix, while $\mathbf{B}_1$ denotes the disturbance input matrix and $\mathbf{w}$ the disturbance vector. For the H_∞ control design, additionally,

$$\mathbf{y}_\infty(t) = \mathbf{C}_\infty \mathbf{x}(t) + \mathbf{D}_{\infty 1}\mathbf{w}(t) + \mathbf{D}_{\infty 2}\mathbf{u}(t) \tag{3.26}$$

with the considered output $\mathbf{y}_\infty$, its output matrix $\mathbf{C}_\infty$ as well as the respective feedthroughs $\mathbf{D}_{\infty 1}$ for the disturbance and $\mathbf{D}_{\infty 2}$ for the input, holds. Analogously,

$$\mathbf{y}_2(t) = \mathbf{C}_2 \mathbf{x}(t) + \mathbf{D}_{22}\mathbf{u}(t) \tag{3.27}$$

is given for the H_2 control design with its respective output matrix and feedthrough.

LMI representation of the H_2 control design

The H_2-norm can be comprehended as a description of the signal energy relating a respective input to the output. For the control design,

$$||\mathbf{G}_{\mathbf{w}\to\mathbf{y}_2}(s)||_2^2 = \gamma_2^2 \stackrel{!}{=} \min \tag{3.28}$$

is desirable, where $\mathbf{G}_\mathbf{w}$ represents the reference transfer functions matrix. Depending on the investigated signal, this norm can either be maximized, if necessary, or minimized if the disturbance input is investigated, e.g. as canceling to white noise or as an input signal representing a disturbance. In this work, the latter is done and, hence, the H_2-norm is minimized and implemented by

$$\begin{bmatrix} \mathbf{Q}\mathbf{A}^T - \mathbf{Y}^T\mathbf{B}_2^T + \mathbf{A}\mathbf{Q} - \mathbf{B}_2\mathbf{Y} & \mathbf{B}_1 \\ \mathbf{B}_1^T & -\mathbf{I} \end{bmatrix} \prec 0 \tag{3.29}$$

as well as

$$\begin{bmatrix} \mathbf{Q} & \mathbf{Q}\mathbf{C}_2^T - \mathbf{Y}^T\mathbf{D}_{22}^T \\ \mathbf{C}_2\mathbf{Q} - \mathbf{D}_{22}\mathbf{Y} & \mathbf{Z} \end{bmatrix} \succ 0 \tag{3.30}$$

while minimizing $\text{trace}\{\mathbf{Z}\} < \gamma_2^2$.

LMI representation of the H_∞ control design

Describing the maximal amplification of the amplitude for each possible combination of inputs and outputs, the H_∞-norm can be used to minimize the control effort by complying with

$$||\mathbf{G}_{\mathbf{w}\to\mathbf{y}_\infty}(s)||_\infty < \gamma_\infty \ . \tag{3.31}$$

This is implemented by including the LMI

$$\begin{bmatrix} \mathbf{Q}\mathbf{A}^T - \mathbf{Y}^T\mathbf{B}_2^T + \mathbf{A}\mathbf{Q} - \mathbf{B}_2\mathbf{Y} & \mathbf{B}_1 & \mathbf{Q}\mathbf{C}_\infty^T - \mathbf{Y}^T\mathbf{D}_{\infty 2}^T \\ \mathbf{B}_1^T & -\gamma_\infty\mathbf{I} & \mathbf{D}_{\infty 1}^T \\ \mathbf{C}_\infty\mathbf{Q} - \mathbf{D}_{\infty 2}\mathbf{Y} & \mathbf{D}_{\infty 1} & -\gamma_\infty\mathbf{I} \end{bmatrix} \prec 0 \ , \tag{3.32}$$

while minimizing γ_∞ within the algorithm calculating the controller gain.

3.2 Reduction of Overestimation in the Polytopic Models

In the previous section, the polytopic model (3.13) was given by an axis-aligned box by considering Eq. (3.15). However, there are different representations of a polytopic model which will be discussed in this section. For this, Eq. (3.13) is reformulated into a more general description

$$\mathcal{D} = \operatorname{conv}\{\mathbf{A}_\nu\} = \left\{ \mathcal{D} = \sum_{\nu=1}^{n_\nu} \xi_v \cdot \mathbf{A}_\nu ; \sum_{\nu=1}^{n_\nu} \xi_\nu = 1; \xi_\nu \geq 0,\ \nu = 1, 2, \ldots, n_\nu \right\} \tag{3.33}$$

with

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{bmatrix} . \tag{3.34}$$

The axis-aligned approach takes into account all uncertain entries regardless of their dependence of one another. In the worst case, this leads to $2^{(m \times n)}$ vertex matrices. Of course, this can be reduced to the actual number of uncertain entries, however, overestimation will still occur. In previous works, possibilities to reduce this overestimation have been presented for linear systems as in [13] as well as for nonlinear systems, see [11].

3.2.1 Analysis of linear systems

As a linear system, an investigation of the following example

$$\mathbf{A} = \begin{bmatrix} 1 & \delta_2 \\ -3\delta_1 & -0.5\delta_2 - \delta_1 \end{bmatrix} \tag{3.35}$$

with $\delta_1 = [0.3\ ;\ 1.5]$ and $\delta_2 = [5\ ;\ 10]$ will be shown. Applying the axis-aligned approach as before, the polytopic model is given by $2^3 = 8$ vertex matrices[2] considering three independent uncertain entries, $a_{12} = \delta_2$, $a_{21} = -3\delta_1$, and $a_{22} = -(0.5\delta_2 + \delta_1)$ in the matrix $\mathbf{A}$. This leads to Fig. 3.1, where an axis-aligned box of the relation between a_{21} and a_{22} is represented.

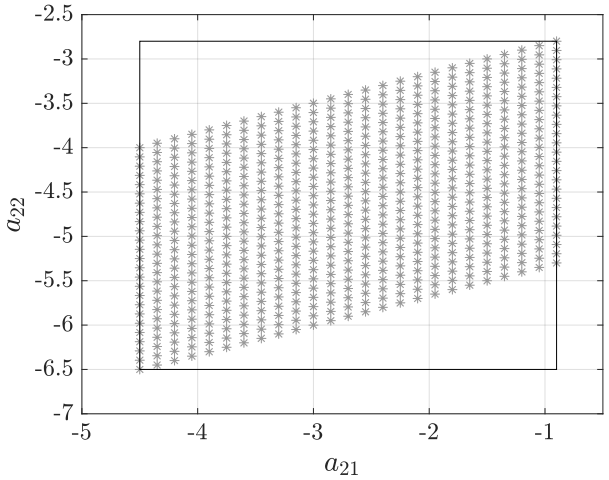


Figure 3.1: Example of an axis-aligned box for two uncertain entries for a linear system.

[2] as there is one known entry

Here, the black box represents the interval evaluation of (3.35). One can clearly see that due to the relation in the presented example entry a_{22} is widely overapproximated. A possibility to reduce this, is the so-called affine-linear approach, presented in [13]. Here, the dependency of the entries will be taken into account explicitly. For this, Eq. (3.35) is reformulated into

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \delta_1 \cdot \begin{bmatrix} 0 & 0 \\ -3 & -1 \end{bmatrix} + \delta_2 \cdot \begin{bmatrix} 0 & 1 \\ 0 & -0.5 \end{bmatrix} . \tag{3.36}$$

Through their dependency, the number of vertex matrices reduces to 2^{n_p}, where n_p is the number of actual uncertain parameters. For the given example, $n_\mathrm{p} = 2$, so that the system can be described by $2^2 = 4$ vertex matrices instead of 8. The reduced box, again for the relation between a_{21} and a_{22}, is shown in Fig. 3.2.

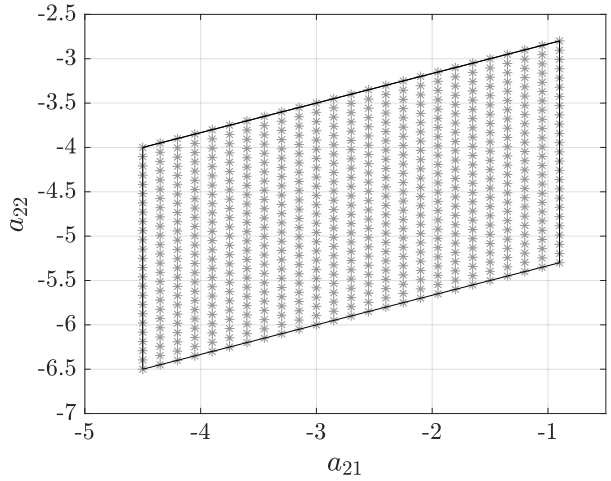


Figure 3.2: Example of an axis-aligned box for two uncertain entries considering their dependency in an affine-linear way.

Obviously, if done for nonlinear cases as their reformulation into the structure of Eq. (3.36), this would yield to a lot of overapproximation. The next section presents a possible solution to avoid that.

3.2.2 Analysis of nonlinear systems

As a numerical example for a nonlinear system

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\sin(\delta) & -0.5 + \delta^2 \end{bmatrix} \tag{3.37}$$

with $\delta = [0.3\ ;\ 1.5]$ is investigated. Applying the axis-aligned approach, the convex hull over a_{21} an a_{22} is presented by Fig. 3.3.

The idea is to linearize the matrix and then apply the methods from the previous subsection. For this, a Taylor linearization according to

$$\mathbf{A}^{(i)}(\boldsymbol{\varphi}) \approx \mathbf{A}(\boldsymbol{\varphi})\Big|_{\boldsymbol{\varphi}=\mathrm{mid}\{[\boldsymbol{\delta}_i]\}} + \frac{\partial \mathbf{A}(\boldsymbol{\varphi})}{\partial \delta}\Big|_{\boldsymbol{\varphi}=\mathrm{mid}\{[\boldsymbol{\delta}_i]\}} (\boldsymbol{\varphi} - \mathrm{mid}\,\{[\boldsymbol{\delta}_i]\}) \tag{3.38}$$

for $i = 1, \ldots, N_l$ linearization points of $[\boldsymbol{\delta}_i] \in \left[\underline{\boldsymbol{\delta}}_i\ ;\ \overline{\boldsymbol{\delta}}_i\right]$ with $\overline{\boldsymbol{\delta}}_{i-1} \leq \underline{\boldsymbol{\delta}}_i$ and $\overline{\boldsymbol{\delta}}_i \leq \underline{\boldsymbol{\delta}}_{i+1}$, cf. (2.14) is applied. For each of those points i, $j = 1, \ldots, 2^{(m \times n)}$ vertex matrices are given in the worst

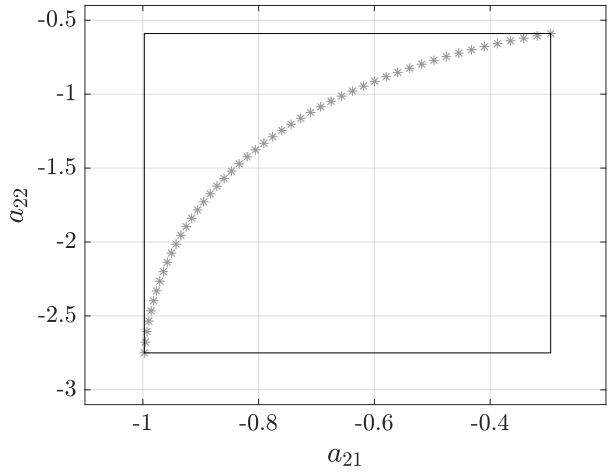
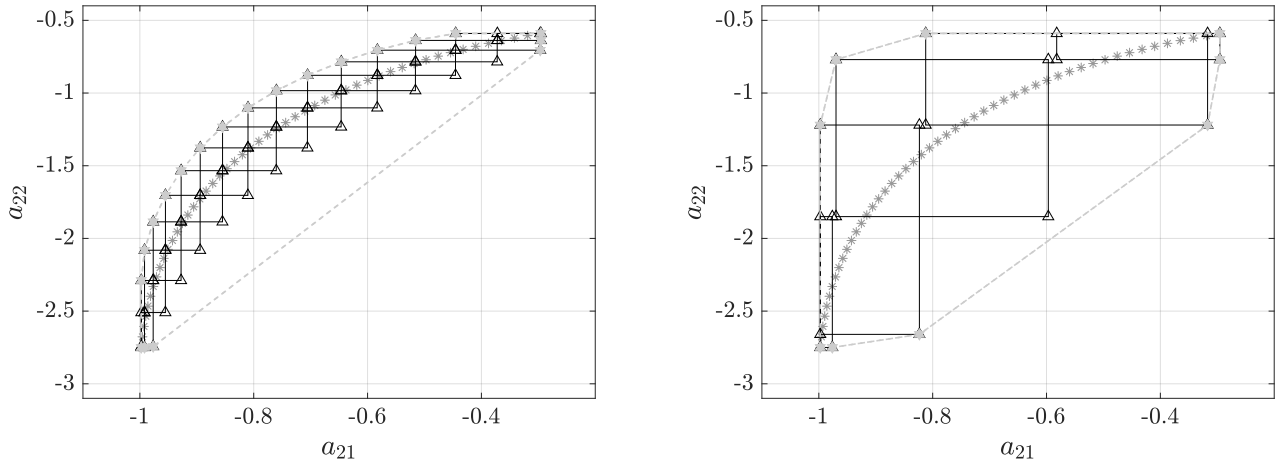


Figure 3.3: Example of an axis-aligned box for two uncertain entries for a nonlinear system.

case.[3] Obviously, choosing N_l has a rather big influence as the difference between Fig. 3.4(a) and 3.4(b) shows. However, a fine gridding would not be necessary, if (3.38) is evaluated with intervals instead of midpoints as in a verified Taylor linearization according to

$$\mathbf{A}^{(i)}(\boldsymbol{\varphi}) \in \mathbf{A}(\boldsymbol{\varphi})\Big|_{\boldsymbol{\varphi}=\mathrm{mid}\{[\boldsymbol{\delta}_i]\}} + \frac{\partial \mathbf{A}(\boldsymbol{\varphi})}{\partial \delta}\Big|_{\boldsymbol{\varphi}=[\boldsymbol{\delta}_i]} ([\boldsymbol{\delta}_i] - \mathrm{mid}\,\{[\boldsymbol{\delta}_i]\}) \tag{3.39}$$

again for $i = 1, \ldots, N_l$ linearization points. Here, small boxes would result, including the complete range. Nevertheless, a rougher gridding could lead to large boxes.



(a) Larger number of linearization points N_l.

(b) Smaller number of linearization points N_l.

Figure 3.4: Example of calculating a polytope using Taylor linearization and a convex hull.

However, the resulting union of vertex matrices is not guaranteed to be convex due to the gridding of the nonlinear relation. Hence, a convex hull is built using the MATLAB function `convhulln()`. This function makes use of the quickhull algorithm, see [3] resulting in the polytope

$$\mathcal{D} = \left\{ \texttt{convhulln}(\mathbf{A}_j^{(i)}) \right\} = \{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_{N_l}\} \ . \tag{3.40}$$

[3] Again assuming that all matrix entries are unknown.

In Fig. 3.4, the black axis-aligned boxes result from the vertex matrices $\mathbf{A}_j^{(i)}$ given by black triangles. The convex hull calculated by `convhulln` is given in dark gray with its respective vertex matrices marked as dark gray stars.
Advantages of this method include

- Expansion to higher dimensions possible,
- Dependency of uncertain parameters is included,
- Prevention of non-physical combinations by choice of N_l, and
- Linearization is already given for quasi-linear systems.

The disadvantage lies in the high number of vertex matrices, which means that a compromise between a tight enough enclosure and the resulting least possible number of vertex matrices needs to be found.

The above methods were measures to reduce overestimation occurring due to the modeling. In the following, an extended control design is presented aiming to reduce the conservativity of the control, especially in case of uncertainty due to nonlinearity.

3.3 Reducing Conservativity by Means of Gain Adaptation

As already mentioned, nonlinear systems are represented in a quasi-linear form in this work, see Eq. (1.4), and a suitable controller gain is given as (3.2) using extended linearization techniques. Since the state dependency is treated as an uncertainty, the controller gain calculated by the LMIs must cover the whole range of possible states. Due to the asymptotic stability, however, it is clear that this range needs to contract to the desired equilibrium for the controlled system. This leads to the idea to use this information to reduce conservativity in the controller gains by excluding those states from the synthesis that will not be reached. For this, we make use of two different approaches. A first, which will compute a constant gain for the complete time horizon, and a second, subdividing the time horizon into time slices with the aim of further reducing conservativity in each of those steps. Both procedures are also explained in [27].

3.3.1 Constant gain with robustness over the whole time horizon

For the first approach, the information gained over the complete time horizon is used in the following way.

- Firstly, a controller gain $\mathbf{K}_{\text{ini}}$ is computed for the interval domain of the initial states. Those are set by the user and usually contain the complete controllable domain including all assumed reachable states

$$\left[\boldsymbol{\mathcal{X}}^{(0)}\right] = \left[\check{\boldsymbol{\mathcal{X}}}\right] \quad \text{for} \quad t \in [t_0\,;\,t_\text{f}]\,. \tag{3.41}$$

- With the help of enclosure techniques to be discussed in this work, all reachable states $\left[\check{\boldsymbol{\mathcal{X}}}\right]$ are predicted for the whole time horizon.
- For a rough outer enclosure of the desired operating domain, this enclosure $\left[\check{\boldsymbol{\mathcal{X}}}\right]$ is inflated and then,
- subsequently tightened to the actually reachable interval domains via recalculating the controller gain.

Here, if

$$\left[\boldsymbol{\mathcal{X}}^{(i+1)}\right] \subseteq \left[\boldsymbol{\mathcal{X}}^{(i)}\right] \subseteq \left[\check{\boldsymbol{\mathcal{X}}}\right] \tag{3.42}$$

holds, the next iteration of the controller gains $\mathbf{K}^{(i+1)}$ is calculated using the hull over all intervals of the state considering the whole time horizon as the new less conservative interval bounds, see Fig. 3.5. Note that this demands a verified computation of the interval enclosures, which will be addressed in Chapter 4.

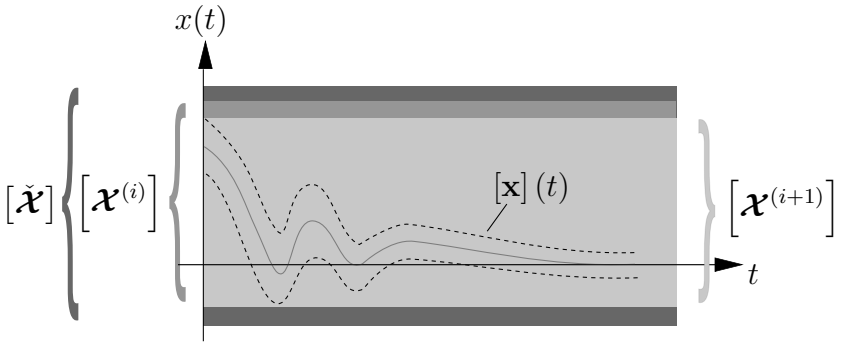


Figure 3.5: Illustration of the basic control approach for a scalar state variable.

Until the optimal solution with the smallest amount of conservativity regarding the controller gains is reached or if a final number of runs has been done, this scheme is repeated, see the structure in Fig. 3.6.

Set $\left[\check{\boldsymbol{\mathcal{X}}}\right]$ as the initial state domain $\left[\boldsymbol{\mathcal{X}}^{(0)}\right]$, $i := 0$
Final number of runs has not been reached $i \leq N \vee \left[\boldsymbol{\mathcal{X}}^{(i+1)}\right] \subseteq \left[\boldsymbol{\mathcal{X}}^{(i)}\right]$
Compute a robust stabilizing controller gain $\mathbf{K} = \mathbf{K}^{(i)}$ (according to the previous specifications, cf. Sec. 2.5.3) for the parameters $[\mathbf{p}]$ corresponding to the complete state interval $[\boldsymbol{\mathcal{X}}] := [\boldsymbol{\mathcal{X}}^{(i)}]$ which the system matrix depends on[4]
Calculate the interval enclosures for the complete time horizon $t = [t_0 \,;\, t_\mathrm{f}]$
Set the hull over all interval enclosures $[\boldsymbol{\mathcal{X}}^{(i)}] = \bigcup_{t \in [t_0 \,;\, t_\mathrm{f}]} [\mathbf{x}](t)$ as the new state domain
Increment the loop counter $i := i + 1$, output: gain $\mathbf{K} = \mathbf{K}^{(i)}$

Figure 3.6: Interval-based gain adaptation procedure for **Approach 1**.

This first approach can still be very conservative, which leads to the need of optimization in terms of a temporal series of controller gains in the next section.

3.3.2 Gain scheduling design over temporal subslices

Instead of a constant controller gain over the complete time horizon, this approach uses a sequence of control matrices $\mathbf{K}_k$, $k \in \{0, 1, 2, \ldots\}$, for fixed sampling times $T = t_{k+1} - t_k$. In

[4] Note that for the first run, this step is omitted and K_ini is used as a controller gain.

analogy to the approach presented in Section 3.3.1, the interval box

$$[\boldsymbol{\mathcal{X}}]\,(t_{k+1}) = [\boldsymbol{\mathcal{X}}_{k+1}] \tag{3.43}$$

denotes a verified enclosure of all states reachable at the point of time $t = t_{k+1}$. A general idea, presented in [23], is based on the fact that asymptotically stable convergence to the desired operating point $\boldsymbol{\mathcal{X}}_\mathrm{s} = \mathbf{0}$ w.l.o.g.[5] for $t \to \infty$ is ensured by the gain $\mathbf{K}_k$ if the system model (1.4) is time-invariant. Here, the aim is to acquire state enclosures for which the prediction result $[\boldsymbol{\mathcal{X}}_{k+1}]$ is a true subset of $[\boldsymbol{\mathcal{X}}_k]$ in all vector components according to

$$\mathbf{0} \in [\boldsymbol{\mathcal{X}}_{k+1}] \subset [\boldsymbol{\mathcal{X}}_k] \ . \tag{3.44}$$

If $[\boldsymbol{\mathcal{X}}_{k+1}] \not\subset [\boldsymbol{\mathcal{X}}_k]$, it cannot be proven that the interval width reduces in each of the vector components and, hence, the controller gain $\mathbf{K}_k$ is adapted to satisfy desired stability, robustness, and optimality criteria for the complete prediction step $t \in [t_k\,;\,t_{k+1}]$. Applying this for sufficiently large t, interval widths between two subsequent time steps should contract, as shown schematically in Fig. 3.7.

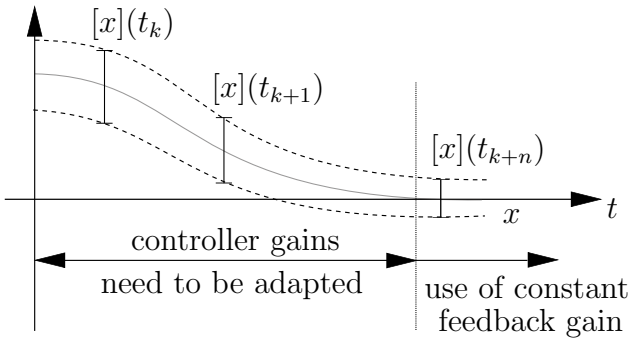


Figure 3.7: Principle of the control approach for a scalar state variable.

This was investigated for a practical application in [23], where, as expected, the gains remain constant in the vicinity of the equilibrium after a certain time. As an extension to this, the time steps can be chosen to not be constant but rather result from an intelligent step size control considering the individual behavior of time regions of the system, which have similar responses and subdividing only those with wider intervals. The procedure is schematically shown in Fig. 3.8.

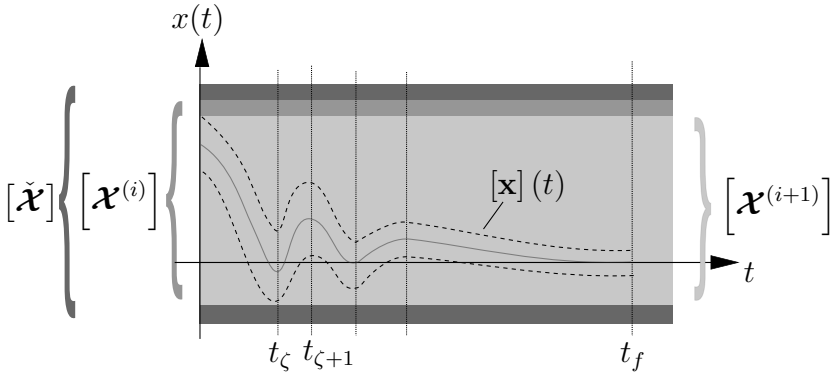


Figure 3.8: Illustration of the control approach for a scalar state variable.

This step-size control will later be extended with a strategy determining the most appropriate step size automatically. The approach with fixed and constant step sizes leads to an even longer

[5] without loss of generality

computation time than for the approach in Section 3.3.1. Analyzing the implementation, it becomes clear that the most consuming part of the computation lies within solving the LMIs for wide interval boxes. It is therefore evident, to divide the time horizon, so that various controller gains are used in the transient phase with wide interval boxes, while the later phase with its tight interval boxes is dominated by constant controller gains, see Fig. 3.8. To include this, the implementation of the previous approach in Sec. 3.3.1 is extended to Fig. 3.9.

Set $\left[\check{\boldsymbol{\mathcal{X}}}\right]$ as the initial state domain $\left[\boldsymbol{\mathcal{X}}^{(0)}\right]$, $i := 0$

Compute a robust stabilizing controller gain $\mathbf{K}_{\text{ini}}$ (according to the previous specifications, cf. Sec. 2.5.3) for the parameters $[\mathbf{p}]$ corresponding to the complete state interval which the system matrix depends on

Calculate the interval enclosures for the complete time horizon $t = [t_0\ ;\ t_\text{f}]$

Set the hull over all interval enclosures $[\boldsymbol{\mathcal{X}}^{(i)}] = \bigcup_{t\in[t_0\ ;\ t_\text{f}]} [\mathbf{x}]\,(t)$ as the new state domain

Analyze the sequence of interval enclosures for optimal switching points and get the respective list of time steps $t_\text{list} \in \{t_\zeta\}$, $\zeta \in \{1, \ldots, \zeta_\text{end}\}$[6]

End of new time vector has not been reached $t_\text{f} \neq t_\text{list}(\zeta_\text{end})$

- **Set** simulation time to $t_0 = t_\text{list}(\zeta)$ and $t_\text{f} = t_\text{list}(\zeta + 1)$
- Final number of runs has not been reached $i \leq N \vee \left[\boldsymbol{\mathcal{X}}^{(i+1)}\right] \subseteq \left[\boldsymbol{\mathcal{X}}^{(i)}\right]$
  - **Compute** a robust stabilizing controller gain $\mathbf{K} = \mathbf{K}^{(i)}$ (according to the previous specifications) for the parameters $[\mathbf{p}]$ corresponding to the complete state interval $[\boldsymbol{\mathcal{X}}] := [\boldsymbol{\mathcal{X}}^{(i)}]$ which the system matrix depends on
  - **Calculate** the interval enclosures for the complete time horizon $t = [t_0\ ;\ t_\text{f}]$
  - **Set** the hull over all interval enclosures $[\boldsymbol{\mathcal{X}}^{(i)}] = \bigcup_{t\in[t_0\ ;\ t_\text{f}]} [\mathbf{x}]\,(t)$ as the new state domain
  - **Increment** the loop counter $i := i + 1$, output: gain $\mathbf{K} = \mathbf{K}^{(i)}$

Figure 3.9: Interval-based gain scheduling procedure for **Approach 2**.

Here, an initial run of the first approach sets all parameters and finds a suitable division of the time horizon into time slices displaying the changes best. The new boxes featuring the hull over all reachable states are computed for each of the time steps t_ζ analogously to the first approach as shown in Fig. 3.6. Here, the very first run for each part of the time horizon, resulting from the subdivision before, is done with the initially chosen interval domain of the complete time horizon. By, again, repeating the computation for a specified number of iteration steps or until an optimal controller gain is found, the interval widths are gradually reduced with each iteration step adapting the controller gains. When the optimum for the current time step is found, the

[6] Here, ζ_end is defined by the user.

procedure moves on to the next time slice until the final time of the complete time horizon is reached.

3.4 Conclusion

This chapter presented control strategies for uncertain systems. Here, a state feedback control strategy was implemented with an LMI-based calculation of suitable controller gains. Uncertainty was given either by parameter uncertainty or by the state-dependency of quasi-linear representations of nonlinear dynamic systems. For both, different polytopic repesentations were discussed with respect to their enclosure properties regarding mathematical overestimation of given parameter variations. Considering this, constant controller gains were calculated, which were valid over a whole predefined time horizon. However, since the uncertainty given by state-dependency changes when applying the stabilizing controller, a new iterative control was introduced for such systems. Here, it was differentiated between a constant controller gain over the whole predefined time horizon, which is subsequently reduced for each rerun of the whole simulation and a gain scheduling design over temporal subslices also taking into account different phases of the control. In the next step, a transformation into a cooperative form will take place to determine reliable interval enclosures for the reruns used in this Chapter. Fig. 3.10 shows the progress so far in the overall approach in relation to Fig. 1.3.

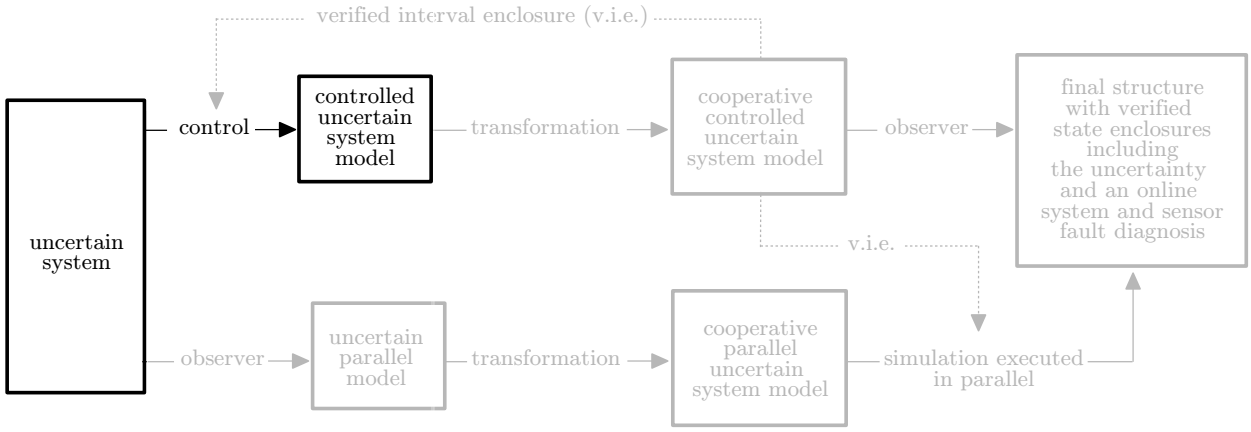


Figure 3.10: Outline of the theoretical aspects of this work - Step 1.

4 Transformation of Non-Cooperative Dynamical Systems into a Cooperative Form

As introduced in Chapter 1, cooperativity yields a lot of advantages which can be used in control engineering to improve computability of interval enclosures for state trajectories. Here, we make use of the decoupling of lower and upper bounds and, hence, can implement our tasks on two separate crisp systems as shown in Eq. (1.5). Assuming the system model is controllable and the desired operating state is set to $\mathbf{x} = \mathbf{x}_\mathrm{s} = \mathbf{0}$ w.l.o.g. for the steady-state input $\mathbf{u} = \mathbf{u}_\mathrm{s} = \mathbf{0}$, a feedback controller with the gain $\mathbf{K}$ is introduced as established in the previous Chapter 3 according to

$$\mathbf{u} = -\mathbf{K}\mathbf{x} \quad \text{or} \quad \mathbf{u} = -\mathbf{K}(\mathbf{x}) \cdot \mathbf{x} \;, \tag{4.1}$$

respectively, leading to a state-space representation

$$\dot{\mathbf{x}} = (\mathbf{A}(\mathbf{x}) - \mathbf{B}(\mathbf{x}) \cdot \mathbf{K}(\mathbf{x})) \cdot \mathbf{x} = \mathbf{A}_\mathrm{C}(\mathbf{x}) \cdot \mathbf{x} \;\; . \tag{4.2}$$

The presented approach now needs to make sure that asymptotic stability is not lost while additionally gaining cooperativity. So, for a system to be asymptotically stable *and* cooperative, it needs to fulfill two main requirements: the system matrix has to be both, Metzler and Hurwitz, simultaneously. In other words, the continuous-time system matrix must have non-negative entries outside its diagonal, but definitely negative ones on the diagonal itself. For observers, an uncertain system was already transformed into ODEs with a Metzler system matrix, e.g. in [43]. Due to the duality principle, this procedure can also be used for state prediction to be used in the control design. In general, a transformation corresponds to

$$\mathbf{z}(t) = \mathbf{\Theta}^{-1}\mathbf{x}(t) \quad \text{with} \quad \dot{\mathbf{z}}(t) = \mathbf{N} \cdot \mathbf{z}(t) \;, \tag{4.3}$$

where $\mathbf{\Theta}$ is invertible and $\mathbf{N}$ is Metzler. For general applications without diagonally dominant system matrices, the transformation consists of **Step 1**

$$\tilde{\mathbf{z}}(t) = \tilde{\mathbf{T}}^{-1}\mathbf{x}(t) \tag{4.4}$$

to get a diagonally dominant system matrix and **Step 2**

$$\mathbf{z}(t) = \mathbf{T}^{-1}\tilde{\mathbf{z}}(t) = \mathbf{T}^{-1} \cdot \tilde{\mathbf{T}}^{-1}\mathbf{x}(t) = (\tilde{\mathbf{T}} \cdot \mathbf{T})^{-1}\mathbf{x}(t) \tag{4.5}$$

to ensure a Metzler structure, resulting in the overall transformation matrix $\mathbf{\Theta}$, which may be a time-invariant or time-varying matrix. This distinction is based on the eigenvalues of the system, where for a system with purely real eigenvalues a time-invariant matrix denoted by $\mathbf{\Theta} = \mathbf{V} \cdot \mathbf{S}$ is sufficient, while systems with complex-conjugate pairs require time-varying transformation matrices further given with $\mathbf{\Theta} = \tilde{\mathbf{T}} \cdot \mathbf{T}$. In the following, the procedure is shown for exactly known systems and then extended to uncertain systems due to the previous distinction.

4.1 Crisp Parameter Systems

For a general description, the system matrix $\mathbf{A}_\mathrm{C}$ in Eq. (4.2) is considered as exactly known, constant and purely linear. In [43], this was done for an observer approach. There, a procedure similar to sensor placement is used to find a similarity transformation of $\mathbf{A}_\mathrm{O} = \mathbf{A} - \mathbf{H}\mathbf{C}_\mathrm{m}$ into a matrix $\mathbf{R}$ that has the same eigenvalues as the system matrix $\mathbf{A}_\mathrm{O}$. The sensor placement could be found in the choice of two vectors $\mathbf{e}_1^T$ and $\mathbf{e}_2^T$. If they can be chosen so that the pairs ($\mathbf{A}_\mathrm{O}$,$\mathbf{e}_1^T$) and ($\mathbf{R}$,$\mathbf{e}_2^T$) are observable, it is possible to calculate a transformation matrix $\mathbf{\Theta} = \mathbf{G}^{-1}\mathbf{F}$ with

$$\mathbf{F} = \begin{bmatrix} \mathbf{e}_1^T \\ \vdots \\ \mathbf{e}_1^T \mathbf{A}_\mathrm{O}^{n-1} \end{bmatrix} \quad \text{and} \quad \mathbf{G} = \begin{bmatrix} \mathbf{e}_2^T \\ \vdots \\ \mathbf{e}_2^T \mathbf{R}^{n-1} \end{bmatrix} . \tag{4.6}$$

Due to the duality principle, it is also possible to use the procedure in [43] to transform a controlled system, if the pairs ($\mathbf{A}_\mathrm{C}$, $\mathbf{e}_1$) and ($\mathbf{R}$, $\mathbf{e}_2$) are controllable setting $\mathbf{A}_\mathrm{C} = \mathbf{A}_\mathrm{O}$ into Eq. (4.6), respectively. In that case, the virtual sensor placement becomes an actuator placement and

$$\mathbf{F}_\mathrm{C} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{A}_\mathrm{C}\mathbf{e}_1 & \dots & (\mathbf{A}_\mathrm{C})^{n-1}\mathbf{e}_1 \end{bmatrix} \tag{4.7}$$

holds. Since the basic idea is, as mentioned, the identity of the eigenvalues, their importance for the transformation becomes clear. This underlines the fact that only stabilized or already originally stable systems should be transformed into a cooperative form to avoid shifting the eigenvalues later on and, hence, making the calculated transformation matrix invalid. The authors of [43] propose to make $\mathbf{R}$ a lower triangular matrix with the eigenvalues of $\mathbf{A}_\mathrm{C}$ on the main diagonal and positive elements below the main diagonal for systems with purely real eigenvalues. As a numerical example, the following system is considered. Let the stable system matrix be

$$\mathbf{A} - \mathbf{B}\mathbf{K} = \mathbf{A}_\mathrm{C} = \begin{bmatrix} 0 & 1 & 0 \\ -4 & -4 & 1 \\ -1 & 0 & 0 \end{bmatrix} \tag{4.8}$$

with the purely real eigenvalues $s_1 = -2.618$, $s_2 = -1$ and $s_3 = -0.382$ enforced by a controller gain $\mathbf{K}$. Following the presented approach, $\mathbf{R}$ is set to be

$$\mathbf{R} = \begin{bmatrix} -2.618 & 0 & 0 \\ 1 & -1 & 0 \\ 1 & 1 & -0.382 \end{bmatrix} \tag{4.9}$$

with a virtual actuator placement ensuring the controllability property of $\mathbf{e}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$ and $\mathbf{e}_2 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. Although if the pair $(\mathbf{A}, \mathbf{B})$ is controllable[1], those vectors always exist. Controllability can be proven with the help of a Kalman controllability matrix, see B.2. Additionally, the similarity transformations $\mathbf{F}_\mathrm{C}\mathbf{A}_\mathrm{C}\mathbf{F}_\mathrm{C}^{-1}$ and $\mathbf{G}_\mathrm{C}\mathbf{R}\mathbf{G}_\mathrm{C}^{-1}$ yield the canonical controllability forms[2] of $\mathbf{A}_\mathrm{C}$ and $\mathbf{R}$ where

$$\mathbf{F}_\mathrm{C}\mathbf{A}_\mathrm{C}\mathbf{F}_\mathrm{C}^{-1} = \mathbf{G}_\mathrm{C}\mathbf{R}\mathbf{G}_\mathrm{C}^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -4 & -4 \end{bmatrix} \tag{4.10}$$

[1] or $(\mathbf{A}, \mathbf{C})$ is observable, respectively

[2] Note that the system matrix in canonical observability form is the transposed of the system matrix in canonical controllability form, underlining the applicability to both observer and controller design interchangeably.

holds. The resulting transformation matrix is

$$\mathbf{\Theta} = {\mathbf{G}_\mathrm{C}}^{-1}\mathbf{F}_\mathrm{C} = \begin{bmatrix} 5.8541 & 4.2361 & -1.6180 \\ -5.4721 & -3.2361 & 1.6180 \\ 1 & 0 & 0 \end{bmatrix} \tag{4.11}$$

leading to a transformed system matrix

$$\mathbf{R} = \mathbf{\Theta}\mathbf{A}_\mathrm{C}\mathbf{\Theta}^{-1} = \begin{bmatrix} -2.618 & 0 & 0 \\ 1 & -1 & 0 \\ 1 & 1 & -0.382 \end{bmatrix} . \tag{4.12}$$

For systems with an order $n \leq 3$ this works reasonably well because a fitting structure for $\mathbf{R}$ can be found manually. However, finding a suitable matrix $\mathbf{R}$ becomes difficult for systems with higher orders or with several multiple eigenvalues as well as conjugate-complex eigenvalues. Additionally, if interval uncertainty is present in the matrix to be transformed, this procedure typically is not applicable. To extend the idea to uncertain system models, approaches for systems with both real and conjugate-complex eigenvalues are presented in the next sections.

4.2 Systems with Purely Real Eigenvalues

An extension of the previous section was proposed by the authors of [15], generalizing the presented method to applications for time-varying systems and systems with uncertainties. For that, the element-wise defined inequality

$$\mathbf{Z}_\mathrm{a} - \mathbf{\Delta} \leq \mathbf{Z} := \mathbf{A}_\mathrm{C} \leq \mathbf{Z}_\mathrm{a} + \mathbf{\Delta} \tag{4.13}$$

is used to express the uncertain system matrix $[\mathbf{A}]_\mathrm{C}$ with the (symmetric) worst-case bounds of all its entries $\mathbf{\Delta}$. Here, $\mathbf{Z}_\mathrm{a} = \mathbf{Z}_\mathrm{a}^T$ is the midpoint matrix, which is assumed to be symmetric in what follows. As previous, we search for a Metzler matrix $\mathbf{R} = \mu\mathbf{E}_n - \mathbf{\Gamma}$ with the same eigenvalues as $\mathbf{Z}_\mathrm{a}$. Here, $\mu \in \mathbb{R}$ is a constant and $\mathbf{\Gamma} \in \mathbb{R}^{n\times n}$ a diagonal matrix, where $\mathbf{\Gamma} = \rho\mathbf{I}_n$ with $\rho > \mu$ and the identity matrix $\mathbf{I}$ of order n. Additionally, $\mathbf{E}_n \in \mathbb{R}^{n\times n}$ is a matrix with all elements equal to 1. According to [15], the following theorem holds.

Theorem 4.2.1
If

$$\mathrm{eig}(\mathbf{R}) = \mathrm{eig}(\mathbf{Z}_\mathrm{a}) \ , \tag{4.14}$$

an orthogonal matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$ *exists, such that* $\mathbf{S}^T\mathbf{Z}\mathbf{S}$ *or* $\mathbf{\Theta}^T\boldsymbol{Z}\mathbf{\Theta}$*, respectively, is Metzler. This holds, if* $\mu > n||\mathbf{\Delta}||_\mathrm{max}$*, where* $||\mathbf{\Delta}||_\mathrm{max}$ *denotes the maximum absolute value of* $\mathbf{\Delta}$*.*

Although this approach makes the procedure more generally applicable, finding the transformation matrix $\mathbf{S}$ is still not trivial in several practical cases. To solve this issue, the approach is converted into a computationally feasible optimization problem formulated with LMI constraints, as also shown in [24], to systematically compute the time-invariant similarity transformation.
For the first step, finding $\mathbf{Z}_\mathrm{a}$ and $\mathbf{\Delta}$, two cases are distinguished. For systems with a diagonally dominant system matrix, $\mathbf{Z}_\mathrm{a}$ is chosen to represent the diagonal entries of the original. Yet, if the initial system matrix is not diagonally dominant, a new system matrix

$$\hat{\mathbf{A}}_\mathrm{C} = \mathbf{V}^{-1}\mathbf{A}_\mathrm{C}\mathbf{V} \tag{4.15}$$

is defined, where the element-wise computed interval midpoint matrix $\mathrm{mid}\{[\mathbf{A}]_\mathrm{C}\}$ is transformed into a diagonal structure (except for numerical round-off errors). Here, the matrix $\mathbf{V}$ is defined by the floating-point approximation of the n linear independent real-valued eigenvectors of $\mathrm{mid}\{[\mathbf{A}]_\mathrm{C}\}$, if they exist. After the transformation of Eq. (4.15), where round-off errors in the matrix inversion are handled by the use of interval arithmetic software libraries, $\mathbf{Z}_\mathrm{a}$ is set to be a diagonal matrix with the asymptotically stable, real eigenvalues of $\mathrm{mid}\{[\mathbf{A}]_\mathrm{C}\}$. In both cases, $\boldsymbol{\Delta}$ is given by $\boldsymbol{\Delta} = \delta \cdot \mathbf{E}_n$ with the worst-case bounds represented by

$$\delta = \max\left(|[\mathbf{A}]_\mathrm{C} - \mathbf{Z}_\mathrm{a}|\right) \quad \text{or} \quad \delta = \max\left(|[\hat{\mathbf{A}}]_\mathrm{C} - \mathbf{Z}_\mathrm{a}|\right) \ , \tag{4.16}$$

respectively. Note that in Eq. (4.16) the maximization is carried out over all matrix entries after determining their absolute values in an element-by-element manner and the `abs` value operation for interval matrices defined as in INTLAB. For the next step, the remarks from [15] are taken into account, including that $\mu^\star = n||\boldsymbol{\Delta}||_\mathrm{max}$ marks the lower bound for μ. Further remarks are reformulated into the requirements

$$\mathbf{R} = \mathbf{S}^T\mathbf{Z}_\mathrm{a}\mathbf{S} \ , \tag{4.17}$$

stating the equality of the eigenvalues, and

$$\mathbf{S}^T\mathbf{S} = \mathbf{I} \ , \tag{4.18}$$

ensuring orthogonality of the transformation matrix. To formulate these constraints into an optimization problem that calculates a suitable matrix $\mathbf{S}$, LMIs, see Chapter 2, are introduced. Hence, the aforementioned Eqs. (4.17) and (4.18) are relaxed into the positive definite matrix inequalities

$$-\mathbf{R} + \mathbf{S}^T\mathbf{Z}_\mathrm{a}\mathbf{S} \succ \mathbf{0} \tag{4.19}$$

and

$$\mathbf{I} - \mathbf{S}^T\mathbf{S} \succ \mathbf{0} \ . \tag{4.20}$$

To bound the norm of $\mathbf{S}$ from both, below and above, the signs of these inequalities are chosen as given in Eqs. (4.19)–(4.20) due to opposite signs of the quadratic terms in $\mathbf{S}$. These quadratic matrix inequalities can be converted into linear ones by the application of the Schur complement formula (see A.3) according to

$$\begin{bmatrix} -\mathbf{R} & \mathbf{S}^T \\ \mathbf{S} & -\mathbf{Z}_\mathrm{a}^{-1} \end{bmatrix} \succ \mathbf{0} \tag{4.21}$$

and

$$\begin{bmatrix} \mathbf{I} & \mathbf{S}^T \\ \mathbf{S} & \mathbf{I} \end{bmatrix} \succ \mathbf{0} \ , \tag{4.22}$$

respectively. Here, $\mathbf{R}$ is again defined as

$$\mathbf{R} = \bar{\mu}\mathbf{E}_n - \boldsymbol{\Gamma} \ , \quad \bar{\mu} > \mu \ . \tag{4.23}$$

The midpoint matrix to be transformed is assumed to be asymptotically stable if the LMI constraints

$$\boldsymbol{\Gamma} \succ \mathbf{0} \quad \text{and} \quad \mathbf{R}^T\mathbf{Q} + \mathbf{Q}\mathbf{R} \prec \mathbf{0} \tag{4.24}$$

with $\mathbf{Q} \succ \mathbf{0}$, e.g., here, set to $\mathbf{Q} = \mathbf{I}$ are fulfilled. To find a unique solution corresponding to Eqs. (4.19) and (4.20), the LMIs (4.21)–(4.24) are solved for $\mathbf{S}$, the diagonal matrix[3] $\boldsymbol{\Gamma}$, and the scalar $\bar{\mu}$ together with a minimization of the cost function

$$J = \mathrm{tr}(\boldsymbol{\Gamma}) + \mathrm{tr}(\mathbf{Z}_\mathrm{a}\mathbf{S} - \breve{\mathbf{S}}\mathbf{R}) - \kappa \cdot \mathrm{tr}(\breve{\mathbf{S}}^T\mathbf{S} - \mathbf{I}) \tag{4.25}$$

[3]Note that the matrix is not restricted to identical entries for all diagonal elements.

with the problem-dependent parameter $\kappa > 0$. Here, the first term minimizes the entries of $\mathbf{\Gamma}$, while the second and third minimize Eqs. (4.19) and (4.20), respectively, to meet the requirements of Eqs. (4.17) and (4.18). At the global minimum, Eqs. (4.19) and (4.20) are almost identical to Eqs. (4.17) and (4.18). Note that, the minimization of Eq. (4.25) in combination with the iterative solution of the LMIs in Eqs. (4.19) and (4.20) leads to the same results as the direct solution of the equality problem (4.17) and (4.18). The optimization task is solved in an iterative manner, to make the cost function (4.25) linear in $\mathbf{S}$. Here, $\breve{\mathbf{S}}$ denotes the solution of the last successful evaluation of the LMI-constrained optimization task. Additionally to the original approach [25], a line-search rule $\mu_+ = \mu + \Delta\mu$ with $\Delta\mu > 0$ gradually increasing the initial $\mu < \mu^\star$ until μ becomes equal to the desired value $\mu^\star$, is implemented to enhance the numerical convergence. This is done, because the solution for $\mu = 0$ corresponds to the known starting point $\mathbf{S} = \mathbf{I}$, specifying the initialization $\breve{\mathbf{S}} = \mathbf{I}$. An overview of the solution procedure can be seen in Fig. 4.1.

Determine the controlled system matrix $\mathbf{A}_\mathrm{C}$ with $\mathbf{A}_\mathrm{C} \in [\mathbf{A}]_\mathrm{C}$

Find initializing eigenvalues and eigenvectors for the midpoint of $[\mathbf{A}]_\mathrm{C}$ and keep them constant for the whole solution procedure

Determine $\mathbf{Z}_\mathrm{a}$ and $\mathbf{\Delta}$ to fulfill Eq. (4.13), possibly after transforming $[\mathbf{A}]_\mathrm{C}$ into diagonally dominant form (transformation matrix $\mathbf{V}$)

Set the initial transformation matrix to $\mathbf{S} = \mathbf{I}$, $\breve{\mathbf{S}} = \mathbf{I}$ for $\mu = 0$

Initialize μ and $\breve{\mathbf{S}}$ with the result of the last successful solution of the LMIs (4.21)–(4.24) in combination with the cost function (4.25)

Is there a solution to the LMIs and is J small enough to guarantee the desired Metzler property?

Yes

Set $\mu_+ = \mu + \Delta\mu$

$\mu := \mu_+$

No

Adjust μ and $\breve{\mathbf{S}}$ by small perturbations to enhance the solution quality

while $\mu < \mu^\star$

Output the complete transformation matrix $\mathbf{\Theta} = \mathbf{VS}$ (see Eqs. (4.3)–(4.5)) of the two-stage transformation procedure according to (4.15) and (4.17).

Figure 4.1: LMI-based computation of the transformation matrix $\mathbf{\Theta}$.

Be reminded, that $\mathbf{V}$ corresponds to $\tilde{\mathbf{T}}$ and $\mathbf{S}$ to $\mathbf{T}$ of the general transformation formulation of Eq. (4.5). Furthermore, the transformation can lead to unstable realizations of $\left[\tilde{\mathbf{A}}\right]_\mathrm{C}$ due to overestimation, which needs to be considered in the following.

4.2.1 Numerical examples

Example 1

A numerical example complying with Eq. (4.13) is introduced as

$$\mathbf{Z}_\mathrm{a} = \begin{bmatrix} -0.632 & 0 \\ 0 & -4.368 \end{bmatrix} \quad \text{and} \quad \mathbf{\Delta} = \begin{bmatrix} 0.8 & 0.8 \\ 0.8 & 0.8 \end{bmatrix} . \tag{4.26}$$

These numeric values correspond to those in [15] to allow for a comparison with or rather to perform a validation of the optimization method presented in this work. Here, the original system matrix

$$\mathbf{A}_\mathrm{C} = \begin{bmatrix} -0.632 - 0.8\sin(t) & 0.5\cos(3t) \\ 0.7\cos(2t) & 0.3\sin(t) \end{bmatrix} \tag{4.27}$$

was extended by a feedback controller, so that $\mathbf{A}_\mathrm{C}$ becomes diagonally dominant and, hence, does not need to be altered before the transformation into a Metzler matrix. If μ is set to equal the value given in [15] ($\mu = 1.8$), the presented approach calculates the same entries for

$$\mathbf{R}_{\mu=1.8} = \begin{bmatrix} -2 & 1.8 \\ 1.8 & -3 \end{bmatrix} \tag{4.28}$$

and

$$\mathbf{S}_{\mu=1.8} = \begin{bmatrix} -0.796 & -0.605 \\ 0.605 & -0.796 \end{bmatrix} \tag{4.29}$$

as given in [15]. However, if an optimization of μ is included, the optimal value is calculated to be $\mu = \mu^\star = 1.6$ and, hence, the entries of the transformation matrix deviate slightly from the ones before in Eq. (4.29)

$$\mathbf{S}_{\mu=1.6} = \begin{bmatrix} -0.871 & 0.492 \\ 0.492 & -0.871 \end{bmatrix} , \tag{4.30}$$

but are still in the solvable range. This outlines the importance and possibilities given by the value of μ regarding the variety of possible solutions for the transformation matrix $\mathbf{S}$ and explains the procedure of adapting μ along the iterative solution procedure as shown in Fig. 4.1.

Example 2

If the system matrix is not diagonally dominant as in

$$[\mathbf{A}]_\mathrm{C} = \begin{bmatrix} 0 & 1 \\ -1 & [-3.5\ ;\ -2.5] \end{bmatrix} , \tag{4.31}$$

a transformation of its midpoint into diagonal form becomes necessary. To achieve this, real-valued eigenvalues and eigenvectors are calculated for the midpoint matrix. Obviously, the eigenvalues λ_i are

$$\mathbf{\Lambda} = \begin{bmatrix} -0.382 & 0 \\ 0 & -2.618 \end{bmatrix} = \mathbf{Z}_\mathrm{a} . \tag{4.32}$$

The corresponding eigenvectors $\mathbf{V}$ are used for a transformation into diagonally dominant form, see Eq. (4.15), so that the new system matrix results in (here and afterwards the infima and suprema are displayed after outward rounding)

$$[\hat{\mathbf{A}}]_\mathrm{C} = \begin{bmatrix} [-0.467\ ;\ -0.297] & [-0.224\ ;\ 0.224] \\ [-0.224\ ;\ 0.224] & [-3.204\ ;\ -2.033] \end{bmatrix} . \tag{4.33}$$

According to Eq. (4.16), the uncertainty is bounded by $\delta = 0.5854$. Hence, the lower bound of μ results in $\mu^\star = 1.17$. To visualize that the novel LMI-based transformation procedure also works for larger bounds of the uncertain quantities, this value is replaced by $\mu = 1.25 \cdot \mu^\star$. The application of the solution procedure summarized in Fig. 4.1 leads to the transformation matrix

$$\mathbf{VS} = \begin{bmatrix} 0.976 & -0.218 \\ -0.488 & 0.873 \end{bmatrix} \tag{4.34}$$

and to the new, stable and Metzler, system matrix

$$[\tilde{\mathbf{A}}]_{\mathrm{C}} = \mathbf{S}^{-1}\mathbf{V}^{-1}\,[\mathbf{A}]_{\mathrm{C}}\,\mathbf{VS} \in \begin{bmatrix} [-0.500\ ;\ -0.357] & [0.191\ ;\ 0.448] \\ [0.000\ ;\ 0.639] & [-3.143\ ;\ -1.999] \end{bmatrix} \tag{4.35}$$

for both $\mu = \mu^\star$ as well as $\mu = 1.25 \cdot \mu^\star$. Finally, a check if overestimation has lead to instability is necessary. Here, the critical matrix for which stability needs to be verified in this case is the supremum matrix $\sup\{[\tilde{\mathbf{A}}]_{\mathrm{C}}\}$.

4.3 Systems with Conjugate-Complex Eigenvalues

In contrast to the previous section, for most uncertain systems with conjugate-complex eigenvalues only time-varying transformations are possible [37,43]. While previously the uncertainty was treated in the system matrix itself, here, it is mapped into the location of the eigenvalues. If the system matrix is evaluated for the whole range of parameters, this leads to a variability of the real and imaginary parts of conjugate-complex eigenvalues, which can be seen in Fig. 4.2 for an illustrative example for a system of order $n = 2$.

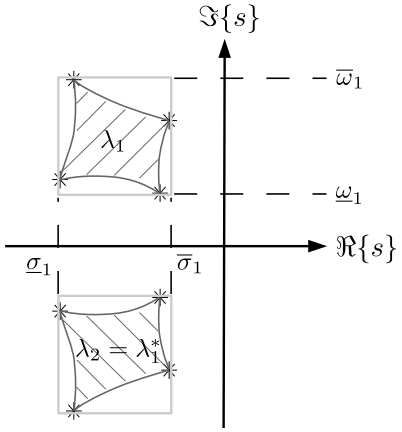


Figure 4.2: Possible locations of conjugate-complex eigenvalues.

Here, the vertices marked by asterisk symbols represent the location of the worst-case eigenvalues λ_i with $i \in \{1, \ldots, n\}$ for all possible vertex matrices. Those allow for axis-parallel boxes as a convex outer interval hull described by the extremal real and imaginary parts $[\sigma_i] = [\underline{\sigma}_i\ ;\ \overline{\sigma}_i]$ and $[\omega_i] = [\underline{\omega}_i;\ \ \overline{\omega}_i]$ to describe the uncertain system matrix in a polytopic form. When n is the (even) number of states and since two eigenvalues make up a conjugate-complex pair, assume $\tilde{n} = \dfrac{n}{2}$ guaranteed mutually disjoint conjugate-complex eigenvalue pairs. Eigenvalues cannot be distributed accordingly, if boxes as in Fig. 4.2 overlap. Hence, it is essential for the following approach that the eigenvalues are disjoint. In that case, there is a transformation matrix

$$\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right], \text{ where } \tilde{\mathbf{T}}_j \in \left[\tilde{\mathbf{T}}_j\right] = [\Re\{[\mathbf{v}_{\lambda j}]\}, \Im\{[\mathbf{v}_{\lambda j}]\}] \tag{4.36}$$

with $j \in \{1, \ldots, \tilde{n}\}$ consisting of interval enclosures for the real and imaginary parts of the eigenvectors of an uncertain system. The computation will be addressed at the end of this and in the subsequent sections. Extrema of the conjugate-complex eigenvalues are obtained by building the hull over their real and imaginary parts

$$\begin{aligned} [\sigma_j] &= [\min(\sigma_j)\ ;\ \max(\sigma_j)] \quad \text{and} \\ [\omega_j] &= [\min(\omega_j)\ ;\ \max(\omega_j)]\ . \end{aligned} \tag{4.37}$$

Applying that transformation, a block diagonal transformed system matrix is obtained as

$$\tilde{\mathbf{A}} = \text{blkdiag}\left(\tilde{\mathbf{A}}_1, \ldots, \tilde{\mathbf{A}}_{\tilde{n}}\right) \quad \text{with} \quad \tilde{\mathbf{A}}_j \in \left[\tilde{\mathbf{A}}_j\right] = \begin{bmatrix} [\sigma_j] & [\omega_j] \\ -[\omega_j] & [\sigma_j] \end{bmatrix} \tag{4.38}$$

representing the so-called real-valued Jordan canonical form. Here, the advantage lies in a direct display of real and imaginary parts of the eigenvalues. The time-varying transformation is done by

$$\mathbf{z} = \mathbf{T}^{-1}(t) \cdot \tilde{\mathbf{z}} \tag{4.39}$$

with

$$\mathbf{T}^{-1}(t) = \text{blkdiag}\left(\mathbf{T}_1^{-1}(t), \ldots, \mathbf{T}_{\tilde{n}}^{-1}(t)\right) = \mathbf{T}^T(t) \tag{4.40}$$

and the orthogonal blocks

$$\mathbf{T}_j \in [\mathbf{T}_j] = \begin{bmatrix} \cos([\omega_j]t) & \sin([\omega_j]t) \\ -\sin([\omega_j]t) & \cos([\omega_j]t) \end{bmatrix} \tag{4.41}$$

for $j \in \{1, \ldots, \tilde{n}\}$. Here, the outer interval enclosures of the imaginary parts of all eigenvalues containing the exact angular frequencies according to the relation $\omega_j^* \in [\omega_j^*] \subseteq [\omega_j]$ are taken into account to evaluate Eq. (4.41). Since the structure of the transformed system matrix (4.38) is known, the evaluation of this matrix is only necessary for

i) proving that the transformation leads to a system matrix in Metzler form and
ii) to determine enclosures $[\mathbf{z}](0)$ of the initial states as a function of $[\mathbf{x}](0)$ for verified simulations (and respectively for a backward transformation of the computed results for $t > 0$).

The state-space representation of the related differential equation (4.39) is calculated on the basis of the product rule of differentiation resulting in

$$\dot{\mathbf{z}} = \dot{\mathbf{T}}^T(t) \cdot \tilde{\mathbf{z}} + \mathbf{T}^T(t) \cdot \dot{\tilde{\mathbf{z}}} = \left[\left[\frac{\mathrm{d}\mathbf{T}^T(t)}{\mathrm{d}t} + \mathbf{T}^T(t)\tilde{\mathbf{A}}\right]\mathbf{T}(t)\right]\mathbf{z} = \mathbf{N} \cdot \mathbf{z}\ . \tag{4.42}$$

Applying this and symbolic simplifications in terms of the exact values ω_j^* to the transformed system matrix $\mathbf{N}$, it can be shown that it is Metzler with the real parts of the eigenvalues on the diagonal according to

$$\mathbf{N} = \text{blkdiag}\left(\sigma_1\mathbf{I}, \ldots, \sigma_{\tilde{n}}\mathbf{I}\right)\ ,\quad \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\ . \tag{4.43}$$

Considering interval values corresponding to Eq. (4.42), the diagonal elements of Eq. (4.43) are replaced by the lower and upper interval bounds to obtain the bounding systems. Since $\mathbf{N}$ is evaluated for the eigenvalues, Hurwitz stability is verified if the suprema are negative $\overline{\sigma}_j < 0$.

As previously mentioned, $\tilde{\mathbf{T}}$ from Eq. (4.36) needs to be determined. For this, we need to find the hull over all eigenvectors. However, the evaluation of the eigenvectors for the vertices

of a polytopic system model does not lead to the extremal values of the corresponding vector components from which a convex interval hull can be formed. A possible solution is to use the function `verifyeig` of INTLAB [57], see also Section 2.1.2, after subdividing the interval parameter domains of $[\mathbf{A}]_{\mathrm{C}}$ for smaller subintervals $\left[p^{\langle\kappa\rangle}\right] \subseteq [p]$ with $\kappa \in \{1 \dots L\}$ and then normalizing to length 1 (using the INTLAB routine `norm`). This subdivision is done, since the axis-parallel box over the n_{p} vertex matrices would be too conservative. One then uses the hull over all subintervals for the presented approach. This leads to two problems addressed in the following.

4.3.1 Division strategies for the parameters

In general, small subintervals are chosen as a countermeasure against overestimation in the eigenvalue and eigenvector computation. The aim here is to permit more variability inside the parameter boxes. However, this comes with higher calculation costs, which means, that we need to find a measure to subdivide the system's parameter intervals in an effective way without producing more subintervals than necessary. To tackle this, one can use a sensitivity analysis as described in Sec. 2.3. The higher the sensitivity of one uncertain parameter, the more subdivisions of this parameter we need in comparison to the rest. However, how many subintervals are needed depends on the system at hand and cannot be answered generally. Here, physical observations should be considered. After all, all those subintervals need to be emerged, which brings us to the next subsection.

4.3.2 Choosing the hull with the least amount of overestimation

Generally, the hull over all subintervals denotes the interval eigenvectors which are used to get $\tilde{\mathbf{T}}$. However, overestimation can occur in case of coupled parameters, if those are — unphysically — treated as decoupled. Since the final goal is to find the transformed states, three possibilities to calculate the initial state intervals accounting for these subintervals arise:

1. Using the hull over all eigenvectors corresponding to Eqs. (4.37) and (4.36) resulting in the transformation matrix

$$\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \dots, \tilde{\mathbf{T}}_{\tilde{n}}\right] \quad \text{with} \tag{4.44}$$

$$\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right]\right] \tag{4.45}$$

   and

$$\mathbf{T}_j \in \begin{bmatrix} \cos\left(\bigcup\limits_{\kappa=1}^{L} [\omega_j]t\right) & \sin\left(\bigcup\limits_{\kappa=1}^{L} [\omega_j]t\right) \\ -\sin\left(\bigcup\limits_{\kappa=1}^{L} [\omega_j]t\right) & \cos\left(\bigcup\limits_{\kappa=1}^{L} [\omega_j]t\right) \end{bmatrix} \tag{4.46}$$

   for Eqs. (4.40) and (4.41), while finally computing $[\mathbf{z}] = [\boldsymbol{\Theta}]^{-1} \cdot [\mathbf{x}]$ with $[\boldsymbol{\Theta}] = \left[\tilde{\mathbf{T}}\right] \cdot [\mathbf{T}]\,(t)$.

2. Using the hull over all transformation matrices

$$[\boldsymbol{\Theta}] = \bigcup_{\kappa=1}^{L} \left[\boldsymbol{\Theta}^{\langle\kappa\rangle}\right] = \bigcup_{\kappa=1}^{L} \left(\left[\tilde{\mathbf{T}}^{\langle\kappa\rangle}\right] \cdot \left[\mathbf{T}^{\langle\kappa\rangle}\right]\right) \tag{4.47}$$

   to calculate $[\mathbf{z}] = [\boldsymbol{\Theta}]^{-1} \cdot [\mathbf{x}]$.

3. Calculating the hull over all transformed initial ($t = 0$) state subintervals

$$[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right] \quad \text{with} \quad \left[\mathbf{z}^{\langle\kappa\rangle}\right] = \left[\boldsymbol{\Theta}^{\langle\kappa\rangle}\right]^{-1} \cdot [\mathbf{x}] \ . \tag{4.48}$$

When not using the third approach, the evaluation of Eqs. (4.44)–(4.47) can lead to excessively wide bounds for $\tilde{\mathbf{T}} \in \left[\tilde{\mathbf{T}}\right]$ if the complete possible domain is considered. If these bounds are too wide, the interval-valued inverse $\left[\tilde{\mathbf{T}}\right]^{-1}$ of $\left[\tilde{\mathbf{T}}\right]$ does not exist. A countermeasure is calculating the inverse as

$$\left[\boldsymbol{\Theta}^{-1}\right] = \bigcup_{\kappa=1}^{L} \left[\text{inv}\left(\boldsymbol{\Theta}^{\langle\kappa\rangle}\right)\right] = \bigcup_{\kappa=1}^{L} \left[\text{inv}\left(\left[\mathbf{T}^{\langle\kappa\rangle}\right]\right) \cdot \text{inv}\left(\left[\tilde{\mathbf{T}}^{\langle\kappa\rangle}\right]\right)\right] \tag{4.49}$$

to reduce overestimation. If this also fails, a possible alternative could be to include the complex, uncertain eigenvector structure directly in the transformation matrix $\tilde{\mathbf{T}} \in \mathbb{C}^{n\times n}$, rendering Eq. (4.36) respectively, into

$$\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right], \text{ where } \tilde{\mathbf{T}}_j = \left[\left[\mathbf{v}_{\lambda j}\right], \left[\mathbf{v}_{\lambda j}^*\right]\right] \ . \tag{4.50}$$

This would lead to a complex block diagonal matrix

$$\tilde{\mathbf{A}} = \text{ blkdiag}\left(\tilde{\mathbf{A}}_1, \ldots, \tilde{\mathbf{A}}_{\tilde{n}}\right) \in \mathbb{C}^{n\times n} \tag{4.51}$$

with

$$\tilde{\mathbf{A}}_j \in \left[\tilde{\mathbf{A}}_j\right] = \begin{bmatrix} [\sigma_j] + \jmath \cdot [\omega_j] & 0 \\ 0 & [\sigma_j] - \jmath \cdot [\omega_j]] \end{bmatrix} \ . \tag{4.52}$$

Besides omitting the time-varying transformation as the system matrix in Eq. (4.52) is already Metzler, another advantage is the decoupled structure of the diagonal matrix $\tilde{\mathbf{A}}$. This permits us to compute the interval enclosures in the new complex-valued coordinate frame symbolically. Now, the interval $[\mathbf{x}_0]$ of initial states needs to be transformed into the new coordinates $[\mathbf{z}_0] = \left[\tilde{\mathbf{T}}\right]^{-1} \cdot [\mathbf{x}_0]$ as above. Then, the simulation is performed using

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}}\mathbf{z} \ , \tag{4.53}$$

and the results are, after that, transformed backwards into $[\mathbf{x}] = \left[\tilde{\mathbf{T}}\right] \cdot [\mathbf{z}]$. Limits of this approach will be discussed in Chapter 9, cf. [44, 55].

4.3.3 Numerical example

A numerical example for the presented approach is given by the system model

$$[\mathbf{A}]_{\text{C}} = \begin{bmatrix} 0 & 1 \\ -a_{21} & -a_{22} \end{bmatrix} \quad \text{with} \quad a_{21} = \sin(\delta) \quad \text{and} \quad a_{22} = 0.5 + \delta^2 \ , \tag{4.54}$$

where δ is not exactly known but bounded with $\delta = [0.3\ ;\ 0.7]$. Fig. 4.3 presents a distribution of the eigenvalues computed by gridding the intervals. Here, at first, the parameters a_{21} and a_{22} were chosen to be independent — see black dots in Fig. 4.3 — and, hence, gridded

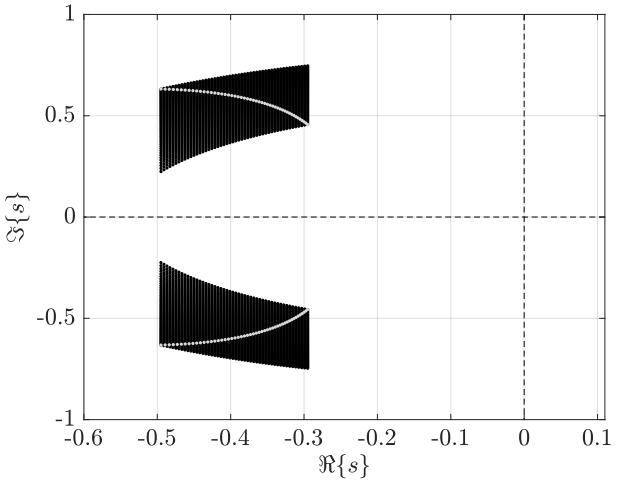


Figure 4.3: Distribution of eigenvalues for the numerical example with conjugate-complex eigenvalues.

separately. This independence, however, reflects the reality poorly. As a countermeasure, δ is gridded instead, followed by a calculation of the parameters a_{21} and a_{22} accounting for the dependence, which results in the gray dots in Fig. 4.3. It becomes clear that accounting for the parameter dependence can reduce overestimation drastically. For educational purposes, both approaches will be used for the transformation, despite the fact that for a real-life application, the approach considering the dependency is favorable. Since the sensitivity analysis for this scenario is trivial, because of only one uncertain parameter, there is no need to separately compute it and the reader is referred to the real-life application in Section 7.2. The number of subintervals is, hence, chosen arbitrarily for this numerical example. Next, the possibilities to reduce overestimation by building different hulls over those subintervals is considered. Furthermore, both transformations, the real-valued approach from Eqs. (4.36)–(4.42) as well as the complex-valued transformation of Eqs. (4.50)–(4.53) are performed.
At first, simulations for the real-valued transformations are given, where Fig. 4.4 shows the results for the first example, using independent parameters, while Fig. 4.5 takes into account the dependency. Both times, parameters were gridded into $N_l = 10$ subintervals. Here, both states are shown with their calculated interval bounds ($[x_1]$ in 4.4(a) as well as in 4.5(a) and $[x_2]$ in 4.4(b) and 4.5(b)) as well as with their respective interval diameters (diam $\{[x_1]\}$ in 4.4(c) as well as in 4.5(c) and diam $\{[x_2]\}$ in 4.4(d) and 4.5(d)). Additionally, as the states are coupled, the interval volume is given in 4.4(e) and 4.5(e). Note that since the eigenvalues are complex, the system is oscillating which is visible when the interval enclosures widen again after deflating in the first phase. This phenomenon can also be seen in the interval diameters and is amplified when overestimation is bigger in the first place. However, in the proximity of the stable operating point, the interval widths are deflating rapidly for both reductions. Obviously, the less conservative system deflates faster. Considering the different approaches for building the hull over the subintervals, one can clearly see the reduced overestimation, where the most conservative is Approach 1 followed by 2, and, with best results regarding the overestimation, Approach 3. Naturally, when considering the system with already lower overestimation, differences are not that strong. Given the results of applying the complex-valued transformation for this numerical example in Fig. 4.6 without considering the dependency and in Fig. 4.7 with the dependency, it becomes clear that this approach produces more overestimation when the parameter is gridded into the same number of $N_l = 10$ subintervals. In fact, the simulation results for independent parameters produced such wide intervals that a fair comparison is not possible even if N_l is increased, which would also come with a much higher computational effort. Note

that Fig. 4.6, therefore, shows the results with a different axes scale as Figs. 4.4, 4.5 and 4.7. This is omitted in the case of considering the dependency, so that a gridding of $N_l = 10$ was applied, see Fig. 4.7. Reasons for the increased overestimation may lie in the fact that for the backward transformation into the original coordinates $\mathbf{x}$, only the real part is considered and all information concerning the imaginary part is lost. However, for systems were a real-valued transformation is difficult to find — especially if concerning the inversion one might run into numerical problems — the complex-valued method may present an alternative.
To conclude, when using the approach with complex conjugate eigenvalues, it is necessary to include the most possible knowledge of the regarded system. If using general procedures, which are not application-specific, overestimation may increase especially for strongly nonlinear and high dimensional systems. This thought process is further explained in the applications in Part II, especially Chapter 7. This approach would be equally able to solve problems with purely real eigenvalues. However, this comes with larger overestimation than in Sec. 4.2 for the reasons given previously.

4.4 Extension to Systems with Mixed Eigenvalues: Real and Conjugate-Complex

Since the previous approach of Sec. 4.3 is equally applicable to both eigenvalue forms, an extension of it is developed in this section. For that, assume that the list of mutually distinct eigenvalues contains $\tilde{n} < \frac{n}{2}$ complex pairs, leading to $n^* = n - 2\tilde{n} \geq 0$ real eigenvalues. For general applicability, the eigenvalues are sorted in such a way that all complex pairs are listed first, see also [49]. Now, the block diagonal structure of Eq. (4.38) is rewritten so that

$$\tilde{\mathbf{A}} = \text{blkdiag}\left(\tilde{\mathbf{A}}_1, \ldots, \tilde{\mathbf{A}}_{\tilde{n}+n^*}\right) \tag{4.55}$$

is given, where a pair of conjugate-complex eigenvalues with $\omega_i = -\omega_{i+1}$, $i \in \{1, 3, \ldots, 2\tilde{n}-1\}$, leads to the known second part of Eq. (4.38) with $j \in \{1, 2, \ldots, \tilde{n}\}$, while an uncertain real eigenvalue ($i \in \{2\tilde{n}+1, 2\tilde{n}+2, \ldots, n\}$) is reflected by

$$\tilde{\mathbf{A}}_j \in [\sigma_i] \ , \quad j = i - \tilde{n} \ . \tag{4.56}$$

The respective transformation matrix formerly given in Eq. (4.36) changes into

$$\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}+n^*}\right] \tag{4.57}$$

again with the second part of Eq. (4.36) for $j \in \{1, 2, \ldots, \tilde{n}\}$ and

$$\tilde{\mathbf{T}}_j \in \left[\tilde{\mathbf{T}}_j\right] = [\mathbf{v}_{\lambda j}] \ , \quad j = i - \tilde{n} \ , \tag{4.58}$$

for the real eigenvalues $i \in \{2\tilde{n}+1, 2\tilde{n}+2, \ldots, n\}$. The time-varying transformation changes from (4.40) to

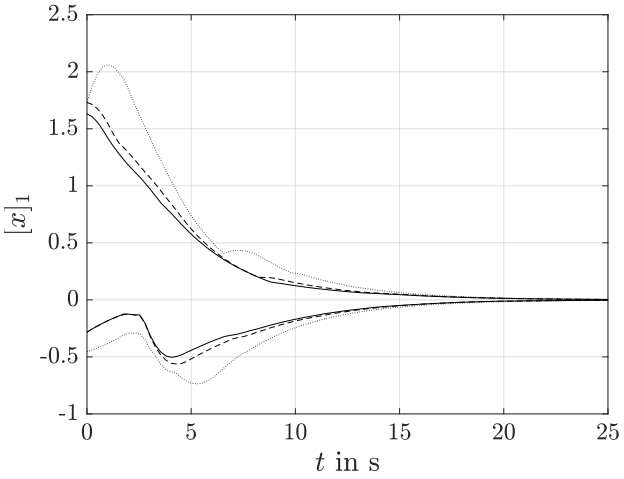
$$\mathbf{S}(t) = \text{blkdiag}\left(\mathbf{S}_1(t), \ldots, \mathbf{S}_{\tilde{n}+n^*}(t)\right) = \left(\mathbf{S}^{-1}(t)\right)^T \tag{4.59}$$

with the orthogonal blocks as in Eq. (4.41) for $j \in \{1, 2, \ldots, \tilde{n}\}$ and

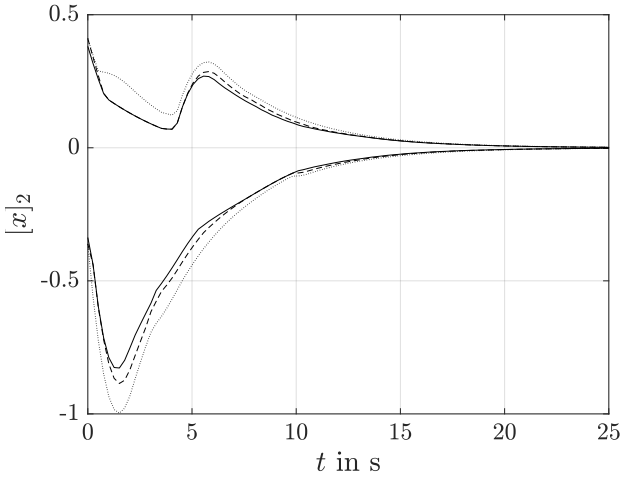
$$\mathbf{S}_{\tilde{n}+1} = \ldots = \mathbf{S}_{\tilde{n}+n^*} = 1 \ . \tag{4.60}$$

The state-space representation of the related differential equation (4.42) still holds and the transformed system matrix becomes Metzler according to

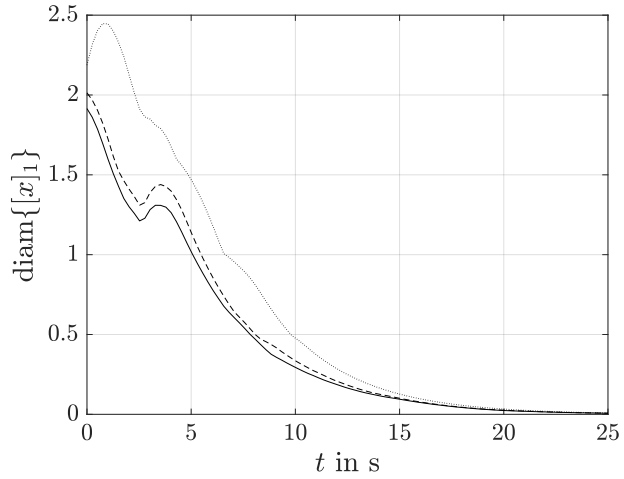
$$[\mathbf{N}] = \text{blkdiag}\left([\sigma_1] \cdot \mathbf{I}, \ldots, [\sigma_{\tilde{n}}] \cdot \mathbf{I}, [\sigma_{\tilde{n}+1}], \ldots, [\sigma_{\tilde{n}+n^*}]\right) \quad \text{with} \quad \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \ . \tag{4.61}$$



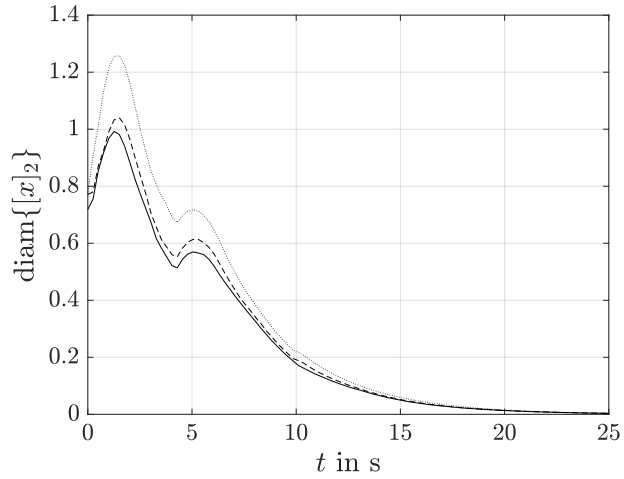
(a) Interval enclosures of state x_1



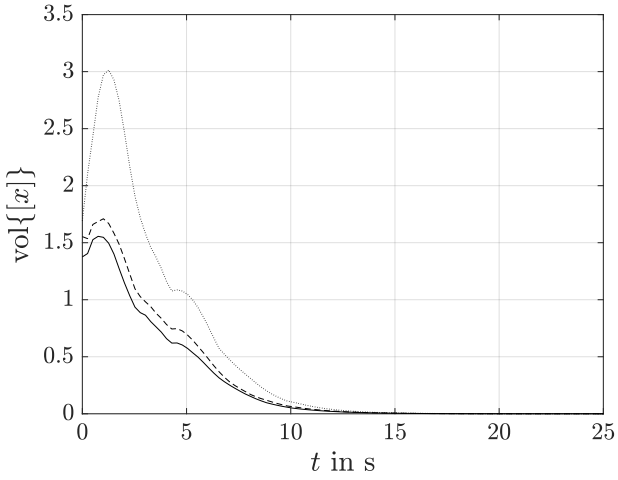
(b) Interval enclosures of state x_2



(c) Interval diameters of state x_1

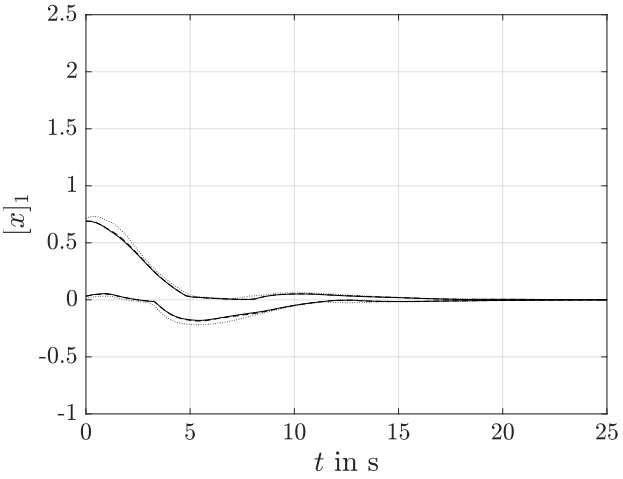


(d) Interval diameters of state x_2

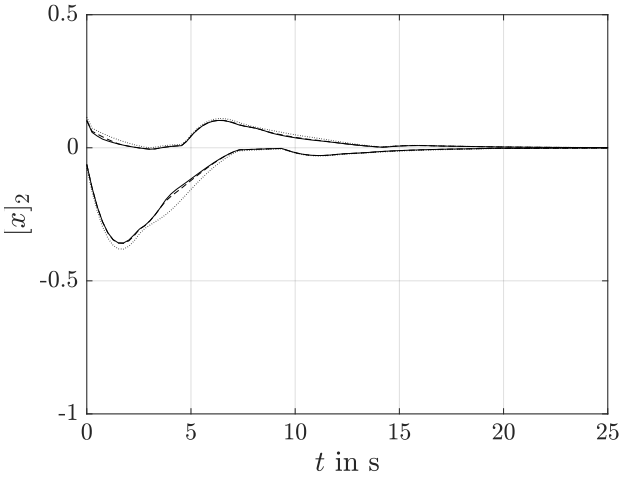


(e) Volume of the $n = 2$ interval box

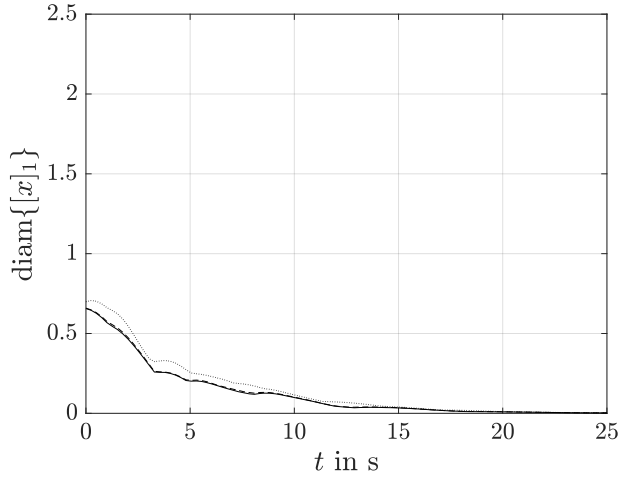
Figure 4.4: Comparison of three approaches — independent treatment of parameters a_{21} and a_{22} — real-valued transformation:
1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right]\right]$ – dotted,
2. $[\mathbf{T}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{T}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.



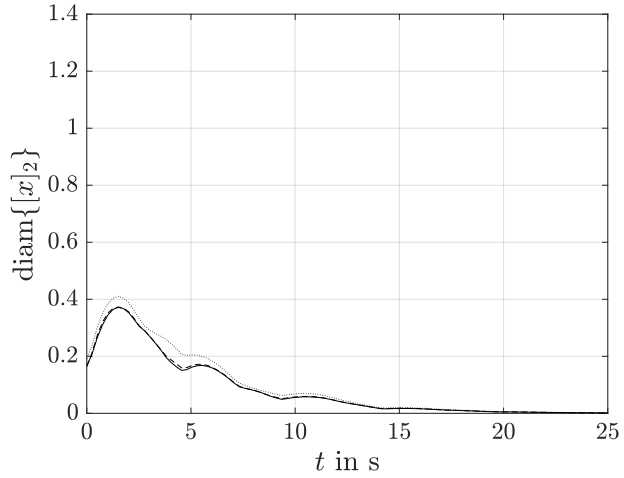
(a) Interval enclosures of state x_1



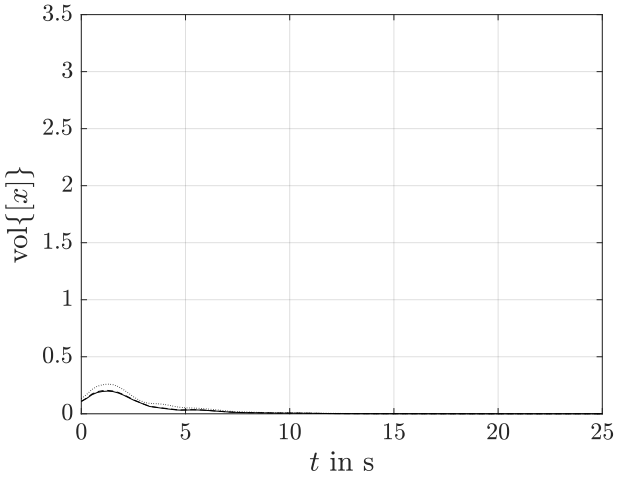
(b) Interval enclosures of state x_2



(c) Interval diameters of state x_1

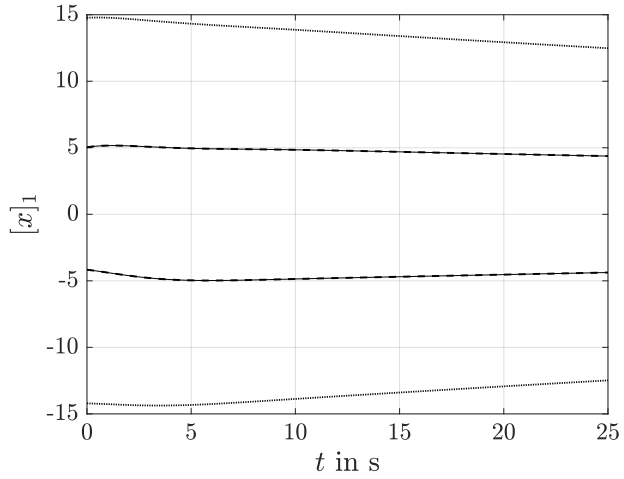


(d) Interval diameters of state x_2

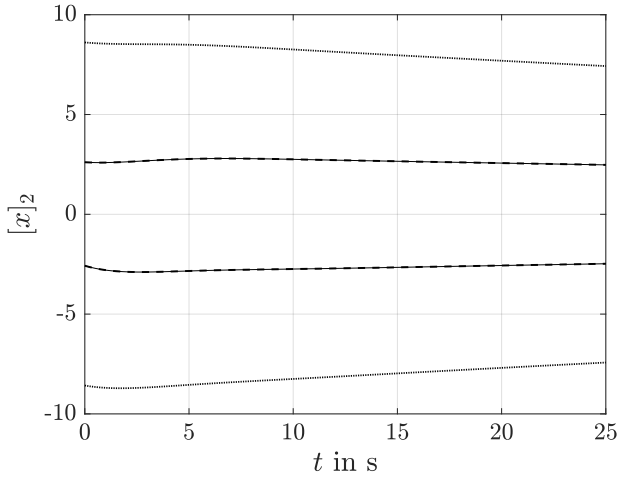


(e) Volume of the $n = 2$ interval box

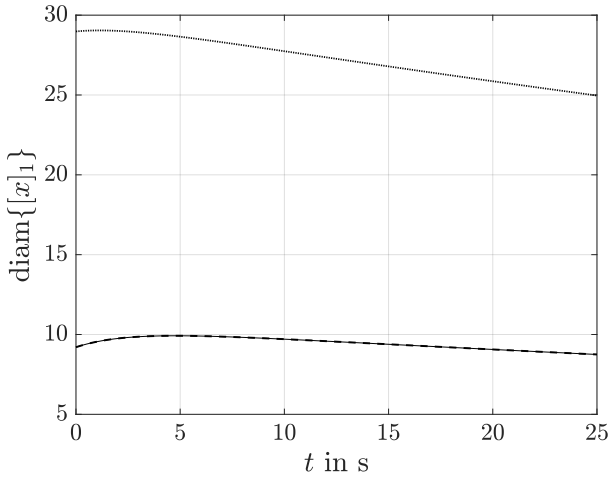
Figure 4.5: Comparison of three approaches — dependent treatment of parameters a_{21} and a_{22} — real-valued transformation:
1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right]\right]$ – dotted,
2. $[\mathbf{T}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{T}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.



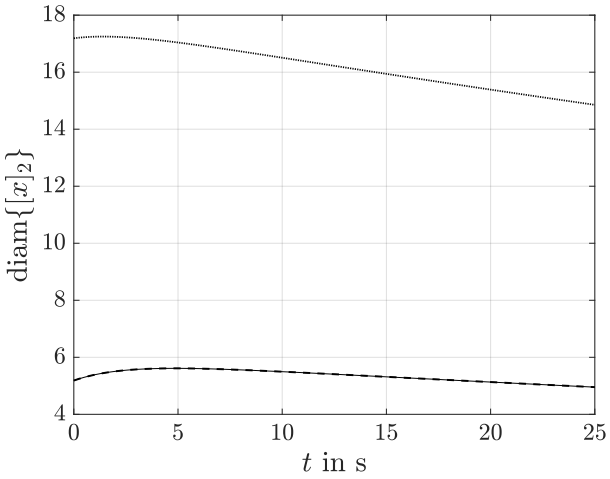
(a) Interval enclosures of state x_1



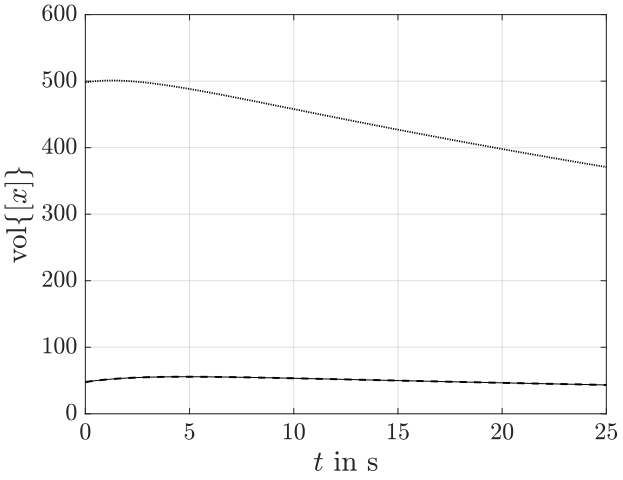
(b) Interval enclosures of state x_2



(c) Interval diameters of state x_1



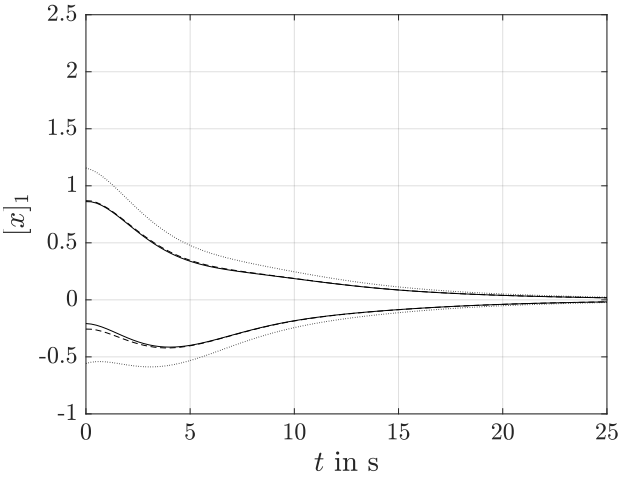
(d) Interval diameters of state x_2



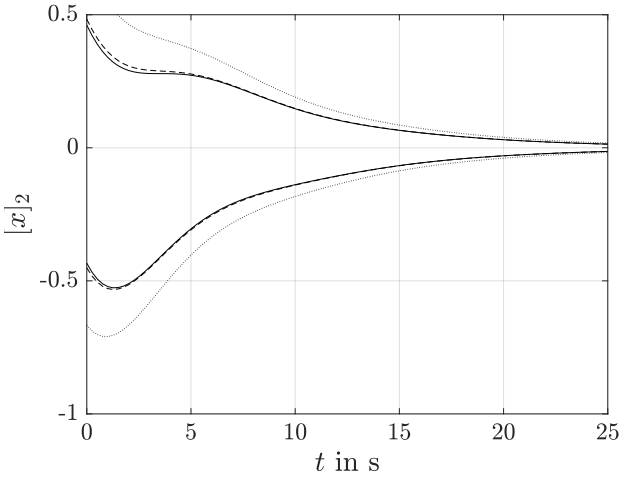
(e) Volume of the $n = 2$ interval box

Figure 4.6: Comparison of three approaches — independent treatment of parameters a_{21} and a_{22} —complex-valued transformation:

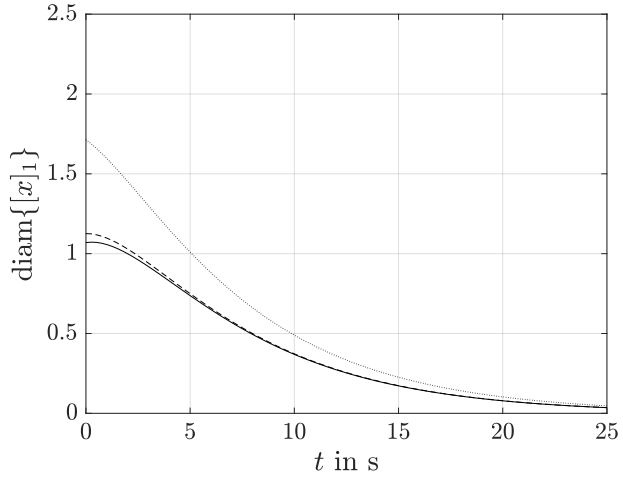
1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L}\left[\Re\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right], \bigcup_{\kappa=1}^{L}\left[\Im\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right]\right]$ – dotted,
2. $[\mathbf{T}] = \bigcup_{\kappa=1}^{L}\left[\mathbf{T}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L}\left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.



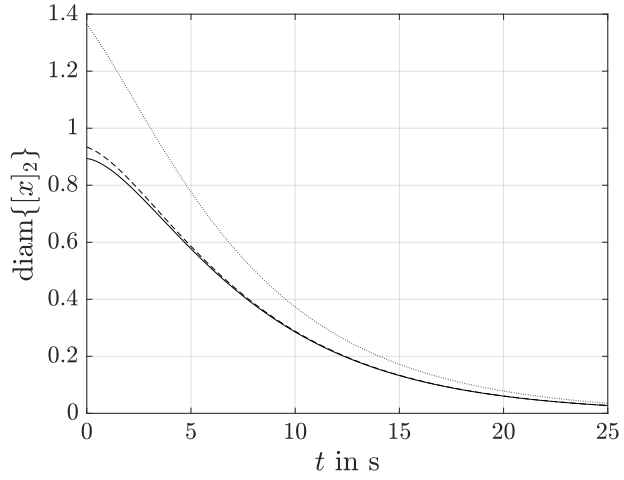
(a) Interval enclosures of state x_1



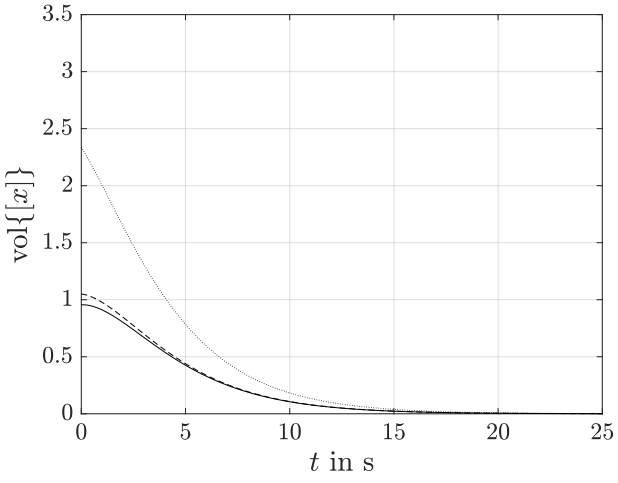
(b) Interval enclosures of state x_2



(c) Interval diameters of state x_1



(d) Interval diameters of state x_2



(e) Volume of the $n = 2$ interval box

Figure 4.7: Comparison of three approaches — dependent treatment of parameters a_{21} and a_{22} — complex-valued transformation:

1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]^{\langle\kappa\rangle}\}\right]\right]$ – dotted,
2. $[\mathbf{T}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{T}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.

4.5 Conclusion

In this chapter, possibilities were investigated to transform a non-cooperative dynamical system into a cooperative form. Since this is valid for autonomous systems, the controlled system matrix was taken into account. The resulting system matrix has either purely real eigenvalues, where a suitable point-valued transformation matrix can be found based on optimality criterions formulated by LMIs, or it includes complex-conjugate eigenvalues, where the cooperative form can be determined symbolically and the respective interval-valued transformation matrices are calculated accordingly. Additionally, approaches to reduce resulting overestimation in those matrices were presented using knowledge about the system. Finally, a complex-valued transformation was given as an alternative, if point-valued matrices are difficult to find for the system in question. However, for the presented numerical example, the complex-valued transformation resulted in much wider intervals, especially, when parameter dependencies were not considered.
Note that an additional feedforward control would have to be transformed separately. Furthermore, when constructing the final signal, positivity of the overall system must be ensured following the rules specified in Eqs. (1.5)–(1.7). In the next and final step, the system will be augmented by an observer as a fault diagnosis tool. Fig. 4.8 shows the progress so far in the overall approach in relation to Fig. 1.3.

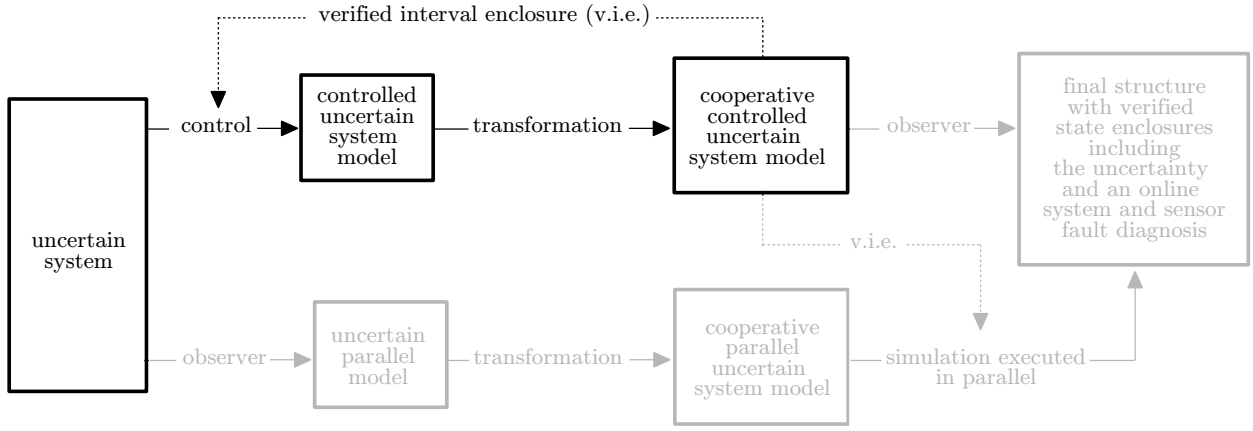


Figure 4.8: Outline of the theoretical aspects of this work - Step 2.

5 Observer Synthesis

States and disturbances, which are not measured whether because of physical or technical restrictions or because of cost factors, can be estimated by an observer. Additionally, in contrast to measured values, observed variables are less affected by noise and therefore sometimes preferable. Generally, when included, the system is augmented to[1]

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{H}(\mathbf{y}_\mathrm{m} - \mathbf{y}) \quad \text{with} \quad \mathbf{y} = \mathbf{C}_\mathrm{m}\hat{\mathbf{x}} \tag{5.1}$$

resulting in

$$\dot{\hat{\mathbf{x}}} = \Big(\mathbf{A} - \mathbf{H}\mathbf{C}_\mathrm{m}\Big)\hat{\mathbf{x}} + \mathbf{H}\mathbf{y}_\mathrm{m} \ , \tag{5.2}$$

where $\mathbf{C}_\mathrm{m}$ denotes the relation of the measured state variables with the system states in contrast to the output matrix $\mathbf{C}$, which refers to a general (controlled) output. This means, that adding state-of-the-art observers to a cooperative system may destroy its structure and, hence, its property of cooperativity.

5.1 Cooperativity-Preserving Observer

A possible solution to preserve cooperativity is to find an observer gain where[2]

$$\Big(\underline{\mathbf{A}} - \mathbf{H}\mathbf{C}_\mathrm{m}\Big) \cdot \hat{\mathbf{v}}(t) + \boldsymbol{\upsilon} = \dot{\hat{\mathbf{v}}}(t) \leq \dot{\hat{\mathbf{x}}}(t) \leq \dot{\hat{\mathbf{w}}}(t) = \Big(\overline{\mathbf{A}} - \mathbf{H}\mathbf{C}_\mathrm{m}\Big) \cdot \hat{\mathbf{w}}(t) + \boldsymbol{\omega} \tag{5.3}$$

holds with

$$\begin{aligned} \boldsymbol{\upsilon} &= \inf\left(\mathbf{H}\left[\mathbf{y}_\mathrm{m}\right]\right) \\ \boldsymbol{\omega} &= \sup\left(\mathbf{H}\left[\mathbf{y}_\mathrm{m}\right]\right) \end{aligned} \tag{5.4}$$

accounting for possible worst-case bounds of measurements resulting in an uncertain measurement vector $[\mathbf{y}_\mathrm{m}] = \mathbf{y}_\mathrm{m} + [-\Delta\mathbf{y}_\mathrm{m} \ ; \ \Delta\mathbf{y}_\mathrm{m}]$, cf. [50], and

$$\begin{aligned} \underline{\mathbf{A}} &= \inf\left(\mathbf{A}([\mathbf{p}])\right) \\ \overline{\mathbf{A}} &= \sup\left(\mathbf{A}([\mathbf{p}])\right) \ . \end{aligned} \tag{5.5}$$

In [48], three methods were presented, which are used in this work and shall be explained in this section. The first requirement, as usually, is the system matrix

$$\mathbf{A}_\mathrm{O}(\mathbf{p}) = \mathbf{A}(\mathbf{p}) - \mathbf{H}\mathbf{C}_\mathrm{m} = \mathbf{A}(\mathbf{p}) - \boldsymbol{\mathcal{H}} \tag{5.6}$$

to be asymptotically stable, checked analogously to Eq. (2.43) with

$$\mathbf{A}_\mathrm{O}(\mathbf{p}) \cdot \mathbf{P}_\mathrm{O} + \mathbf{P}_\mathrm{O} \cdot \mathbf{A}_\mathrm{O}^T(\mathbf{p}) \prec 0 \ , \tag{5.7}$$

[1] $\mathbf{B} \cdot \mathbf{u}$ is to be added for non-autonomous systems.

[2] Again, under the assumption of $\hat{\mathbf{v}} \geq \mathbf{0}$

and to be, furthermore, Metzler according to the conditions that all off-diagonal elements are non-negative for all possible parameter combinations. If $\mathbf{A}$ is Metzler and stable, an obvious choice is

$$\mathbf{H} = (\boldsymbol{\mathcal{K}}\boldsymbol{C}_{\mathrm{m}})^T \tag{5.8}$$

with

$$\boldsymbol{\mathcal{K}} = \mathrm{diag}\,\{\boldsymbol{\kappa}\} \tag{5.9}$$

and

$$\boldsymbol{\kappa} = \begin{bmatrix}\kappa_1 & \dots & \kappa_m\end{bmatrix}\ , \quad \text{where} \quad \kappa_i > 0,\ i \in \{1,\dots,m\}\ , \tag{5.10}$$

when $\mathbf{A}_{\mathrm{O}} \in \mathbb{R}^{n\times n}$ resulting in

$$\boldsymbol{\mathcal{H}} = \mathbf{H}\mathbf{C}_{\mathrm{m}} = \mathbf{C}_{\mathrm{m}}^T\boldsymbol{\mathcal{K}}\mathbf{C}_{\mathrm{m}} \tag{5.11}$$

with $\mathbf{C}_{\mathrm{m}} \in \mathbb{R}^{m\times n}$ and

$$\mathcal{H} = \begin{cases}\kappa_i & \text{for} \quad i = j \\ 0 & \text{else}\end{cases} \tag{5.12}$$

with $i \in \{1,\dots,m\}$ and $j \in \{1,\dots,n\}$ as the simplest possibility. However, the parameterization can be optimized according to the next section.

5.1.1 Optimized observer parameterization

Considering the difference between estimated and true values for both, lower and upper state bounds, the error vector

$$\mathbf{e} = \begin{bmatrix}\hat{\mathbf{v}} - \mathbf{v} \\ \hat{\mathbf{w}} - \mathbf{w}\end{bmatrix} \tag{5.13}$$

is obtained. The resulting ODEs

$$\dot{\mathbf{e}} = \begin{bmatrix}\underline{\mathbf{A}} - \mathbf{H}\mathbf{C} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{A}} - \mathbf{H}\mathbf{C}\end{bmatrix}\mathbf{e} + \begin{bmatrix}\mathbf{H} \\ \mathbf{H}\end{bmatrix}\boldsymbol{\zeta} \tag{5.14}$$

denote the estimation errors in dependence of a measurement tolerance vector $\boldsymbol{\zeta}$, where $\boldsymbol{\zeta} \in [-\Delta\mathbf{y}_{\mathrm{m}}\ ;\ \Delta\mathbf{y}_{\mathrm{m}}]$, see also [46]. To account for a comparison of those measurement errors $\boldsymbol{\zeta}$ and the weighted, with $\nu > 0$, state diameter $(\hat{\mathbf{w}} - \mathbf{w}) - (\hat{\mathbf{v}} - \mathbf{v})$, we make use of the augmented system output[3]

$$\mathbf{y}_\infty = \begin{bmatrix}\mathbf{0} & \mathbf{0} \\ -\nu\cdot\mathbf{I} & \nu\cdot\mathbf{I}\end{bmatrix}\mathbf{e} + \begin{bmatrix}\mathbf{I} \\ \mathbf{0}\end{bmatrix}\boldsymbol{\zeta} = \mathbf{C}_\infty\mathbf{e} + \mathbf{D}_\infty\boldsymbol{\zeta}\ . \tag{5.15}$$

Accordingly, an LMI-based optimization problem

$$\boldsymbol{\mathcal{L}}(\boldsymbol{\Theta}) := \begin{bmatrix}\boldsymbol{\Theta} & \mathbf{H} & \mathbf{P}_{\mathrm{O}}\mathbf{C}_\infty^T \\ \mathbf{H}^T & -\mathbf{I} & \mathbf{D}_{\infty,1}^T \\ \mathbf{C}_\infty\mathbf{P}_{\mathrm{O}} & \mathbf{D}_{\infty,1} & -\gamma_\infty^2\mathbf{I}\end{bmatrix} \prec 0 \tag{5.16}$$

is formulated for both extremal systems $\boldsymbol{\Theta} \in \{\underline{\boldsymbol{\Theta}}, \overline{\boldsymbol{\Theta}}\}$, where

$$\begin{aligned}\underline{\boldsymbol{\Theta}} &:= \underline{\mathbf{A}}_{\mathrm{O}}\cdot\mathbf{P}_{\mathrm{O}} + \mathbf{P}_{\mathrm{O}}\cdot\underline{\mathbf{A}}_{\mathrm{O}}^T \\ \overline{\boldsymbol{\Theta}} &:= \overline{\mathbf{A}}_{\mathrm{O}}\cdot\mathbf{P}_{\mathrm{O}} + \mathbf{P}_{\mathrm{O}}\cdot\overline{\mathbf{A}}_{\mathrm{O}}^T\end{aligned} \tag{5.17}$$

[3] inspired by the H_∞-norm

with

$$\begin{aligned}\underline{\mathbf{A}}_{\mathrm{O}} &= \inf\left(\mathbf{A}_{\mathrm{O}}([\mathbf{p}])\right)\\ \overline{\mathbf{A}}_{\mathrm{O}} &= \sup\left(\mathbf{A}_{\mathrm{O}}([\mathbf{p}])\right)\ .\end{aligned} \tag{5.18}$$

Here, a suitable Lyapunov function candidate is denoted by a joint — valid for both $\underline{\boldsymbol{\Theta}}$ and $\overline{\boldsymbol{\Theta}}$ — weighting matrix $\mathbf{P}_{\mathrm{O}} = \mathbf{P}_{\mathrm{O}}^T \succ 0$. Analogously to the control approach, a linearizing change of variables

$$\mathbf{Q}_{\mathrm{O}} = \mathbf{Q}_{\mathrm{O}}^T = \mathbf{P}_{\mathrm{O}}^{-1} \succ 0 \quad \text{with} \quad \mathbf{Y}_{\mathrm{O}}^T = \mathbf{Q}_{\mathrm{O}}\mathbf{H} = \mathbf{P}_{\mathrm{O}}^{-1}\mathbf{H} \tag{5.19}$$

rearranges Eq. (5.16) into the design LMI

$$\boldsymbol{\mathcal{M}}(\boldsymbol{\Sigma}) := \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{Y}_{\mathrm{O}}^T & \mathbf{C}_{\infty}^T \\ \mathbf{Y}_{\mathrm{O}} & -\mathbf{I} & \mathbf{D}_{\infty,1}^T \\ \mathbf{C}_{\infty} & \mathbf{D}_{\infty,1} & -\mu_{\infty}\mathbf{I} \end{bmatrix} \prec 0 \tag{5.20}$$

for both $\boldsymbol{\Sigma} \in \left\{\underline{\boldsymbol{\Sigma}}, \overline{\boldsymbol{\Sigma}}\right\}$,

$$\begin{aligned}\underline{\boldsymbol{\Sigma}} &:= \mathbf{Q}_{\mathrm{O}}\underline{\mathbf{A}} - \mathbf{Y}_{\mathrm{O}}^T\mathbf{C} + \underline{\mathbf{A}}^T\mathbf{Q}_{\mathrm{O}} - \mathbf{C}^T\mathbf{Y}_{\mathrm{O}}\\ \overline{\boldsymbol{\Sigma}} &:= \mathbf{Q}_{\mathrm{O}}\overline{\mathbf{A}} - \mathbf{Y}_{\mathrm{O}}^T\mathbf{C} + \overline{\mathbf{A}}^T\mathbf{Q}_{\mathrm{O}} - \mathbf{C}^T\mathbf{Y}_{\mathrm{O}}\end{aligned} \tag{5.21}$$

with

$$\mu_{\infty} := \gamma_{\infty}^2 \geq 0\ , \tag{5.22}$$

which can be used for the observer design taking into account all former considerations and, in addition, minimizes μ_{∞}. To ensure cooperativity of the estimation error dynamics, the element-wise defined inequality constraint

$$\mathrm{col}\Big((\underline{\mathbf{A}} - \check{\mathbf{Q}}_{\mathrm{O}}^{-1})\mathbf{Y}_{\mathrm{O}}^T\mathbf{C} \circ (\mathbf{E} - \mathbf{I})\Big) \geq \mathbf{0} \tag{5.23}$$

needs to be satisfied with $\circ$ indicating the Hadamard product, see Appendix A.4. Note that Eq. (5.23) has to be solved iteratively due to the nonlinearity.

Applying Eqs. (5.20)–(5.23), the observer matrix $\mathbf{H}$ can be fully determined. However, a simplified version is achieved, when taking into account the considerations of Eqs. (5.8)–(5.10). The LMIs (5.20) as well as (5.21) turn into

$$\boldsymbol{\mathcal{N}}(\boldsymbol{\Xi}) := \begin{bmatrix} \boldsymbol{\Xi} & \check{\mathbf{Q}}_{\mathrm{O}} \cdot (\boldsymbol{\mathcal{K}}\boldsymbol{C})^T & \mathbf{C}_{\infty}^T \\ (\boldsymbol{\mathcal{K}}\boldsymbol{C}) \cdot \check{\mathbf{Q}}_{\mathrm{O}} & -\mathbf{I} & \mathbf{D}_{\infty,1}^T \\ \mathbf{C}_{\infty} & \mathbf{D}_{\infty,1} & -\mu_{\infty}\mathbf{I} \end{bmatrix} \prec 0 \tag{5.24}$$

with $\boldsymbol{\Xi} \in \left\{\underline{\boldsymbol{\Xi}}, \overline{\boldsymbol{\Xi}}\right\}$ according to

$$\begin{aligned}\underline{\boldsymbol{\Xi}} &:= \mathbf{Q}_{\mathrm{O}}\underline{\mathbf{A}} - \check{\mathbf{Q}}_{\mathrm{O}}\mathbf{C}^T\boldsymbol{\mathcal{K}}\boldsymbol{C} + \underline{\mathbf{A}}^T\mathbf{Q}_{\mathrm{O}} - \mathbf{C}^T\boldsymbol{\mathcal{K}}\boldsymbol{C}\check{\mathbf{Q}}_{\mathrm{O}}\\ \overline{\boldsymbol{\Xi}} &:= \mathbf{Q}_{\mathrm{O}}\overline{\mathbf{A}} - \check{\mathbf{Q}}_{\mathrm{O}}\mathbf{C}^T\boldsymbol{\mathcal{K}}\boldsymbol{C} + \overline{\mathbf{A}}^T\mathbf{Q}_{\mathrm{O}} - \mathbf{C}^T\boldsymbol{\mathcal{K}}\boldsymbol{C}\check{\mathbf{Q}}_{\mathrm{O}}\end{aligned} \tag{5.25}$$

with $\check{\mathbf{Q}}_{\mathrm{O}}$ representing the matrix $\mathbf{Q}_{\mathrm{O}}$ computed in the previous solution stage to regain linearity in the LMI. To summarize, this leads to three possible observer parameterizations according to

1. The observer gain $\mathbf{H}$ is determined by Eqs. (5.20) with (5.21), which minimizes the H_{∞} criterion (5.22) for the augmented output (5.15). Ensuring the Metzler property, constraint (5.23) is introduced, where $\check{\mathbf{Q}}_{\mathrm{O}}$ is used to represent the matrix $\mathbf{Q}_{\mathrm{O}}$ computed in the previous solution stage. Here, an identity matrix with the respective dimensions is used to initialize $\mathbf{Q}_{\mathrm{O}}$ in the first step.

2. Find optimal gain values $\kappa_i > 0$ for the structure presented in Eqs. (5.8)–(5.10) as solution of the LMIs in (5.24) with (5.25) which also minimizes the H_∞ criterion (5.22) for the augmented output (5.15). However, we do not need to take into account the inequality constraint in (5.23), because the pre-defined structure of $\mathbf{H} = (\boldsymbol{\mathcal{K}}\boldsymbol{C}_\mathrm{m})^T$ is already Metzler. To ensure that (5.25) is linear despite multiplicative coupling of $\mathbf{Q}_\mathrm{O}$ and $\boldsymbol{\mathcal{K}}$, the solution is again determined iteratively.
3. Find optimal gain values $\kappa_1 = \ldots = \kappa_m > 0$ using the same approach as in the previous parameterization (2.).

Which observer approach is best, needs to be determined for each application scenario individually without predefinable indicators. However, due to their similar implementation, they can be switched against one another quite easily. As stated above, this approach does not include anything else as point-valued entries in the matrix $\mathbf{C}$, which means that this is not possible to apply directly to systems which where either transformed by interval-valued matrices or by complex-valued ones. However, an inclusion into the LMIs would be possible, but presumably less efficient than the following alternative.

5.2 Alternative Solution: LMI-based Observer

If the cooperativity-preserving approach is not applicable, i.e., in case of conjugate-complex eigenvalues, which lead to interval-valued or complex-valued transformation matrices according to Sec. 4.3, an alternative solution presents itself in form of an LMI-based observer similar to the control approach. Here, the idea is to not only calculate the controller gains but also a suitable observer before a transformation is applied.
Applying the duality principle on the presented control approach, (3.24) becomes

$$\mathbf{D}_0 \otimes \mathbf{Q}_\mathrm{O} + \mathbf{D}_1 \otimes (\mathbf{Q}_\mathrm{O}\mathbf{A}_\nu - \mathbf{Y}_\mathrm{O}^T\mathbf{C}) + \mathbf{D}_1^T \otimes (\mathbf{A}_\nu^T\mathbf{Q}_\mathrm{O} - \mathbf{C}^T\mathbf{Y}_\mathrm{O}) \prec 0 \tag{5.26}$$

with a linearizing change of variables, similar to (3.10), of

$$\begin{aligned} &\mathbf{P}_\mathrm{O} = \mathbf{Q}_\mathrm{O}^{-1} \ , \quad \mathbf{H}^T = \mathbf{Y}_\mathrm{O}\mathbf{P}_\mathrm{O} \ , \\ &\mathbf{Q}_\mathrm{O} = \mathbf{P}_\mathrm{O}^{-1} \ , \quad \mathbf{Y}_\mathrm{O} = \mathbf{H}^T\mathbf{P}_\mathrm{O}^{-1} \quad \text{as well as} \quad \mathbf{Y}_\mathrm{O}^T = \mathbf{P}_\mathrm{O}^{-1}\mathbf{H} \ , \end{aligned} \tag{5.27}$$

accordingly. There is, furthermore, the possibility to add the H_∞-norm, analogously to the approach in Chapter 3, by including

$$\begin{bmatrix} \mathbf{Q}_\mathrm{O}\mathbf{A} - \mathbf{Y}_\mathrm{O}^T\mathbf{C} + \mathbf{A}^T\mathbf{Q}_\mathrm{O} - \mathbf{C}^T\mathbf{Y}_\mathrm{O} & \mathbf{Y}_\mathrm{O}^T & \mathbf{C}_\infty^T \\ \mathbf{Y}_\mathrm{O} & -\mathbf{I} & \mathbf{D}_{\infty 1}^T \\ \mathbf{C}_\infty & \mathbf{D}_{\infty 1} & -\gamma_\infty^2\mathbf{I} \end{bmatrix} \prec 0 \tag{5.28}$$

with

$$\mathbf{C}_\infty = \begin{bmatrix} \mathbf{0} \\ \rho\mathbf{I} \end{bmatrix} \quad \text{and} \quad \mathbf{D}_{\infty 1} = \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \ , \tag{5.29}$$

where ρ is a weighting factor and γ_∞ is minimized.

5.3 Conclusion

This chapter completes the overall theoretical approach of Fig. 1.3 by presenting observer strategies to be used as a fault diagnosis tool operating on the closed-loop system model either for sensors or for the system itself.

Firstly, a method to add an observer while preserving cooperativity was presented complying to the structure in Fig. 5.1. Here, three different methods were considered, working with LMIs to find optimal solutions for the observer gain **H**. There are cases, however, where such a cooperativity-preserving observer cannot be applied e.g. due to an inefficient increase of complexity by a complex valued output matrix **C**.

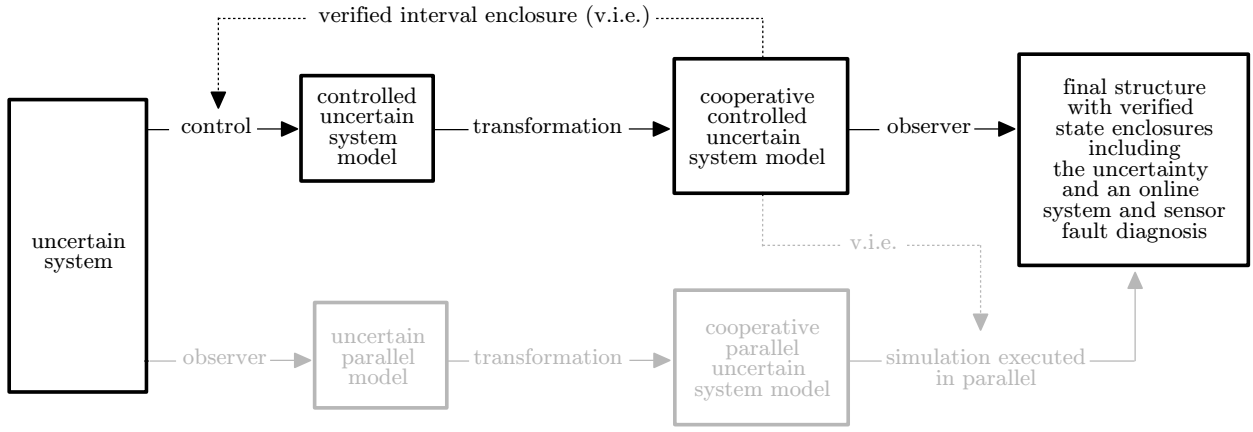


Figure 5.1: Outline of the theoretical aspects of this work - Path 1.

For such systems, an additional solution was given complying to the structure given in Fig. 5.2. Based on the duality principle, those systems are treated analogously to the control design, meaning that the observer is calculated to find a suitable parallel model, which is then transformed. A subsequent parallel simulation of both the controlled cooperative system and the cooperative parallel model presents comparable outputs which can be used for a diagnosis.

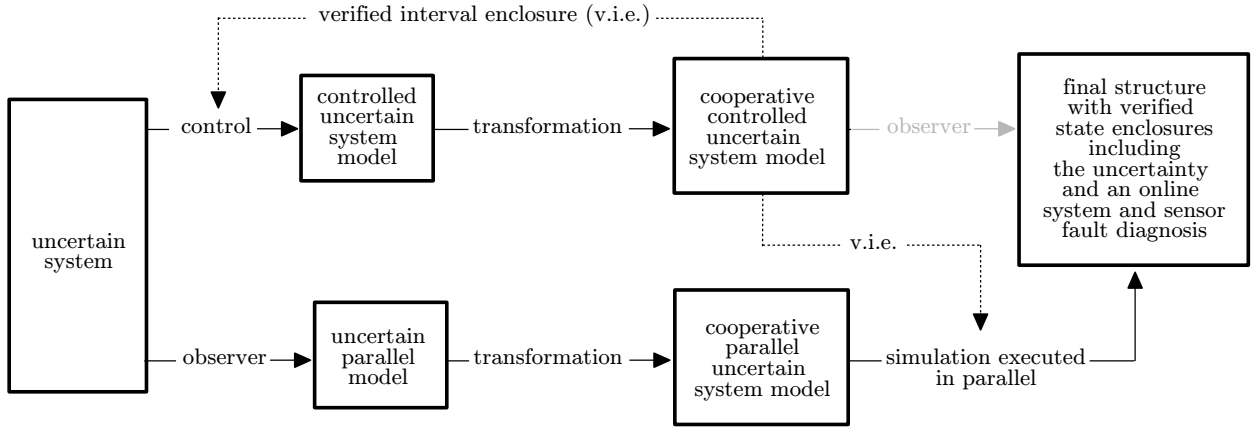


Figure 5.2: Outline of the theoretical aspects of this work - Path 2.

However, this approach often means larger overestimation due to the two-sided transformation concerning both the controlled as well as the parallel system, see Chapter 7, while in the case of a cooperativity-preserving structure one transformation step can be omitted. Obviously, this step should especially be avoided when dealing with originally cooperative systems derived by first principle modeling. It is to be seen as a last resort if a cooperativity-preserving observer cannot be found, e.g. due to infeasible LMI problems.

Part II

Application Scenarios

6 Electrical Circuit - An Example for Systems with Purely Real Eigenvalues

As an example for applications with purely real eigenvalues, electrical circuits are considered in this chapter. Note that to fulfill this eigenvalue requirement, parameters are chosen accordingly. In a first scenario, a simple RLC network is subject to treatment with the presented methods, including an LMI-based control design, a transformation into cooperativity and a cooperativity-preserving observer. The latter can be used as a tool for fault diagnosis of the measurements used in the control. Afterwards, this network will be extended to analyze the behavior with time varying parameter uncertainties. Finally, a down-step converter is considered based on the findings of the simple RLC network.

6.1 A Simple RLC Network

At first, the system's dynamical behavior is described by a derivation of the mathematical modeling. It is based on the model in Fig. 6.1.

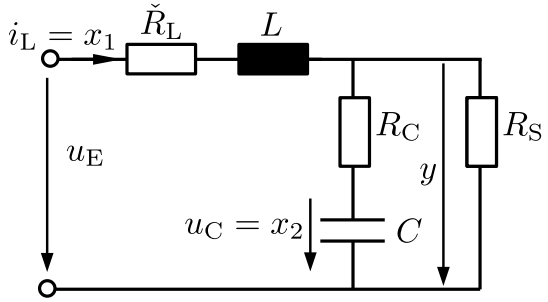


Figure 6.1: Simplified model.

A variable load is summed up to $R_\mathrm{S} = \tilde{R}_\mathrm{S} + \Delta R_\mathrm{S}$, where ΔR_S is implemented as a series connection of various resistances that can be activated and deactivated by semi-conductive switches. Additionally, $\check{R}_\mathrm{L} = R_0 + R_\mathrm{L}$ combines a limiting resistance R_0 and the inner resistance R_L of the real inductivity. There are two voltage loops, which are described by

$$u_\mathrm{E} = u_{\check{R}_\mathrm{L}} + u_\mathrm{L} + u_{R_\mathrm{C}} + u_\mathrm{C} \tag{6.1}$$

and

$$u_\mathrm{C} + u_{R_\mathrm{C}} = u_{R_\mathrm{S}} \tag{6.2}$$

as well as Kirchhoff's node equation with

$$i_{R_\mathrm{S}} + i_\mathrm{C} = i_\mathrm{L} \ . \tag{6.3}$$

The component equations for all Ohmic resistances are represented by

$$u_{R_i} = R_i \cdot i_{R_i} \tag{6.4}$$

with $i \in \{\mathrm{L}, \mathrm{S}, \mathrm{C}\}$, while the inductivity is given by

$$u_\mathrm{L} = L \cdot \frac{\mathrm{d}}{\mathrm{d}t} i_\mathrm{L} \tag{6.5}$$

and the capacity by

$$i_\mathrm{C} = C \cdot \frac{\mathrm{d}}{\mathrm{d}t} u_\mathrm{C} \ . \tag{6.6}$$

The differential equations for the physical storage variables result in

$$\frac{\mathrm{d}}{\mathrm{d}t} i_\mathrm{L} = \frac{1}{L} \left[- \left(\check{R}_\mathrm{L} + \frac{R_\mathrm{S} R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} \right) \cdot i_\mathrm{L} - \left(1 - \frac{R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} \right) \cdot u_\mathrm{C} + u_\mathrm{E} \right] \tag{6.7}$$

as a relation for variations of the magnetic field energy and in

$$\frac{\mathrm{d}}{\mathrm{d}t} u_\mathrm{C} = \frac{R_\mathrm{S}}{C \left(R_\mathrm{S} + R_\mathrm{C} \right)} \cdot i_\mathrm{L} - \frac{1}{C \left(R_\mathrm{S} + R_\mathrm{C} \right)} \cdot u_\mathrm{C} \tag{6.8}$$

for changes of the electric field energy. Those physical storage variables serve as the system's states, so that $x_1 = i_\mathrm{L}$ is represented by the current and $x_2 = u_\mathrm{C}$ by the voltage. The resulting state-space representation is given by

$$\dot{\mathbf{x}} = \begin{bmatrix} -\frac{1}{L} \left(\check{R}_\mathrm{L} + \frac{R_\mathrm{S} R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} \right) & \frac{1}{L} \left(\frac{R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} - 1 \right) \\ \frac{R_\mathrm{S}}{C \left(R_\mathrm{S} + R_\mathrm{C} \right)} & -\frac{1}{C \left(R_\mathrm{S} + R_\mathrm{C} \right)} \end{bmatrix} \cdot \mathbf{x} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \cdot u_\mathrm{E} \tag{6.9}$$

with the output equation

$$y = u_{R_\mathrm{C}} + u_\mathrm{C} = \begin{bmatrix} \frac{R_\mathrm{S} R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} & 1 - \frac{R_\mathrm{C}}{R_\mathrm{S} + R_\mathrm{C}} \end{bmatrix} \cdot \mathbf{x} \tag{6.10}$$

according to Fig. 6.1. The required parameters are given in Table 6.1. Note that the system is

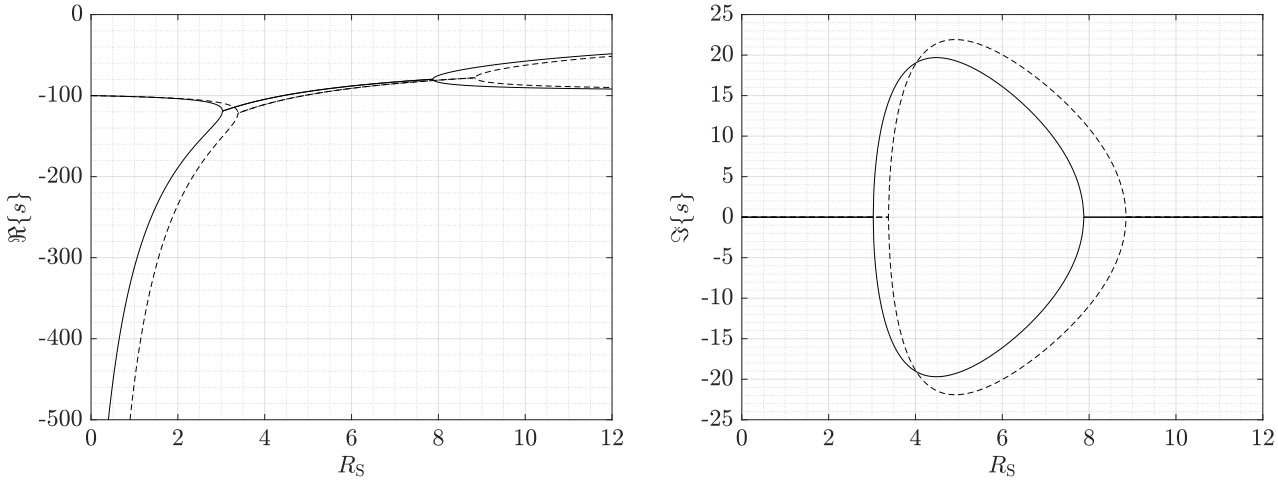
Table 6.1: Parameters of the low-power electrical circuit.

Variable	Unit/Value	Meaning
L	1 H	Inductivity[1]
C	2 mF	Capacity
R_S	[0.1 ; 3] Ω	Ohmic resistance of the load[2]
R_C	[0.1 ; 0.6] Ω	Parasitic Ohmic resistance of the capacity[2]
R_L	100 Ω	summed up Ohmic resistance of the inner resistance of the inductivity and a limiting resistance

designed to provide an electrical load resistance up to 12 Ω. However, since we want to restrict ourselves to a system with purely real eigenvalues, the locations of the eigenvalues are evaluated in dependence of the electrical load resistance R_S. Fig. 6.2 shows the result with real parts on the left in 6.2(a) and imaginary parts on the right in 6.2(b).

[1] To apply such large inductance values in a small-power network, it is implemented with the help of a gyrator circuit that turns capacitors into virtual inductors with non-zero internal resistances [63], see C.1.

[2] These values are given in interval representation, because of uncertainty due to tolerances and variability within the range.



(a) Real parts of the eigenvalues

(b) Imaginary parts of the eigenvalues

Figure 6.2: Change of eigenvalues according to the load resistance: The case of $R_C = 0.1\,\Omega$ is depicted with a dashed line and $R_C = 0.6\,\Omega$ with a solid line.

One can clearly see a change of the system's behavior due to the dependence on the load resistance. These changes are called bifurcation points, where — in this case — the two real eigenvalues become a conjugate-complex pair only to change again into two distinct purely real eigenvalues. This change of system behavior can also occur in nonlinear systems. To show that this point may vary with variable parameters, the system was evaluated for $R_C = 0.1\,\Omega$ with a dashed line and $R_C = 0.6\,\Omega$ with a solid line. To guarantee purely real eigenvalues for the given example, the load will be restricted to $R_S = [0.1\ ;\ 3]\ \Omega$, i.e., to resistances smaller than the left bifurcation point.

6.1.1 Robust state-feedback control

A control design is now determined with the approach of Chapter 3 for the system (6.9) with the parameters of Table 6.1. For this linear system, the interval box is given by the two independent parameters R_S and R_C and, hence, the approach of an axis-aligned box, see Fig. 3.1, is applied. To force the eigenvalues to stay purely real, a sufficiently large damping ratio, see Fig. 2.5, is introduced as described in Sec. 2.5.3. For this, the matrices in Eq. (2.46) are set to

$$\mathbf{D}_0 = \mathbf{0} \quad \text{and} \quad \mathbf{D}_1 = \begin{bmatrix} \sin(\theta) & \cos(\theta) \\ -\cos(\theta) & \sin(\theta) \end{bmatrix} \tag{6.11}$$

with a minimization of θ up to $\theta = \dfrac{\pi}{8}$ subsequently showing that the resulting eigenvalues are purely real. This leads to

$$\mathbf{k}^T = \begin{bmatrix} 1.211 & 0.291 \end{bmatrix} \cdot 10^4 \tag{6.12}$$

as a solution for the controller gains resulting in

$$[\mathbf{A}]_C = \begin{bmatrix} [-1.223\ ;\ -1.221] & [-0.291\ ;\ -0.290] \\ [0.001\ ;\ 0.751] & [-0.251\ ;\ -0.013] \end{bmatrix} \cdot 10^4 \tag{6.13}$$

as the numerical representation of the controlled system matrix. The eigenvalue locations are given in Fig. 6.3.

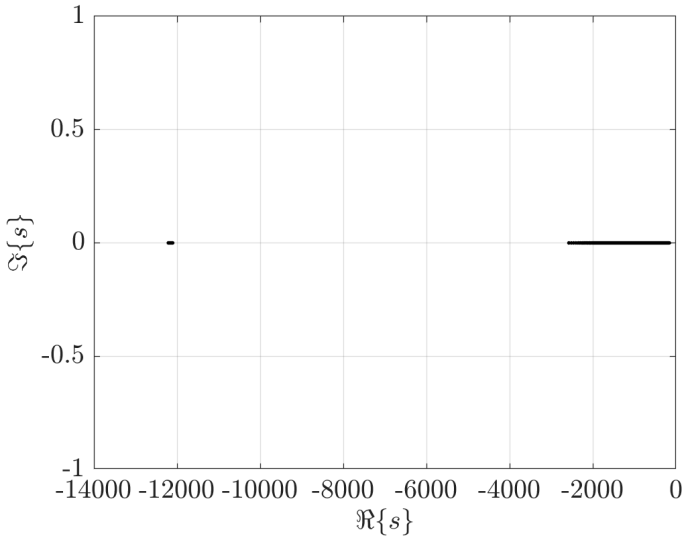


Figure 6.3: Distribution of eigenvalues for the controlled system.

6.1.2 Transformation

Since the system matrix (6.13) is not Metzler, but all eigenvalues are purely real, the transformation according to Sec. 4.2 is applied. Note that an internal random initialization of the algorithm to solve the respective LMIs can lead to different results if executed multiple times. Therefore, to find the optimal solution, the transformation algorithm is repeated as long as a tighter interval can be found with the same controller gain. Due to the non-diagonally dominant form for the presented case, the resulting transformation matrix is determined by means of the two-stage approach in Sec. 4.2 according to

$$\boldsymbol{\Theta} = \mathbf{VS} = \begin{bmatrix} 0.7768 & -0.5841 \\ -0.9680 & -0.2714 \end{bmatrix} , \tag{6.14}$$

cf. Eq. (4.34). With that, the system in cooperative form becomes

$$\tilde{\mathbf{A}}_{\mathrm{C}} \in \begin{bmatrix} [-0.428\ ;\ -0.260] & [0.218\ ;\ 0.286] \\ [0.540\ ;\ 0.763] & [-1.015\ ;\ -0.925] \end{bmatrix} \cdot 10^4 \ . \tag{6.15}$$

The simulations are done for a time horizon of 30 ms with the initial states

$$\mathbf{x}(0) \in \begin{bmatrix} [0\ ;\ 5 \cdot 10^{-3}] \\ [0\ ;\ 5] \end{bmatrix} \ . \tag{6.16}$$

Table 6.2 gives the calculated hulls of the interval enclosures for both states.

Table 6.2: Hulls of interval enclosures over the whole time horizon of 30 ms.

step	$\underline{i}_{\mathrm{L}}$ in A	$\overline{i}_{\mathrm{L}}$ in A	$\underline{u}_{\mathrm{C}}$ in V	$\overline{u}_{\mathrm{C}}$ in V
1	$-\underline{2.879}897$	$\underline{2.884}897$	$-\underline{0.001}743$	$\underline{5.001}743$
2	-2.879542	2.884542	-0.001731	5.001731

The algorithm is implemented to repeat the calculation at least 10 times. If the calculated intervals are tighter than the ones before, they are added to the list as a new optimum. Hence, the table shows only two successful steps as the reasoning that the optimum was found. Since no further reduction occurred after the second step, the optimal solution was found and the algorithm was aborted and further executions were canceled.

6.1.3 Numerical results

For a reliable classification of the presented method using cooperative systems, it is compared with a state-of-the-art Taylor series expansion [23, 39] of order 2 without preconditioning the state equations by a QR factorization or similar method.

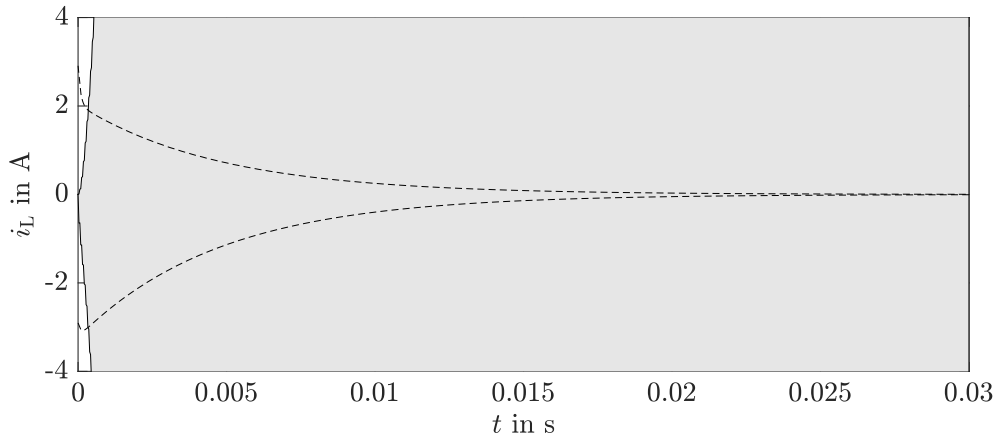


Figure 6.4: Comparison of the presented method relying on cooperativity in a dashed line and a state-of-the-art Taylor series prediction in gray for the state variable i_L.

Fig. 6.4 shows the predicted progress of the first state i_L. Due to the transformation, there is still overestimation when applying the method based on cooperativity, although both bounds reach the desired operating point in under 20 ms. In contrast to that, the Taylor series expansion fails fastly to predict reasonable states. Note that the Picard iteration included in this method is based on a discrete Euler procedure as presented in Sec. 2.2. This means that discretization errors are also added to the occurring wrapping effect. The chosen method, here, are 300 steps to discretize the time horizon of 30 ms, resulting in a discretization time of $T = 0.1$ ms. For fast dynamics, this can range from being difficult for an online prediction of state intervals up to sometimes becoming impossible even for an offline prediction.

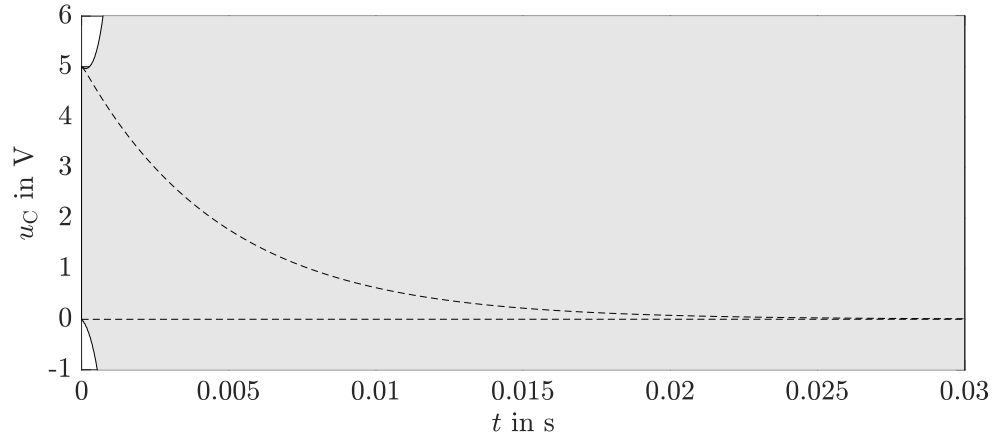


Figure 6.5: Comparison of the presented method relying on cooperativity in a dashed line and a state-of-the-art Taylor series prediction in gray for the state variable u_C.

As one can see in this case, the system's fast dynamic, see also the eigenvalue distribution in Fig. 6.3, renders an offline prediction by the simple Taylor series-based method impossible due to overestimation. Mathematically, the overestimation can slightly be improved by choosing a faster discretization time of, e.g. $1\,\mu$s, which would result in a reduced discretization error

of the Euler procedure and, hence, influence the overall approximation. However, this only underlines the fact that this is even more impractical for real life applications. A possible advantage could be found when intersecting both solutions. As the Taylor approximation has slightly tighter results for the bounds of the starting phase, an intersection may lead to an improvement regarding the tight enclosure of the state. This does not hold for the second state u_{C}, which can be found in Fig. 6.5, because here, the cooperativity-based method is not subject to high overestimation in the initial interval and, hence, is not wider than the Taylor-based approach. Figs. 6.6 and 6.7 show the respective states in their transformed coordinates, where one can clearly see cooperativity and positivity of the system.

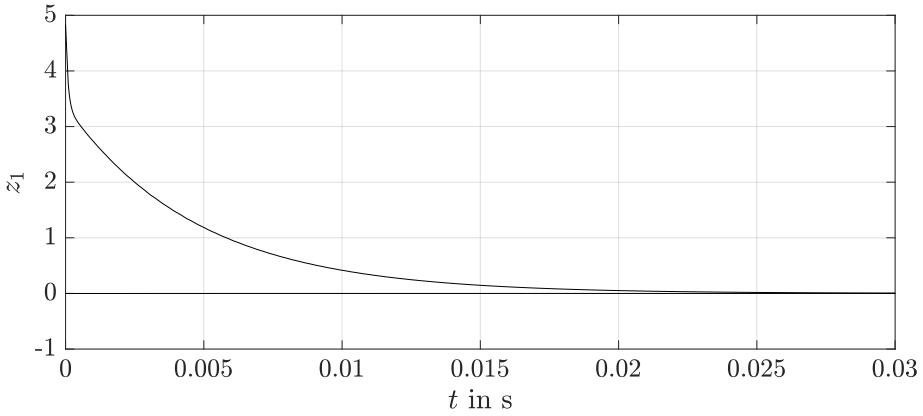


Figure 6.6: Prediction of the state variable z_1 in transformed coordinates.

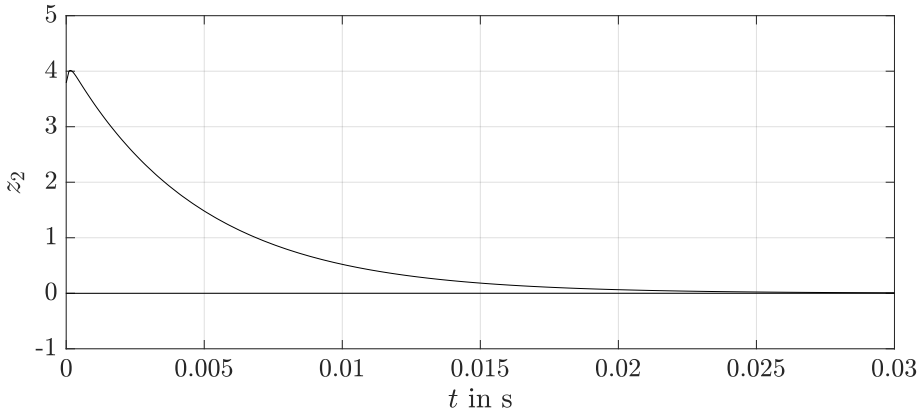


Figure 6.7: Prediction of the state variable z_2 in transformed coordinates.

6.1.4 Observer

So far, the measurements of the states are assumed to be directly available for the controller. As mentioned before, an observer is added to the system as a fault diagnosis tool to evaluate those measurements. Additional applications may be the estimation of the load resistance as well as the deduction of component failures. In the present case, the controller acts on the basis of state measurements alone and the observer is purely instated as a separate system diagnozing the closed-loop system dynamics. Since adding a state-of-the-art observer would destroy the cooperative structure, the cooperativity-preserving observer according to Chapter 5 with an optimized parameterization of case 2 from Sec. 5.1.1 is added. The resulting system matrix is obviously still Metzler due to the fact that the structure of the observer gain vector $\mathbf{h}$ is

predefined. Note that the output vector $\mathbf{c}^T$ of Eq. (6.10) also needs to be transformed into the new coordinates, resulting in

$$\mathbf{h}\tilde{\mathbf{c}}^T = \begin{bmatrix} 0.0389 & 0.1372 \\ 0.1372 & 0.4849 \end{bmatrix} \tag{6.17}$$

with $\tilde{\mathbf{c}}^T = \mathbf{c}^T(\mathbf{VS})^{-1}$, to calculate the observed system $\tilde{\mathbf{A}}_\mathrm{O} = \tilde{\mathbf{A}}_\mathrm{C} - \mathbf{h}\tilde{\mathbf{c}}^T$. Fig. 6.8 shows a possible application of this observer.

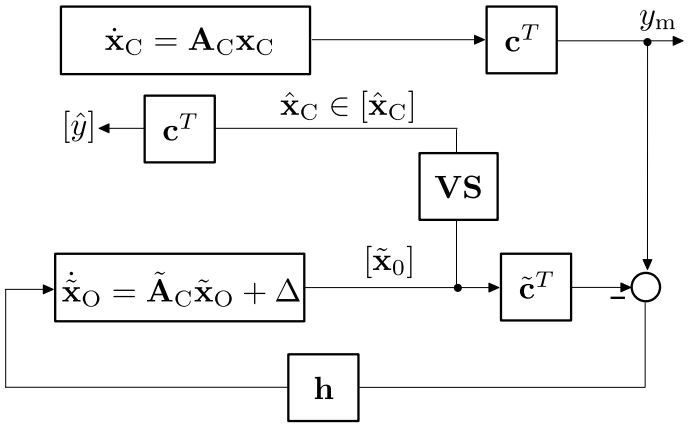


Figure 6.8: Use of the cooperativity-preserving observer as a fault diagnosis tool.

Here, two usages are possible. Firstly, a system fault diagnosis could be given when checking if $\mathbf{x}_\mathrm{C} \in [\hat{\mathbf{x}}_\mathrm{C}]$. Secondly, a sensor fault is detected by comparing the output of the observer $[\hat{y}]$ with the measured output of the system y_m.

6.2 Adding a Time-Varying Parameter: Power-Dependent Load Resistance Variation

As already mentioned, the system is equally solvable when the parameter changes over time, because those variations are already included in the prior example. However, if the corresponding values of this parameter as well as their respective time instants are known, tighter enclosures may be reached with the application of gain adaptation designed controllers similar to the approach presented in Sec. 3.3.2. For this, a power-dependent load resistance variation is implemented. It is based on the idea that with a constant input a certain power level can be held. This complies to reality, where the consumer defines the amount of power needed. To implement this, one must determine the necessary resistance R_S for the desired power P_d. To calculate the respective desired load resistance, a simulation was done according to Fig. 6.9.

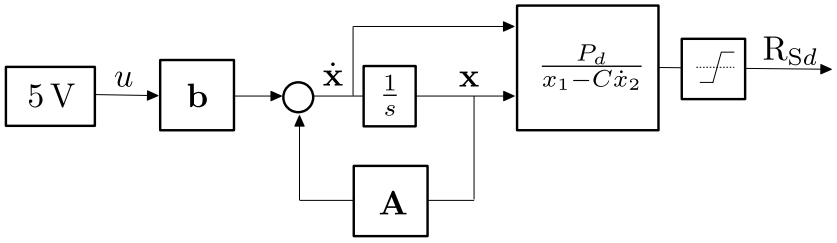


Figure 6.9: Determination of the desired load resistance $R_{\mathrm{S}d}$.

Here, the constant voltage input is set to 5 V and $\mathbf{A}$ and $\mathbf{b}$ are point-valued. The corresponding states are computed by following Eq. (6.9). Those states can now be used to calculate the desired

load resistance $R_{\mathrm{S}d}$ by reformulating

$$P_d = u_{R_\mathrm{S}} \cdot i_{R_\mathrm{S}} = R_\mathrm{S} \cdot i_{R_\mathrm{S}}^2 = R_\mathrm{S} \cdot (i_\mathrm{L} - C\dot{u}_\mathrm{C})^2 \tag{6.18}$$

into

$$R_{\mathrm{S}d} = \frac{P_d}{(i_\mathrm{L} - C\dot{u}_\mathrm{C})^2} \tag{6.19}$$

and taking into account Eq. (6.8). Since calculated intervals need to match the restrictions of the test rig in question, a saturation is introduced to keep $0.1\,\Omega < R_{\mathrm{S}d} < 2.9\,\Omega$. Additionally, only discretized values are realizable, which is why the parameter was rounded to the nearest possible value.

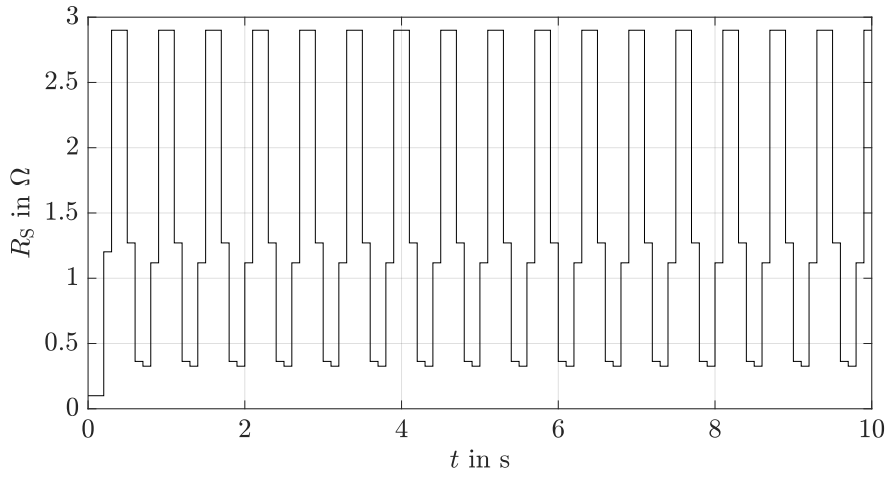


Figure 6.10: Resulting temporal variability of the desired load resistance $R_{\mathrm{S}d}$ over the complete time horizon of 10 s.

Fig. 6.10 shows the resulting time-dependent load resistance for a desired power of $P_d = 3\,\mathrm{mW}$ taking into account realizable steps of 100 ms on the test rig in question. The simulation was done over a time horizon of 10 s and, hence, including 100 steps. Note that the switching is purely time-dependent with crisp values for R_S for each time step, which leaves only R_C as an uncertain parameter.

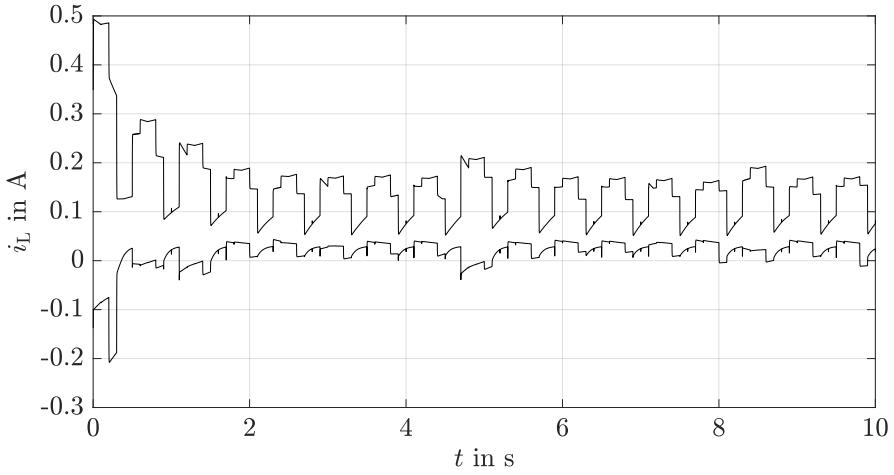


Figure 6.11: Predicted state enclosure for the first state i_L over the complete time horizon of 10 s including 100 variations of $R_{\mathrm{S}d}$.

With another simulation, predicted state enclosures shall now be computed based on the switching signal. This is done for each time step and its respective value for $R_{\mathrm{S}d}$ according to the idea presented in Sec. 3.3.2, where a gain scheduling control was designed for temporal subslices. Note, those slices are temporally predefined after establishing the sequence of the desired load resistance, which, hence, remains valid over the whole subslice. The results are given in Fig. 6.11 for the current i_{L} as the first state and in Fig. 6.12 for the second state u_{C}.

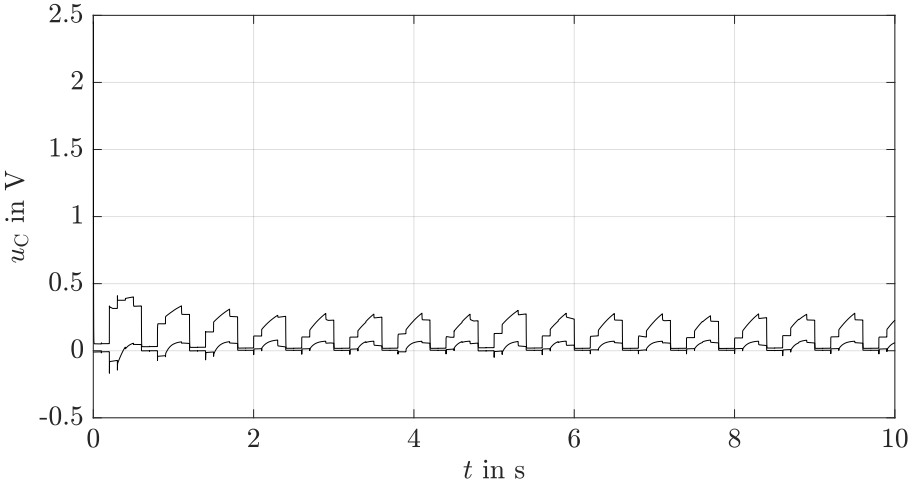


Figure 6.12: Predicted state enclosure for the second state u_{C} over the complete time horizon of 10 s including 100 variations of $R_{\mathrm{S}d}$.

Note that, again, the initial state is given by Eq. (6.16). This holds for the first time step. For the following time steps, the calculated enclosure at the end of the prior time step acts as the initial state. The control design is fast in reaching the stationary point and keeps the dynamics there despite the continued variations of the parameter. Additionally, the predicted enclosures tighten fast owing to the fact that in contrast to the two uncertainties from before only one parameter remains uncertain and, hence, overestimation due to the transformation can be reduced. Furthermore, an incremental updating of the transformation matrix according to

$$\mathbf{T}_i = \mathbf{T}_{i-1}^{-1} \cdot \Delta\mathbf{T} \tag{6.20}$$

is applied. Here, the actual transformation matrix is calculated using the information from the step $i-1$ from before as a basis for the next step i.

6.3 Down-Step Converter

Finally, a real-life application for such an electric circuit is considered. This comes in form of a down-step converter. Those systems are widely applied in electronics to reduce the output in contrast to the input voltage, which is necessary for a number of applications such as

- notebooks; to provide the processor supply voltage
- battery chargers
- semiconductors and
- LED flashlights

to name a few. Fig. 6.13 shows the electrical circuit of a down-step converter.

To realize this as a test rig, further requirements on the system are introduced to guarantee general applicability. Besides a student-friendly and cheap structure in terms of a breadboard

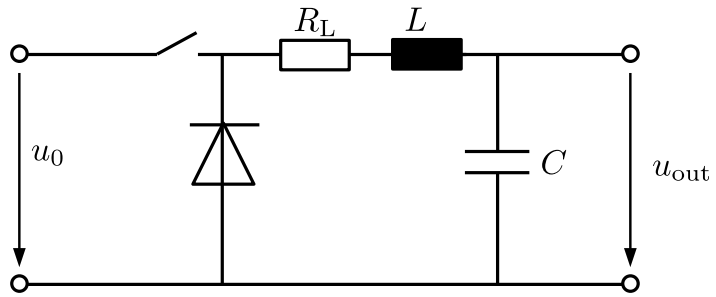


Figure 6.13: Down-step converter.

setup, low voltages and currents are necessary. Furthermore, we want to make sure that extensions can easily be added. Note that, if used as a down step converter, this electric circuit is subject to a fixed duty cycle.

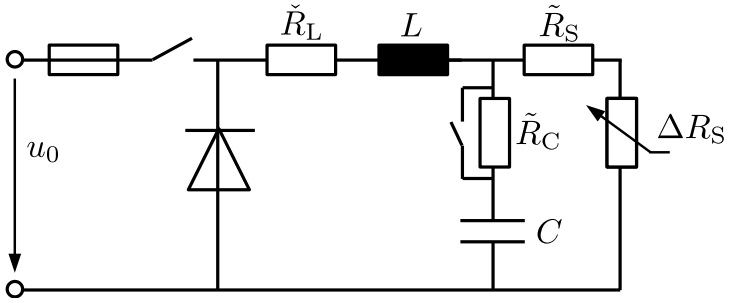


Figure 6.14: Test rig set up.

The requirements lead to the structure shown in Fig. 6.14, which highlights the similarities to the simple electric circuit system considered in the first part of this chapter. Here, a down-step converter is combined with a fuse for protection while the variable load $R_\mathrm{S} = \tilde{R}_\mathrm{S} + \Delta R_\mathrm{S}$, and the connectable additional resistor $\tilde{R}_\mathrm{C}$ to influence the break-away point of the root locus are kept. Subsequently, the model derived by the simplified circuit in Eq. (6.9) can be used again. In this case, the states for the two operating modes $u_0 = 0\,\mathrm{V}$ or $u_0 = 5\,\mathrm{V}$ shall be predicted based on the system's inherent asymptotic stability using the cooperative form. Hence, in a first step, the system (6.9) is transformed into a cooperative form. Following the general procedure, the states then can be computed in the new coordinates with

$$\dot{\tilde{\mathbf{x}}} = \mathbf{\Theta}^{-1}\mathbf{A}\mathbf{\Theta} \cdot \tilde{\mathbf{x}} + \mathbf{\Theta}^{-1}\mathbf{b} \cdot u_0 \tag{6.21}$$

and afterwards be transformed back into the original coordinates analogously as before. With the transformation procedure presented in this work, the resulting cooperative system matrix is

$$\hat{\mathbf{A}} \in \begin{bmatrix} [-1.055\ ;\ -0.557] & [0.010\ ;\ 0.818] \\ [0.228\ ;\ 0.599] & [-0.849\ ;\ -0.246] \end{bmatrix} \cdot 10^4 \ . \tag{6.22}$$

Checking the eigenvalues with

$$\lambda_i = \mathrm{eig}\left(\sup\left(\hat{\mathbf{A}}\right)\right) \ , \tag{6.23}$$

however, reveals a problem in terms of a positive real part $\lambda_2 = 0.316 \cdot 10^4$, which shows that the new system matrix contains unstable realizations in addition to obviously stable ones ($\lambda_1 = -1.119 \cdot 10^4$). Applying the Gershgorin circle theorem underlines this potential loss of stability. Since the original system is stable, this loss must result from overestimation in the

transformation step. Techniques to reduce this overestimation were presented in [51] and are briefly summarized here. A first possible solution was found by applying an interval-valued transformation matrix similar to systems including complex-conjugate eigenvalues, see Sec. 4.3. Here, the advantage of a diagonal structure may help in reducing the overestimation.

For this, the parameter boxes of the uncertain parameters R_C and R_S were subdivided in a chessboard-like manner and interval-valued eigenvalues were obtained. Figs. 6.15 and 6.16 show the predicted states i_L and u_C, respectively. The simulation was done for 75 ms.

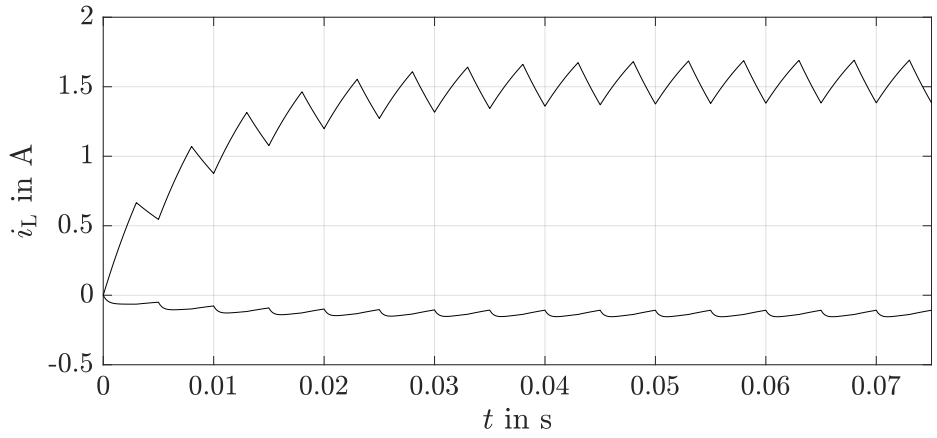


Figure 6.15: Predicted state enclosure for the down-step converter: i_L using a diagonal structure.

For this simulation, R_C was subdivided 200 times and R_S 500 times. This resulted in the interval-valued eigenvalues of

$$\begin{aligned}[\sigma_1] &= [-115.15\ ;\ -100.1] \\ [\sigma_2] &= [-2500\ ;\ -124.2]\ .\end{aligned} \tag{6.24}$$

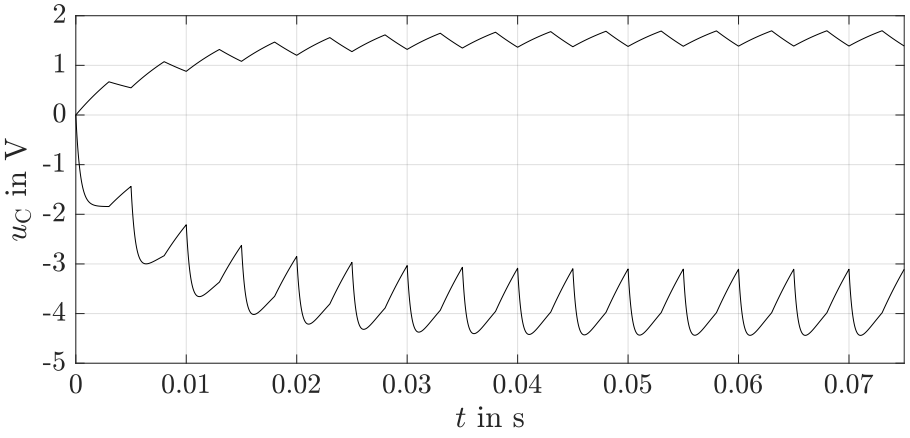


Figure 6.16: Predicted state enclosure for the down-step converter: u_C using a diagonal structure.

To find even tighter enclosures and avoid the loss of stability, [51] presented an algorithm, which uses a subdivision-based collection of local, point-valued coordinate transformations. There, the parameter space is partitioned into certain subdomains created by the algorithm. Then, point-valued transformations are obtained for each subdomain, where temporal variations of parameters are only admissible within each of the resulting parameter boxes. In reducing

the parameter variation, overestimation is naturally decreased. A manually chosen value N determines into how many subintervals an uncertain parameter is split if necessary. Here, necessity arises either if a transformation into Metzler form cannot be found after a certain predefined wall clock time or if instability after the transformation is detected despite the fact that stability of the system model was proven beforehand. Those subdomains are then again partitioned into smaller ones until admissible, unique transformation matrices are found for each subdomain. Figs. 6.17 and 6.18 show the result of applying this to the presented scenario of the down-step converter for $N = 10$ with dashed lines and $N = 50$ with solid ones. Both yield significantly tighter state enclosures as shown in Figs. 6.15 and 6.16.

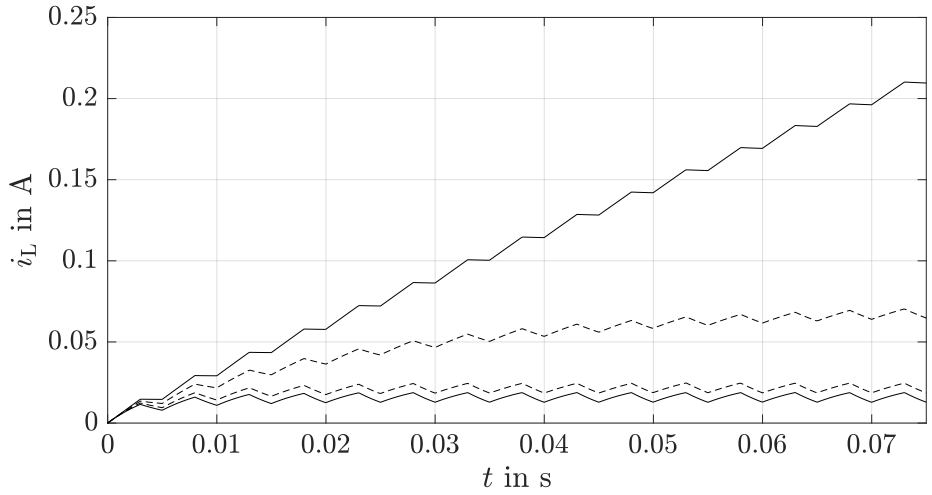


Figure 6.17: Predicted state enclosure for the down-step converter: i_L using two different numbers of subdivision.

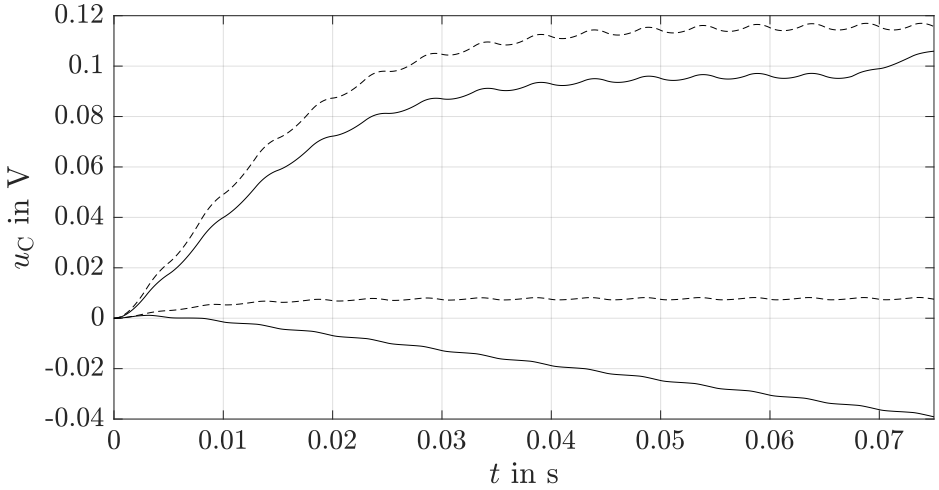
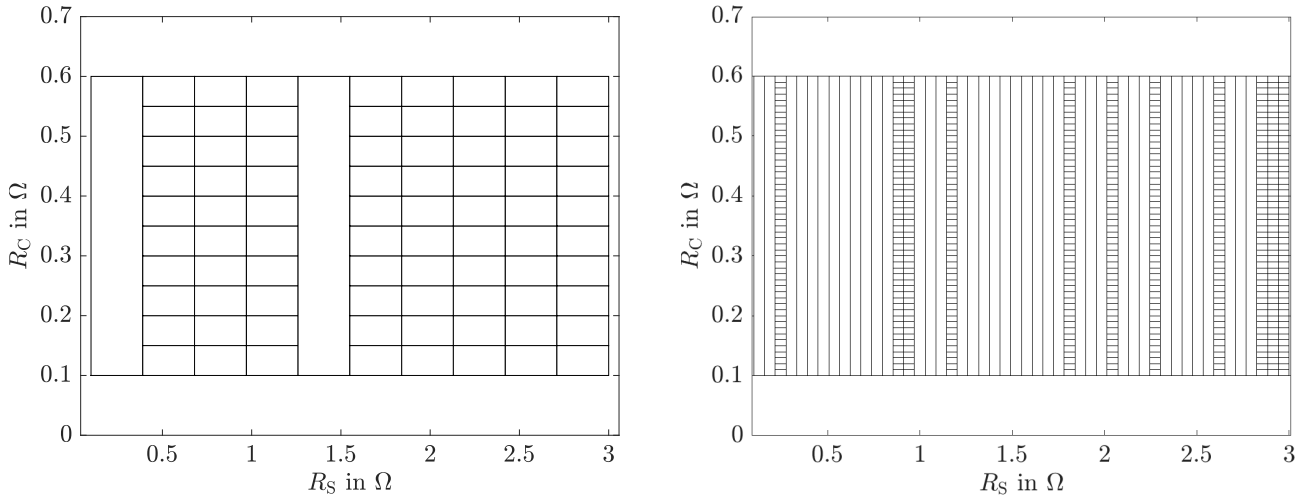


Figure 6.18: Predicted state enclosure for the down-step converter: u_C using two different numbers of subdivision.

Be reminded, that for this approach there is no joint transformation matrix for all possible parameter variations. Since the interval hull over all individual state enclosures represents the solution, the validity of the computed bounds is restricted to variations of the parameters within each of the subdomains determined by the parameter boxes. Those boxes and, hence, admissible subdomains are given in Fig. 6.19, for both $N = 10$ in Fig. 6.19(a) and $N = 50$ in Fig. 6.19(b). Obviously, the increased number of splitting results in more and subsequently

smaller subdomains. However, those smaller subdomains do not necessarily yield tighter interval bounds as the simulations of the predicted states show. Hence, an arbitrary increase of the parameter N does not lead to favorable simulation results. A possible optimization of this parameter may help and should be investigated in future work as also stated in [51].



(a) Splitting with $N = 10$

(b) Splitting with $N = 50$

Figure 6.19: Resulting subdomains determined by the parameter boxes.

Note that for all applied simulations Eqs. (1.5)–(1.7) are used accordingly.

7 Oscillations in Mechanical Engineering - Two Examples for Systems Including Complex Eigenvalues

A broad number of mechanical systems are subject to oscillations. Sometimes those can be desired, i.e., in a shaker to separate coarser material from finer particles, but in other cases oscillations need to be damped to reduce material wear or guarantee safety measures. This chapter will deal with two applications, where oscillations are attenuated by the controller. The first one regards a high-bay rack feeder as a linear system with uncertain parameters in the system matrix, while the second scenario deals with a boom crane as a nonlinear example. The latter includes uncertain parameters as well as a state dependency in the system matrix of a suitable quasi-linear state-space representation.

7.1 Oscillation Attenuation for a High-Bay Rack Feeder

Rack feeders are widely used as an operating system for automatic performances in high-bay warehouses, which are built in terms of light-weighted structures into height rather than covering too much space horizontally. Obviously, those systems can be improved by control to efficiently follow a trajectory while simultaneously reducing structural oscillations. This would lead to reduced transport times and, hence, increase transshipment capacity.

s shown. It is based on an

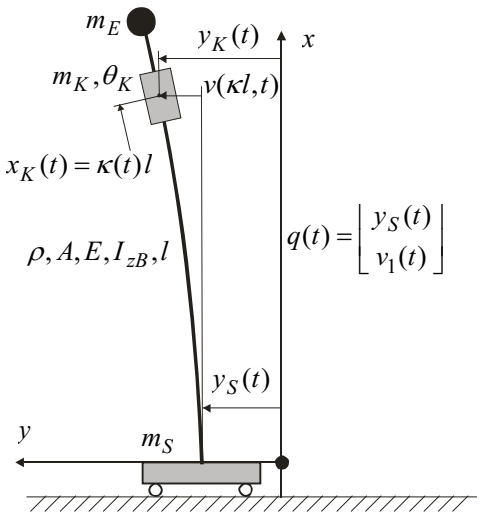


Figure 7.1: Mechanical model of the stacker crane.

existing prototypical test rig at the Chair of Mechatronics at the University of Rostock, which was built for mimicking the reality. For this, a carriage with the mass m_S moves along a track carrying a double beam. Additionally, a mass m_K imitating a cage is mounted on said beam moving vertically along. A control-oriented modeling was already developed in [2,54]. Here, the system is considered as an elastic multibody model consisting of the three rigid bodies already mentioned; the carriage, an end mass m_E at the tip as well as the additional mass of the cage, and a spatially distributed Bernoulli beam as an elastic component. The latter is defined by the mass moment of inertia θ_K, the density ρ, the cross sectional area A, Young's modulus E, the second moment of area I_{zB}, and the length l. To describe the time-varying vertical position $x_K(t)$ of the cage on the beam, a dimensionless system parameter

$$\kappa\left(t\right) = \frac{x_K\left(t\right)}{l} \tag{7.1}$$

is formulated and assumed to be measured. A one-dimensional Ritz ansatz

$$v_1\left(x,t\right) = \bar{\bar{v}}_1\left(x\right) v_1\left(t\right) \quad \text{with} \quad \bar{\bar{v}}_1\left(x\right) = \frac{3}{2}\left(\frac{x}{l}\right)^2 - \frac{1}{2}\left(\frac{x}{l}\right)^3 \tag{7.2}$$

accounts for the bending deflection of the beam structure by the corresponding elastic degree of freedom for the first bending mode, which is suitable for small bending. The second-order ordinary differential equation model

$$\mathbf{M}\ddot{\mathbf{q}}\left(t\right) + \mathbf{D}\dot{\mathbf{q}}\left(t\right) + \mathbf{K}\mathbf{q}\left(t\right) = \mathbf{g} \cdot \left(F_{SM}\left(t\right) - F_{SR}\left(t\right)\right) \tag{7.3}$$

is derived by applying Lagrange's equations of second kind, cf. [60] with the mass matrix $\mathbf{M}$, the damping matrix $\mathbf{D}$, the stiffness matrix $\mathbf{K}$, and the input vector of generalized forces $\mathbf{g}$. The generalized force itself is given by the difference of the actuation force of the carriage F_{SM} as well as a friction force F_{SR} acting opposite to the direction of motion. Here, the vector of generalized coordinates consists of

$$\mathbf{q}\left(t\right) = \left[\begin{array}{cc} y_S\left(t\right) & v_1\left(t\right) \end{array}\right]^T , \tag{7.4}$$

the carriage position and the bending deflection of the considered first eigenmode. The test rig in question is equipped with an underlying velocity control operating on the electric drive for the carriage with mass m_S. This means, that the resulting dynamics for $\ddot{y}_S$ can be replaced by the first-order lag dynamics

$$T_1 \ddot{y}_S\left(t\right) + \dot{y}_S\left(t\right) = v_S\left(t\right) - v_{S0} \tag{7.5}$$

with the time constant T_1 and the usually small velocity disturbance v_{S0}. However, experiments show that this velocity control does not display the real dynamics completely, which happens due to inertia couplings via the reaction forces from the beam motion backward to the carriage. In [2], an observer-based estimation for v_{S0} was implemented to account for that. Another approach is presented in the following. Here, v_{S0} is neglected and the resulting imperfection in the dynamics of the underlying velocity control will be added to the original uncontrolled dynamics with an uncertain factor η, which specifies a level of confidence, that is put in the underlying velocity control. The assumption that the underlying velocity control displays the model with an accuracy of $80 - 97\%$ results in

$$\eta = \left[0.8\ ;\ 0.97\right] \ . \tag{7.6}$$

For the sake of compactness, time arguments are omitted in the following. Finally, when substituting the ODE (7.5) into (7.3), the equations of motion are given by

$$\ddot{\mathbf{q}} = -\mathbf{M}^{-1}\mathbf{K}\mathbf{q} - \mathbf{M}^{-1}\mathbf{D}\dot{\mathbf{q}} + \mathbf{M}^{-1}\mathbf{g}\, v_S + \mathbf{e} \ . \tag{7.7}$$

Here, v_S is the desired carriage velocity acting as the new control input u. The original inputs are compensated for by choosing the mistrust of the underlying control to be larger than necessary as well as in an additional error vector $\mathbf{e}$, which adds bounded noise to the accelerations of the system. The mistrust or rather the level of confidence of the controller is also included in the mass and damping matrices. The modified mass matrix

$$\mathbf{M}(\eta) = \begin{bmatrix} (1-\eta)\cdot m_{11} + \eta \cdot T_1 & (1-\eta)\cdot m_{12} \\ m_{12} & m_{22} \end{bmatrix} \tag{7.8}$$

includes

$$\begin{aligned} m_{11} &= m_S + \rho A l + m_K + m_E \ , \\ m_{12} &= \frac{3}{8}\rho A l + \frac{m_K \kappa^2}{2}(3-\kappa) + m_E \ , \\ m_{22} &= \frac{33}{140}\rho A l + \frac{6\rho I_{zB}}{5l} + \frac{m_K \kappa^4}{4}(3-\kappa)^2 + m_E \ . \end{aligned} \tag{7.9}$$

The damping matrix corresponds to

$$\mathbf{D}(\eta) = \begin{bmatrix} \eta & 0 \\ 0 & \frac{3k_d E I_{zB}}{l^3} \end{bmatrix} \ , \tag{7.10}$$

while the stiffness matrix is represented by

$$\mathbf{K} = \begin{bmatrix} 0 & 0 \\ 0 & k_{22} \end{bmatrix} \tag{7.11}$$

with

$$k_{22} = \frac{3EI_{zB}}{l^3} - \frac{3}{8}\rho A g - \frac{3m_K g \kappa^3}{l}\left(1 + \frac{3\kappa^2}{20} - \frac{3\kappa}{4}\right) - \frac{6m_E g}{5l} \ . \tag{7.12}$$

Lastly, the input vector of generalized forces is given by $\mathbf{g} = \begin{bmatrix} \eta & 0 \end{bmatrix}^T$. Table 7.1 summarizes all parameters of the considered test rig.

7.1.2 Control

For the system to be controlled by a robust LMI-based feedback controller $v_S = -\mathbf{k}^T \cdot \begin{bmatrix} \mathbf{q^T} & \dot{\mathbf{q}}^\mathbf{T} \end{bmatrix}^T$ with the approach of Chapter 3, it is transformed into its state-space representation

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{g} \end{bmatrix} v_S + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\delta} \end{bmatrix} \tag{7.13}$$

with a parameter uncertainty in $\mathbf{M}$ and $\mathbf{D}$ due to their dependence on η and a disturbance $\boldsymbol{\delta} > \mathbf{0}$ influencing both acceleration terms. Here, δ_1 also includes ignored values from v_{S0} while δ_2 covers modeling errors in terms of worst-case bounds, e.g. ignored dynamics of higher order. In the considered scenario, κ is assumed to be a constant parameter given by $\kappa = 0.75$. For this example, the Γ-region, see Eq. (2.45) and Fig. 2.5, is given by an absolute stability margin

Table 7.1: Parameters of the high-bay rack feeder.

Variable	Unit/Value	Meaning
A	$6 \cdot 10^{-4}\,\mathrm{m}^2$	cross sectional area of the beam
l	$1.07\,\mathrm{m}$	length of the beam
ρ	$2.7 \cdot 10^{3}\,\frac{\mathrm{kg}}{\mathrm{m}^3}$	density of the beam
E	$70 \cdot 10^{9}\,\frac{\mathrm{N}}{\mathrm{m}^2}$	Young's modulus of the beam
I_{zB}	$1.2 \cdot 10^{-8}\,\mathrm{m}^4$	the second moment of area of the beam
m_K	$0.95\,\mathrm{kg}$	mass of the cage
m_S	$2\,\mathrm{kg}$	mass of the carriage
m_E	$0.9\,\mathrm{kg}$	end mass at the tip of the beam
k_d	$1.1 \cdot 10^{-3}\,\frac{\mathrm{N}}{\mathrm{m}}$	damping factor
T_1	$7 \cdot 10^{-3}\,\mathrm{s}$	time constant of the underlying velocity control

of $\gamma = 4$, thus the setting of the matrices in Eq. (2.46) corresponds to $D_0 = 2\gamma$ and $D_1 = 1$. With the controller gain

$$\mathbf{k}^T = \begin{bmatrix} 2.44 \cdot 10^2 & -3.90 \cdot 10^2 & 42.52 & 12.24 \end{bmatrix} , \tag{7.14}$$

the controlled system matrix results in

$$\begin{aligned} \mathbf{A}_\mathrm{C} &= (\mathbf{A} - \mathbf{BK}) \\ &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ [a_{31}] & [a_{32}] & [a_{33}] & [a_{34}] \\ [a_{41}] & [a_{42}] & [a_{43}] & [a_{44}] \end{bmatrix} \end{aligned}$$

with

$$\begin{aligned} a_{31} &\in [-2.570\ ;\ -0.339] \cdot 10^3 & a_{41} &\in [0.432\ ;\ 3.274] \cdot 10^3 \\ a_{32} &\in [0.678\ ;\ 9.740] \cdot 10^3 & a_{42} &\in [-1.992\ ;\ -0.105] \cdot 10^4 \\ a_{33} &\in [-4.584\ ;\ -0.606] \cdot 10^2 & a_{43} &\in [0.772\ ;\ 5.838] \cdot 10^2 \\ a_{34} &\in [-1.281\ ;\ -0.160] \cdot 10^2 & a_{44} &\in [0.191\ ;\ 1.617] \cdot 10^2\ . \end{aligned} \tag{7.15}$$

The eigenvalue placement is given in Fig. 7.2, where one can clearly see that the system is stable since all real parts lie in the open left half plane.

A detailed view for the slower eigenvalues is given in Fig. 7.3. Obviously, the required stability margin is robustly satisfied by design.

7.1.3 Transformation

The transformation is done using a real-valued transformation matrix according to Sec. 4.3 with Eqs. (4.36)–(4.42). A respective system matrix (4.43) results in

$$\tilde{\mathbf{A}}_\mathrm{C} \in \begin{bmatrix} [\tilde{a}_{11}] & 0 & 0 & 0 \\ 0 & [\tilde{a}_{22}] & 0 & 0 \\ 0 & 0 & [\tilde{a}_{33}] & 0 \\ 0 & 0 & 0 & [\tilde{a}_{44}] \end{bmatrix}$$

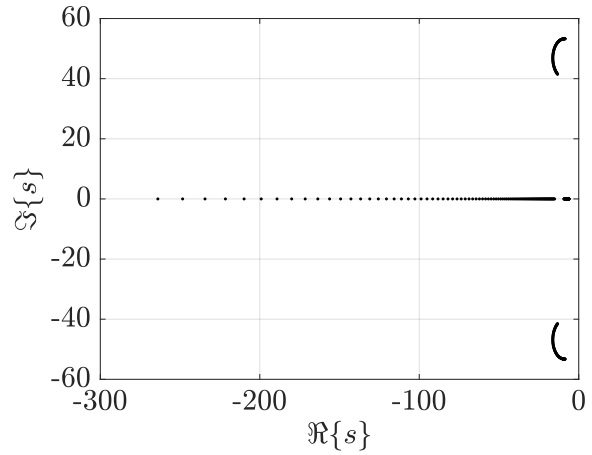


Figure 7.2: Distribution of eigenvalues for the controlled system $\mathbf{A}_\mathrm{C}$.

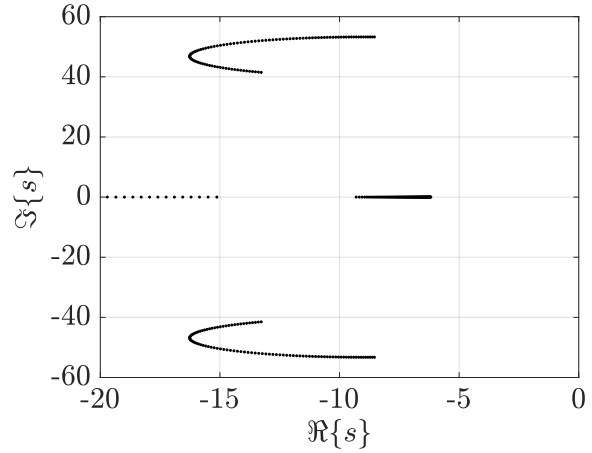


Figure 7.3: Distribution of eigenvalues for the controlled system $\mathbf{A}_\mathrm{C}$ — enlarged view.

with the closed-loop matrix entries

$$\begin{aligned}[\tilde{a}_{11}] &= [-2.681\ ;\ -0.148] \cdot 10^2 \\ [\tilde{a}_{22}] &= [9.503\ ;\ -5.503] \\ [\tilde{a}_{33}] = [\tilde{a}_{44}] &= [17.242\ ;\ -8.465]\ .\end{aligned}$$

When applying a complex-valued transformation according to Sec. 4.3 with Eqs. (4.50)–(4.52), the result would change to

$$\tilde{\mathbf{A}}_\mathrm{C} \in \begin{bmatrix} [\tilde{a}_{11}] & 0 & 0 & 0 \\ 0 & [\tilde{a}_{22}] & 0 & 0 \\ 0 & 0 & [\tilde{a}_{33}] & 0 \\ 0 & 0 & 0 & [\tilde{a}_{44}] \end{bmatrix}$$

with the closed-loop matrix entries

$$\begin{aligned}[\tilde{a}_{11}] &= \langle -1.414,\ 1.267 \rangle \cdot 10^2 \\ [\tilde{a}_{22}] &= \langle -7.503,\ 2.000 \rangle \\ [\tilde{a}_{33}] &= \langle -12.853 + 45.445\jmath,\ 12.307 \rangle \\ [\tilde{a}_{44}] &= \langle -12.853 - 45.445\jmath,\ 12.307 \rangle\ .\end{aligned}$$

Note that the solution is given in the midpoint-radius form, see Fig. 2.1(b), where the radius is rounded upward to the number of displayed digits. In the present case, the real-valued transformation produces less overestimation and is therefore used for the simulations in the following. However, the complex-valued transformation provides a useful information regarding the eigenvalues. Here, it becomes clear that for an unfavorable partitioning into subintervals, the overall system matrix may become unstable due to overestimation. This is to be considered, when analyzing the system and designing the transformation.

7.1.4 Numerical results

Using the transformation from the previous section and Approach (3) of 4.3.2, the interval enclosures are calculated. Note that a common approach with the state-of-the-art Taylor series expansion is difficult due to the wrapping effect since the evaluation of the n symbolically given system matrices using standard interval techniques for the chosen interval of η yields a solution that includes unstable eigenvalues of the system matrix. A comparison is, therefore, not fair for this specific example since the system neither reaches the desired operating point $y_{S0} = 0\,\mathrm{m}$ nor produces useful interval widths, as can be seen in Fig. 7.4.

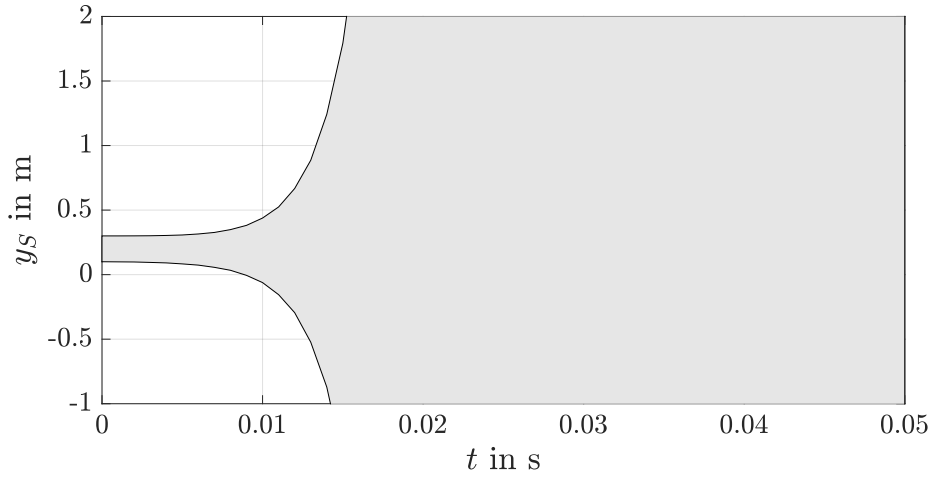


Figure 7.4: State prediction for the carriage position y_S with a Taylor series expansion of order 2 for a shortened time horizon of 50 ms.

The simulation was done with an initial uncertainty of $y_S = [0.1\ ;\ 0.3]\,\mathrm{m}$, which also holds for the cooperativity-based approach. It is given in Fig. 7.5. In contrast to the Taylor series expansion, both decoupled bounds of the cooperativity-based method converge rapidly to said stationary operating point due to the efficient controller and the resulting asymptotic stability of the system.[1] As in previous examples, there is a small deviation from the true[2] state in the starting phase. Again, this is due to overapproximation in the transformation and stems, in this example from Eqs. (4.36)–(4.43) and the resulting overestimation of the complex eigenvalues. Combining both Taylor series expansion and cooperativity-based methods, however, would not be possible in this scenario since no stable matrix could be found to be used in the Taylor series expansion. Nevertheless, the overestimation problem of the cooperativity approach in the starting phase occurs in a very short time interval of approximately 0.19 s till realistic values

[1] Note that implementing a feedforward control, this asymptotic stability would be reduced to input to state stability (ISS). This is due to bounded uncertainty in the mass matrix $\mathbf{M}$, which typically imposes uncertainty in the stationary system gain.

[2] The rack feeder in question is a small-scale model on a 1.5m test rig.

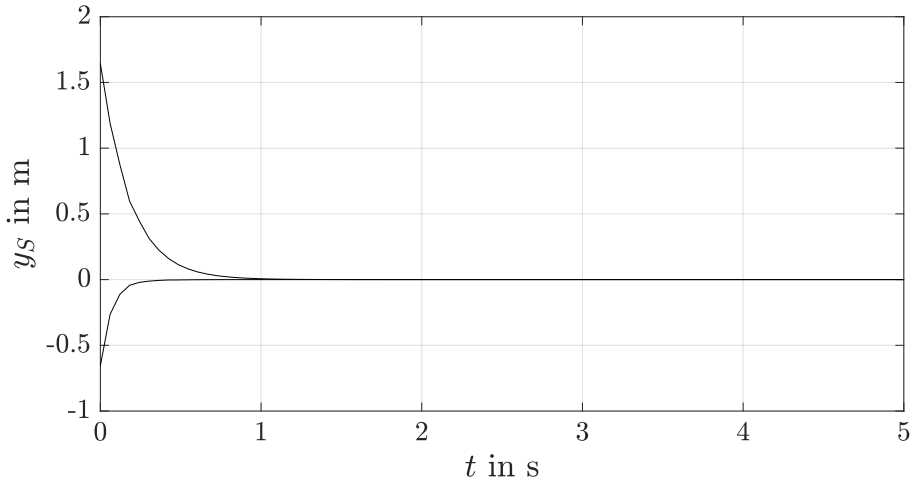


Figure 7.5: Upper and lower bound of the interval for the carriage position y_S for all $t \in [0\ ;\ t_\mathrm{f}]$

— regarding the test rig — are reached in the interval-based simulation. To emphasize the meaning of this, the response times of the presented cooperativity-based method are compared with a grid-based simulation for the parameter η. The original system was simulated with ten equally spaced grid-points included in the interval of η. In Table 7.2, times show when the specified deviation to the stationary point $y_S = 0$ is reached in each approach.

Table 7.2: Comparison of response times for the interval computation vs. a parameter gridding.

	deviation		
computation via	≤ 0.1m	≤ 0.05m	≤ 0.01m
gridding	0.15 s	0.25 s	0.51 s
interval	0.51 s	0.63 s	0.93 s

Despite the fact that the gridding method predicts shorter settling times, the problem in gridding is to find suitable values to include all worst cases. In contrast to that, the interval method definitely includes such cases as it is oriented on the slowest time constant of the overall system. Considering this big advantage, it can be shown that except for the starting phase, the response time of the controlled system can be estimated well by means of the interval procedure. However, it may happen that for some cases, this too long starting phase increases the settling time unnecessarily. This happens, e.g. for an implementation of variable values for κ to predict states based on a gain scheduling control for a given trajectory for κ. For now, κ was implemented as a constant parameter, yet, as Eq. (7.1) shows, it may vary over time, which is not considered in the presented scenario. As it is, the presented method is not usable for this extension. Nonetheless, a look-up table connecting κ values to their respective controller gain could help in solving this if the parameter κ can be measured accurately.

7.1.5 Observer

For the given example a cooperativity-preserving controller is difficult to implement due to the fact that the considered system already leans towards numerical instability. As mentioned in Chapter 5, a possible solution is to calculate the observer for the original system and transform the overall system including the observer into cooperative form. For this, the observer gain is

calculated using the method in Sec. 5.2 for the closed-loop to follow

$$\mathbf{A}_{\mathrm{O}} = \mathbf{A}_{\mathrm{C}} - \mathbf{H}\mathbf{C}_{\mathrm{m}} \ , \tag{7.16}$$

where

$$\mathbf{C}_{\mathrm{m}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{7.17}$$

assuming that the carriage position as well as the bending deflection can be measured. Corresponding to the Γ-region of the control design, the observer is implemented with an absolute stability margin of $\gamma = 6$. Hence, the resulting observer gain

$$\mathbf{H} = \begin{bmatrix} 25.63 & 25.95 \\ 3.33 & 26.08 \\ 380.00 & 987.95 \\ 119.07 & -1.35 \cdot 10^3 \end{bmatrix} \tag{7.18}$$

was calculated, so that the overall system matrix becomes

$$\begin{aligned} \mathbf{A}_{\mathrm{CO}} &= (\mathbf{A}_{\mathrm{C}} - \mathbf{H}\mathbf{C}_{\mathrm{m}}) \\ &\in \begin{bmatrix} -25.63 & -25.95 & 1 & 0 \\ -3.33 & -26.08 & 0 & 1 \\ [a_{31}] & [a_{32}] & [a_{33}] & [a_{34}] \\ [a_{41}] & [a_{42}] & [a_{43}] & [a_{44}] \end{bmatrix} \end{aligned} \tag{7.19}$$

with

$$\begin{aligned} a_{31} &\in [-2.950 \ ; \ -0.719] \cdot 10^3 & a_{41} &\in [0.313 \ ; \ 3.154] \cdot 10^3 \\ a_{32} &\in [-0.194 \ ; \ 8.032] \cdot 10^3 & a_{42} &\in [-1.765 \ ; \ 0.015] \cdot 10^4 \\ a_{33} &\in [-458.335 \ ; \ -60.617] & a_{43} &\in [77.203 \ ; \ 583.741] \\ a_{34} &\in [-128.004 \ ; \ -16.049] & a_{44} &\in [19.101 \ ; \ 162.687] \ . \end{aligned} \tag{7.20}$$

in comparison to (7.15). The resulting new eigenvalue locations are shown in Fig. 7.6.

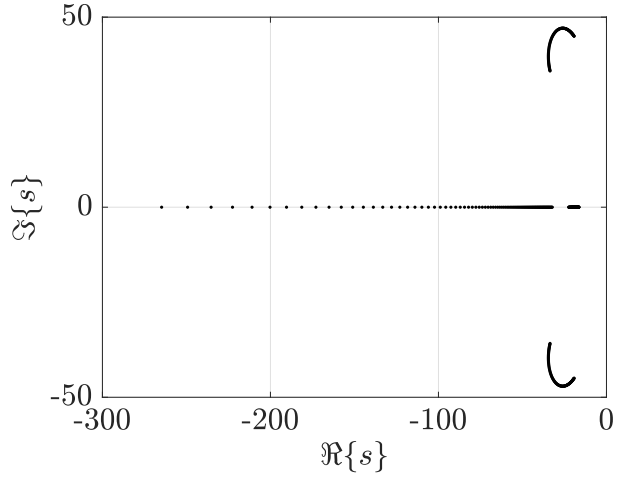


Figure 7.6: Distribution of eigenvalues for the controlled and observed system $\mathbf{A}_{\mathrm{CO}}$.

In contrast to this approach, [53] presented a stochastical method in terms of a Kalman filter.

7.1.6 Transformation of the parallel model

For this model, a real-valued transformation matrix was found for the given uncertain parameter η in Eq. (7.6). The system matrix (4.43) then results in

$$\tilde{\mathbf{A}}_{\mathrm{CO}} = \mathbf{N} \in \begin{bmatrix} [\tilde{a}_{11}] & 0 & 0 & 0 \\ 0 & [\tilde{a}_{22}] & 0 & 0 \\ 0 & 0 & [\tilde{a}_{33}] & 0 \\ 0 & 0 & 0 & [\tilde{a}_{44}] \end{bmatrix} \tag{7.21}$$

with the closed-loop matrix entries

$$\begin{aligned} [\tilde{a}_{11}] &= [-265.161\ ;\ -32.225] \\ [\tilde{a}_{22}] &= [-22.523\ ;\ -16.198] \\ [\tilde{a}_{33}] = [\tilde{a}_{44}] &= [-34.711\ ;\ -19.234]\ . \end{aligned}$$

7.1.7 Numerical results of the parallel simulation

Now, a parallel model is simulated to account for a comparison with the controlled system to use it as a fault diagnosis tool for the controlled system. The presented parallel model already considers the impact of the controller by Eq. (7.19), however, an additional uncertainty regarding the input errors is implemented with an actuator inaccuracy $\Delta\mathbf{u}$ resulting in

$$\dot{\hat{\mathbf{x}}} = \left(\mathbf{A}_{\mathrm{C}} - \mathbf{H}\mathbf{C}_{\mathrm{m}}\right) \cdot \hat{\mathbf{x}} + \mathbf{H}\mathbf{y}_{\mathrm{m}} + \mathbf{B}\Delta\mathbf{u}\ , \tag{7.22}$$

cf. Eqs. (5.3) and (5.4). Here, measurements result from the midpoint of the predicted states of the controlled system $\mathbf{y}_{\mathrm{m}} = \mathbf{C}_{\mathrm{m}}\mathrm{mid}\left\{[\mathbf{x}_{\mathrm{C}}]\right\} + \Delta\mathbf{y}_{\mathrm{m}}$ with an added sensor inaccuracy of $\Delta\mathbf{y}_{\mathrm{m}} \in \left[[-5\cdot10^{-4}\ ;\ 5\cdot10^{-4}]\quad[-1\cdot10^{-6}\ ;\ 1\cdot10^{-6}]\right]^T$ and $\Delta u \in [-1\cdot10^{-5}\ ;\ 1\cdot10^{-5}]$.

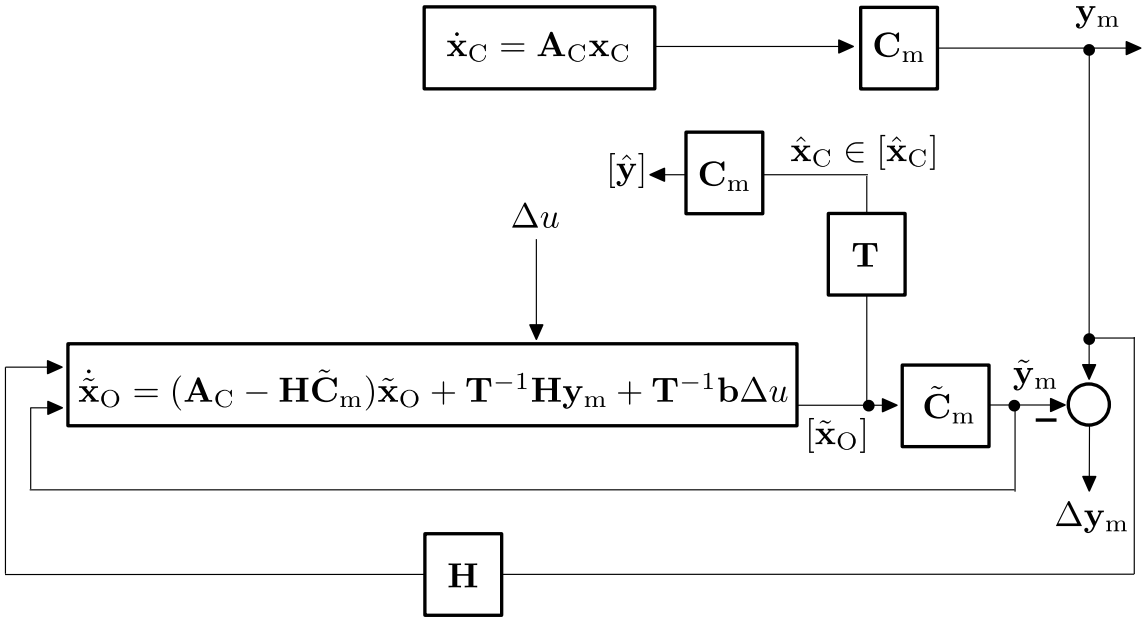


Figure 7.7: Use of the observer with a parallel model as a fault diagnosis tool.

The resulting time course of the observed carriage position is given in Fig. 7.8.

Here, a fault would be detected if those bounds are exceeded by the system motion. Note that the smaller — in comparison to Sec. 7.1.4 — starting interval of $y_{S,\mathrm{m}} = [0.199\ ;\ 0.201]\,\mathrm{m}$ was used for the simulation of the parallel model.

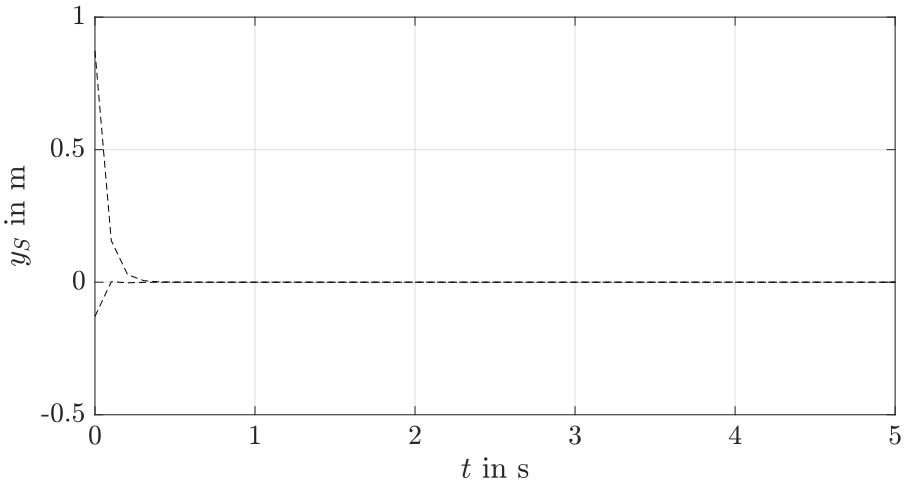


Figure 7.8: Upper and lower bound of the interval for the observed carriage position $\hat{y}_S$ for all $t \in [0\ ;\ t_\mathrm{f}]$

7.2 Oscillation Attenuation of a Boom Crane Load

Similar to the first application scenario of this chapter, oscillations of a boom crane load increase the transport time while reducing the security of the overall process. A controlled boom crane furthermore facilitates the handling for the operator. The task is, hence, an oscillation attenuation for the boom crane's payload.

7.2.1 Modeling

A schematic representation of the system is given in Fig. 7.9.

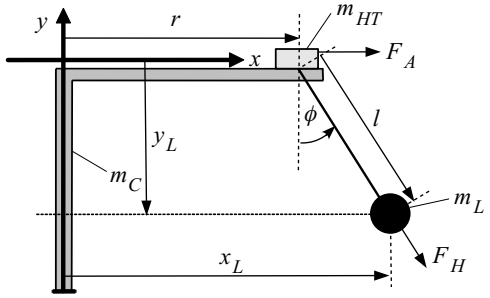


Figure 7.9: Schematic representation of a boom crane with the generalized coordinates of the position r and the angle ϕ.

A mass of a payload m_L is connected by a rope of length l to a moving carriage m_{HT}. Here, the rope length is variable and is, hence, chosen as an uncertain parameter. Like in Sec. 7.1, Lagrange's equations of second kind

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}(t)}\right) - \frac{\partial L}{\partial \mathbf{q}(t)} + \frac{\partial D}{\partial \dot{\mathbf{q}}} = \mathbf{Q}\ , \tag{7.23}$$

are used to derive the equations of motion. The Lagrange function represents the difference of kinetic and potential energy

$$L = E_\mathrm{kin} - E_\mathrm{pot}\ . \tag{7.24}$$

Here, the overall kinetic energy of the system is

$$\begin{aligned} E_{\text{kin}} &= E_{\text{kin},HT} + E_{\text{kin},L} \\ &= \frac{1}{2} m_{HT} \dot{r}^2(t) + \frac{1}{2} m_L \cdot \left[\left(\dot{r}(t) + l \cos(\phi(t)) \dot{\phi}(t) \right)^2 + \left(l \sin(\phi(t)) \dot{\phi}(t) \right)^2 \right] , \end{aligned} \tag{7.25}$$

while the potential energy is described by

$$E_{\text{pot}} = E_{\text{pot},L} = m_L l g \cos(\phi(t)) \ . \tag{7.26}$$

Furthermore, the vector of external forces consists of

$$\mathbf{Q} = \begin{bmatrix} F_A & 0 \end{bmatrix}^T \tag{7.27}$$

with the force F_A acting on the carriage. Internal velocity-proportional dissipation of the payload energy is included with the help of a dissipation term using the Rayleigh function

$$D = \frac{1}{2} \delta \dot{\phi}^2 \ . \tag{7.28}$$

Evaluating Lagrange's equations (7.23) with the vector of generalized coordinates $\mathbf{q} = \begin{bmatrix} r(t) & \phi(t) \end{bmatrix}^T$ results in the two equations of motion

$$(m_{HT} + m_L) \ddot{r} + m_L l \cos(\phi) \ddot{\phi} - m_L l \sin(\phi) \dot{\phi}^2 = F_A \tag{7.29}$$

and

$$m_L l \cos(\phi) \ddot{r} + m_L l^2 \ddot{\phi} + m_L l g \sin(\phi) + \delta \dot{\phi} = 0 \ . \tag{7.30}$$

Again, an underlying velocity control for the carriage is employed, where

$$T_1 \ddot{r}(t) + \dot{r}(t) = v(t) \tag{7.31}$$

replaces Eq. (7.29). A state-space representation

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{T_1} & 0 \\ 0 & -\frac{g \cdot \text{si}(\phi)}{l} & \frac{\cos(\phi)}{T_1 l} & -d \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_1} \\ -\frac{\cos(\phi)}{T_1 l} \end{bmatrix} u \tag{7.32}$$

with

$$\text{si}(\phi) = \frac{\sin(\phi)}{\phi} \tag{7.33}$$

is derived for the state vector

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \in \mathbb{R}^4 \tag{7.34}$$

consisting of the vector of generalized coordinates and their respective velocities. Here, the desired carriage velocity v denotes the input u. Uncertain parameters are the rope length l and the damping coefficient $d = \frac{\delta}{m_L}$ as well as the state-dependency of ϕ, which is also treated as an interval-bounded uncertainty. Since ϕ is included in combination with a cosine function in entries in $\mathbf{A}(\mathbf{x})$ and equally in $\mathbf{b}(\mathbf{x})$ as well as in combination with a sinc-function[3] in $\mathbf{A}(\mathbf{x})$, two independent parameters

$$p_1 = \frac{g \cdot \text{si}(\phi)}{l} \quad \text{with} \quad \text{si}(\phi) = \frac{\sin \phi}{\phi} \ , \quad \text{and} \quad p_2 = \frac{\cos(\phi)}{T_1 \cdot l} \tag{7.35}$$

[3]Here, in the not normalized form.

are included to account for this parameter dependency. Replacing the respective matrix entries, the parameter-dependent system matrix and input vector

$$\mathbf{A}(\mathbf{p}) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{T_1} & 0 \\ 0 & -p_1 & p_2 & -d \end{bmatrix} \quad \text{and} \quad \mathbf{b}(\mathbf{p}) = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_1} \\ -p_2 \end{bmatrix} , \tag{7.36}$$

are obtained.

7.2.2 Control

To account for a more robust approach as in the previous examples, worst-case scenarios assuming independent parameters p_1 and p_2 are considered to parameterize a robust control. Afterwards, the transformation, and hence, the simulation is done with finer intervals also considering partially dependent parameters as explained in Sec. 3.2. This means that the reality is overapproximated by the simulation scenario which is robustly controlled by an even higher overapproximation accounting for a secure enclosure of the reality and its eventual deviations from the model. For this, the controller is calculated for the following interval-bounded parameters of[4]

$$\begin{aligned} [\phi] &= [-0.1\ ;\ 0.1]\,\text{rad} \\ [l] &= [0.1\ ;\ 0.5]\,\text{m} \\ [d] &= [0.3\ ;\ 1.5]\,\text{Nm}\cdot\text{s}\ . \end{aligned} \tag{7.37}$$

Here, the LMI-based controller includes a stability margin of $\gamma = 0.1$ and the H_2-norm with no disturbances $\mathbf{B}_1 = \mathbf{0}$, see Eqs. (3.29)–(3.30). In the presented case, the H_2-norm is used as a robust LQR (linear-quadratic regulator) design with

$$J = \frac{1}{2}\int_0^\infty \left(\mathbf{x}^T\mathcal{Q}\mathbf{x} + \mathbf{u}^T\mathcal{R}\mathbf{u}\right)\mathrm{d}t \tag{7.38}$$

and

$$\mathbf{C}_2 = \begin{bmatrix} \mathbf{0} \\ \mathcal{Q}^{\frac{1}{2}} \end{bmatrix} \quad \text{and} \quad \mathbf{D}_{22} = \begin{bmatrix} \mathcal{R}^{\frac{1}{2}} \\ \mathbf{0} \end{bmatrix} , \tag{7.39}$$

where $\mathcal{Q} = \mathrm{diag}(\begin{bmatrix} 1 & 0.1 & 1 & 0.1 \end{bmatrix})$ and $\mathcal{R} = 1$, cf [1]. With those specifications, the controller gains result in

$$\mathbf{k}^T = \begin{bmatrix} 0.01 & -0.13 & -0.94 & -0.16 \end{bmatrix} . \tag{7.40}$$

Additionally, a further successful stability proof was done for a widened angle interval of $[\phi] = [-1\ ;\ 1]\,\text{rad}$, resulting in the controlled system matrix

$$\mathbf{A}_\mathrm{C} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{7.41}$$

with

$$\begin{aligned} a_{31} &\in [-0.197\ ;\ -0.196] & a_{41} &\in [0.392\ ;\ 1.970] \\ a_{32} &\in [2.625\ ;\ 2.626] & a_{42} &\in [-124.356\ ;\ -24.812] \\ a_{33} &\in [-1.254\ ;\ -1.253] & a_{43} &\in [2.495\ ;\ 12.540] \\ a_{34} &\in [3.048\ ;\ 3.049] & a_{44} &\in [-31.683\ ;\ -6.565]\ . \end{aligned} \tag{7.42}$$

[4]Note that those parameters comply with a scaled model of a real boom crane [52].

7.2.3 Transformation

As mentioned, smaller intervals

$$\begin{aligned}[\phi] &= [-0.1\ ;\ 0.1]\,\mathrm{rad}\\ [l] &= [0.4\ ;\ 0.5]\,\mathrm{m}\\ [d] &= [0.5\ ;\ 1.2]\,\mathrm{Nm}\cdot\mathrm{s}\end{aligned} \tag{7.43}$$

representing the domain of operation as a subset of the stabilized state domain for the system will be used for the transformation. The controlled matrix then becomes

$$\mathbf{A}_\mathrm{C} = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ a_{31} & a_{32} & a_{33} & a_{34}\\ a_{41} & a_{42} & a_{43} & a_{44}\end{bmatrix} \tag{7.44}$$

with

$$\begin{aligned} a_{31} &\in [-0.197\ ;\ -0.196] & a_{41} &\in [0.392\ ;\ 0.493]\\ a_{32} &\in [2.625\ ;\ 2.626] & a_{42} &\in [-31.089\ ;\ -24.812]\\ a_{33} &\in [-1.254\ ;\ -1.253] & a_{43} &\in [2.495\ ;\ 3.135]\\ a_{34} &\in [3.048\ ;\ 3.049] & a_{44} &\in [-8.821\ ;\ -6.565]\ . \end{aligned} \tag{7.45}$$

For this example, we want to find an intelligent approach to subdivide the system parameters as requested in Sec. 4.3. In the modeling, a parameter independency is introduced by choosing p_1 and p_2. Hence, the current system (7.36) has three independent parameters: p_1, p_2 and d. However, both p_1 and p_2 depend on the same length l, which is not considered, when both parameters p_1 and p_2 are subdivided independently. To include this, l is chosen to be gridded, then the parameters are calculated and finally gridded again. An extension to this is obviously to introduce a further subdivision of the parameter ϕ, omitting the subdivision of p_1 and p_2 and hence, reduce the effort by one subdivision. This results in three different approaches, namely

(a) subdivision of p_1, p_2 and d
(b) subdivision of l, p_1, p_2 and d
(c) subdivision of ϕ, l, and d.

The variation of the positions of eigenvalues for the different approaches is shown in Fig. 7.10. Here, the effect of the different approaches becomes clear and it is obvious why Approach (c) is the least conservative routine. On an additional note, it avoids a transition between complex-conjugate and real eigenvalues. Now, to subdivide the parameters, one could choose an arbitrary number of subintervals, where an obvious, easy choice could be $L^{n_\mathrm{P}} = L^3$, or we make use of a sensitivity analysis as mentioned in Secs. 2.3 and 4.3. For the transformation of the controlled system, the first method is applied and all parameter intervals were subdivided with $L = L_\phi = L_L = L_d = 25$.

Consisting of a complex-conjugate pair of eigenvalues as well as two real ones, the resulting system matrix

$$\mathbf{N} = \begin{bmatrix} \sigma_1 & 0 & 0 & 0\\ 0 & \sigma_2 & 0 & 0\\ 0 & 0 & \sigma_3 & 0\\ 0 & 0 & 0 & \sigma_4\end{bmatrix} \tag{7.46}$$

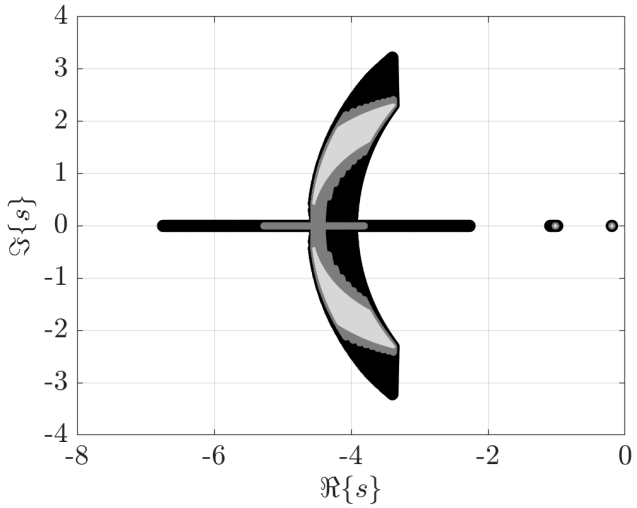


Figure 7.10: Distribution of eigenvalues of the controlled system (without the observer): (a) – black, (b) – dark gray and (c) – light gray.

with

$$\begin{aligned} \sigma_1 = \sigma_2 &\in [-4.321\ ;\ -3.155] \\ \sigma_3 &\in [-1.242\ ;\ -1.221] \\ \sigma_4 &\in [-0.194\ ;\ -0.192] \end{aligned} \tag{7.47}$$

is Hurwitz and Metzler.

7.2.4 Numerical results

For the calculation of the interval enclosures, it is furthermore important to chose the right hull over the L^3 small subintervals resulting from a gridding with Approach (c) with the least amount of overestimation as explained in Sec. 4.3.

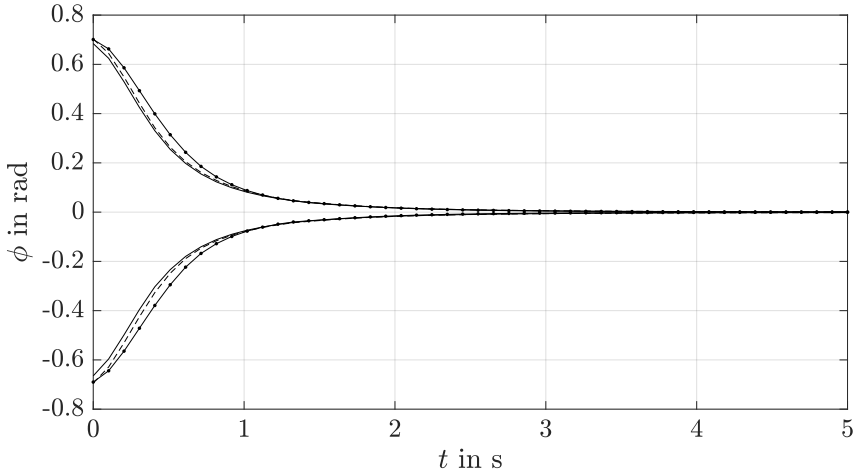


Figure 7.11: Interval enclosure for the state ϕ – approaches from Sec. 4.3.2: 1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]\}^{\langle\kappa\rangle}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]\}^{\langle\kappa\rangle}\right]\right]$ – dotted[5], 2. $\left[\tilde{\mathbf{T}}\right] = \bigcup_{\kappa=1}^{L} \left[\tilde{\mathbf{T}}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.

[5]Here, the linear interpolation is used for a better visualization.

Exemplary for the angle ϕ, Fig. 7.11 shows a comparison of all three methods for the uncertain initial states

$$\mathbf{x}(0) = \begin{bmatrix} [0.1\ ;\ 0.2]\,\mathrm{m} \\ [-0.1\ ;\ 0.1]\,\mathrm{rad} \\ [0.01\ ;\ 0.02]\,\mathrm{m}\cdot\mathrm{s}^{-1} \\ [0.05\ ;\ 0.08]\,\mathrm{rad}\cdot\mathrm{s}^{-1} \end{bmatrix}, \tag{7.48}$$

testing the robustness of the control design. A detailed view is given in Fig. 7.12, zoomed in on the starting phase.

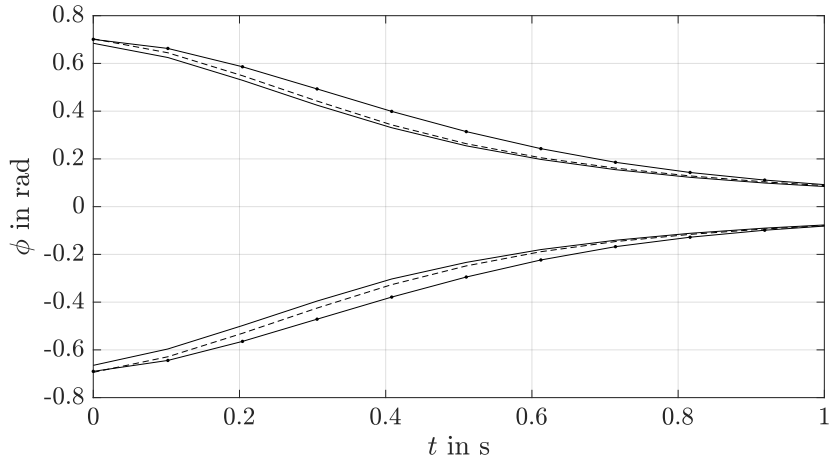


Figure 7.12: Zoomed in interval enclosure for the state ϕ – approaches from Sec. 4.3.2: 1. $\tilde{\mathbf{T}} = \left[\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\tilde{n}}\right]$ with $\tilde{\mathbf{T}}_j \in \left[\bigcup_{\kappa=1}^{L} \left[\Re\{[\mathbf{v}_{\lambda j}]\}^{\langle\kappa\rangle}\right], \bigcup_{\kappa=1}^{L} \left[\Im\{[\mathbf{v}_{\lambda j}]\}^{\langle\kappa\rangle}\right]\right]$ – dashed-dotted, 2. $\left[\tilde{\mathbf{T}}\right] = \bigcup_{\kappa=1}^{L} \left[\tilde{\mathbf{T}}^{\langle\kappa\rangle}\right]$ – dashed, 3. $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$ – solid.

For the presented case, there is not much difference in the resulting interval enclosures. However, Approach (3), which is included as a solid line in Figs. 7.11 and 7.12, is the least conservative. This can be explained by the semi-distributivity of interval-based set representations, where it becomes clear, that building hulls too early in the algorithm enhances the interval of the final result. On an additional note, if wide intervals are included it is difficult to compute the inverted transformation matrix. This means, that Approach (1) can be — as it is the case in the presented application scenario — impossible to apply. It is then recommendable to calculate the inverse transformation matrix for each subinterval and build the hull over those, cf. Eq. (4.49).

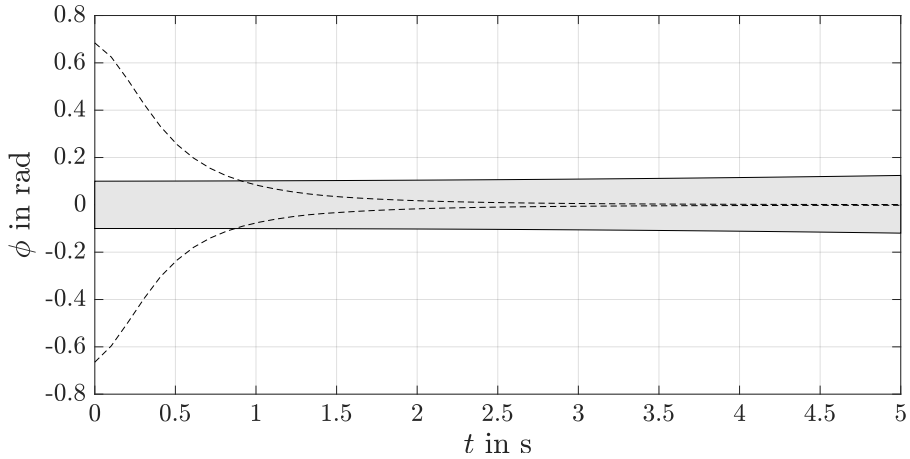


Figure 7.13: Interval enclosure for the state ϕ: comparison of Approach (3) from Sec. 4.3.2 and a state-of-the-art Taylor series expansion.

Finally, Fig. 7.13 shows a comparison of the presented method and a state-of-the-art Taylor series expansion. In contrast to the examples before, the presented cooperativity-based method does not start with a much too wide interval but still tightens when the operating point is reached. However, the Taylor series expansion continues to widen with increasing time due to the wrapping effect. In conclusion, solely using the cooperativity-based interval enclosure works for this specific example, but a combined method would be preferable again.

7.2.5 Observer

Here, the measured output is

$$\mathbf{C}_\mathrm{m} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} , \tag{7.49}$$

because the measured variables are the position of the moving carriage r and the rope angle ϕ. Again, the cooperativity-preserving method of Chapter 5 is difficult to apply. Hence, the LMI-based approach presented in Sec. 5.2 was used to calculate the observer gain before a transformation of the overall system. However, we will make use of another method in contrast to the rack feeder. For this, the observer gains are calculated for the *open loop*[6] according to $\mathbf{A}_\mathrm{O} = \mathbf{A} - \mathbf{H}\mathbf{C}_\mathrm{m}$ and result in

$$\mathbf{H} = \begin{bmatrix} 0.391 & 2.142 \\ -0.073 & 26.946 \\ 12.373 & -83.305 \\ -48.290 & 3.872 \cdot 10^3 \end{bmatrix} \tag{7.50}$$

using the same intervals as for the control design, see Eq. (7.37). Furthermore, a stability margin of $\gamma = 0.2$ was added. Repeating the stability proof from before for the widened angle interval, the new system matrix of the parallel model results in

$$\begin{aligned} \mathbf{A}_\mathrm{O} &= (\mathbf{A} - \mathbf{H}\mathbf{C}_\mathrm{m}) \\ &= \begin{bmatrix} -0.391 & -2.142 & 1 & 0 \\ 0.073 & -26.946 & 0 & 1 \\ -12.373 & 83.305 & -20.000 & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{aligned} \tag{7.51}$$

with

$$\begin{aligned} a_{41} &\in [48.290\ ;\ 48.291] \\ a_{42} &\in [-3.970\ ;\ -3.888] \cdot 10^3 \\ a_{43} &\in [21.612\ ;\ 200.000] \\ a_{44} &\in [-1.500\ ;\ -0.300]\ . \end{aligned} \tag{7.52}$$

7.2.6 Transformation of the parallel model

Again, the tighter intervals from Eq. (7.43) are used for the transformation. The observed system matrix then becomes

$$\mathbf{A}_\mathrm{O} = \begin{bmatrix} -0.391 & -2.142 & 1 & 0 \\ 0.073 & -26.946 & 0 & 1 \\ -12.373 & 83.305 & -20.000 & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{7.53}$$

[6] in contrast to the closed-loop approach from Sec. 7.1.5

with

$$\begin{aligned} a_{41} &\in [48.290\ ;\ 48.291] \\ a_{42} &\in [-3.897\ ;\ -3.891] \cdot 10^3 \\ a_{43} &\in [39.800\ ;\ 50.000] \\ a_{44} &\in [-1.200\ ;\ -0.500]\ . \end{aligned} \tag{7.54}$$

Since it is already known that the least conservative results are produced with Approach (c), this is the only subdivision applied for this transformation. Hence, the parameters ϕ, l, and d are subdivided accordingly. Note that in contrast to before, a sensitivity analysis was used to determine a suitable subdivision routine. It was shown that the eigenvalues were most sensitive to a change of parameter l. Hence, different distributions for each parameter were chosen accordingly, so that

$$\begin{aligned} L_\phi = L_d &= 10 \\ L_l &= 25 \end{aligned} \tag{7.55}$$

was applied resulting in $L = L_\phi \cdot L_l \cdot L_d = 2,500$ subintervals. In contrast to a straight implementation of $L^3 = 25^3 = 15,625$ subintervals, the simulation time was reduced drastically. Fig. 7.14 shows the variation of the positions of all eigenvalues for Approach (c).

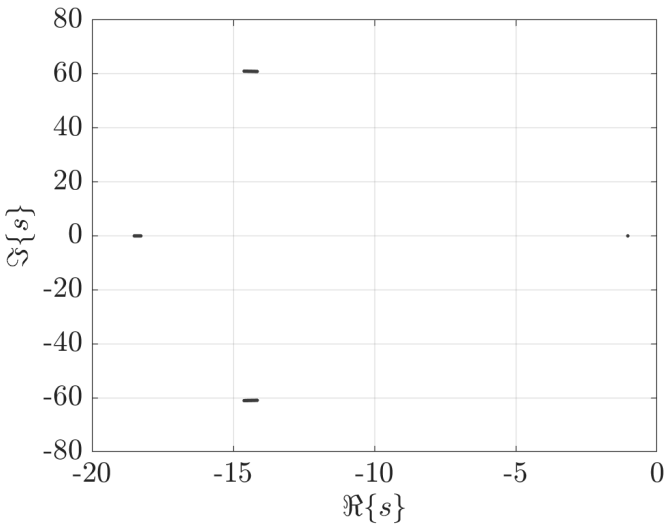


Figure 7.14: Distribution of eigenvalues of the overall system: Approach (c) – black.

As one can clearly see from the distribution of the eigenvalues, the system consists of two real eigenvalues and one conjugate-complex pair. Hence, the system matrix

$$\tilde{\mathbf{A}}_{\mathrm{O}} = \mathbf{N} = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 \\ 0 & 0 & 0 & \sigma_4 \end{bmatrix} \tag{7.56}$$

is obtained with

$$\begin{aligned} \sigma_1 = \sigma_2 &\in [-14.608\ ;\ -14.151] \\ \sigma_3 &\in [-18.501\ ;\ -18.270] \\ \sigma_4 &\in [-1.034\ ;\ -1.029] \end{aligned} \tag{7.57}$$

from the transformation described in Sec. 4.3 and is clearly Metzler and Hurwitz. Note, the complex-conjugate pairs result in equal real parts for σ_1 and σ_2.

7.2.7 Numerical results of the parallel model

In the next step, the parallel model is simulated to account for a comparison with the controlled system. The aim, here again, is to use the observer as a fault diagnosis tool for the controlled system. However — as already mentioned —, we will have a look into an approach differing from Sec. 7.1 to achieve this. Here, the parallel model is simulated with the measurements[7], so that[8]

$$\dot{\hat{\mathbf{x}}} = \left(\mathbf{A} - \mathbf{H}\mathbf{C}_\mathrm{m}\right) \cdot \hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{H}\mathbf{y}_\mathrm{m} \ , \tag{7.58}$$

cf. Eqs. (5.3) and (5.4). Since, the observer approach in (7.58) was done for the open loop, the controller must be added in another way. To account for this, the input is given by

$$\mathbf{u} = -\mathbf{k}^T\mathbf{x} \ , \tag{7.59}$$

while the result from Sec. 7.1.4 is used as $\mathbf{x} = \mathbf{x}_\mathrm{C}$ and

$$\mathbf{y}_\mathrm{m} = \mathbf{C}_\mathrm{m}\mathrm{mid}\left\{[\mathbf{x}_\mathrm{C}]\right\} + \Delta\mathbf{y}_\mathrm{m} \tag{7.60}$$

with

$$\Delta\mathbf{y}_\mathrm{m} \in \begin{bmatrix}[-1\cdot10^{-3}\ ;\ 1\cdot10^{-3}]\\ [-1\cdot10^{-3}\ ;\ 1\cdot10^{-3}]\end{bmatrix} \ . \tag{7.61}$$

Fig. 7.15 shows the resulting structure of the overall system. Note that once again, different outputs can be compared for a fault diagnosis.

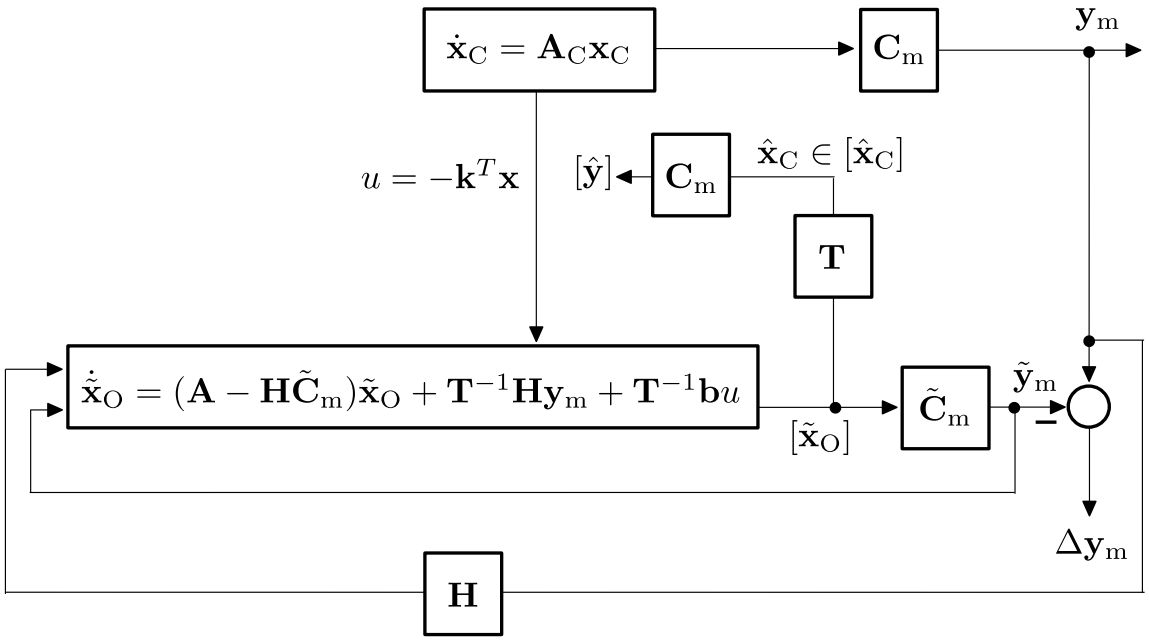


Figure 7.15: Use of the observer with a parallel model as a fault diagnosis tool.

However, for the presented scenario measurement errors shall be detected. For this, Fig. 7.16 shows the resulting interval enclosure for the observed angle ϕ. Note that the findings of Sec. 7.2.4 are applied and the parallel simulation will be done by using only the third Approach

[7] which are the results of the controller simulation in the presented case
[8] in original coordinates

with building the hull over the $L = 2500$ small subintervals with $[\mathbf{z}](0) = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right](0)$. Here, the uncertain initial states are chosen as in (7.48).

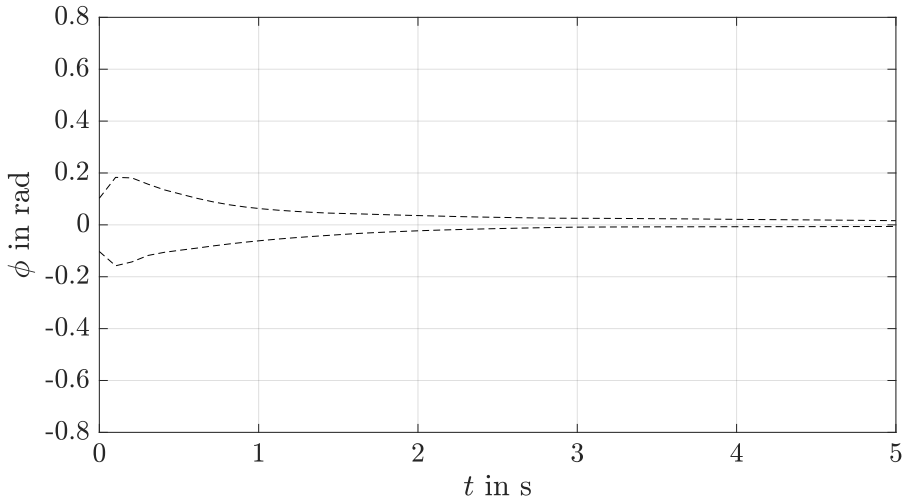


Figure 7.16: Interval enclosure of the parallel simulation for the state ϕ – approach 3: $[\mathbf{z}] = \bigcup_{\kappa=1}^{L} \left[\mathbf{z}^{\langle\kappa\rangle}\right]$.

Part III

Extensions

8 Comments on Fractional-Order Systems

Fractional-order models (FOM) were introduced because some systems cannot be described adequately by integer-order models due to their non-standard dynamical behavior, which can appear when the system is subject to e.g. long memory or hereditary effects, such as electrical conductance of biological systems or batteries. This also means that fractional-order systems (FOS) are different to integer-order systems in their properties. Hence, certain well-known methods can be applied likewise but others need to be adapted with exceptions and/or extensions to be equally valid for FOS. This section will discuss those peculiarities focusing on requirements for the proposed methods of this work and their resulting reformulations.
A Caputo definition is chosen because of its advantage that initial conditions have to be considered only for the state vector at $t = 0$ [41]. A fractional system model is given by

$$\mathbf{x}^{(\nu)} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{8.1}$$

with the state vector $\mathbf{x} \in \mathbb{R}^n$ and the control vector $\mathbf{u} \in \mathbb{R}^m$. Here, if the order ν is equal for all states, the system is called commensurate otherwise it is non-commensurate. A state-space representation with polytopic uncertainties can be formulated as

$$\mathbf{x}^{(\nu)} = \mathbf{A}(\mathbf{p}) \cdot \mathbf{x} + \mathbf{B}(\mathbf{p}) \cdot \mathbf{u} \tag{8.2}$$

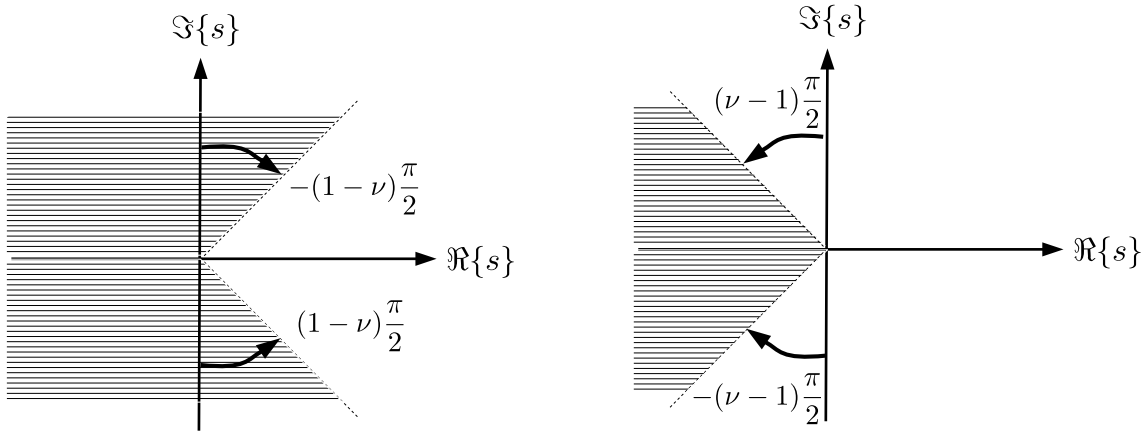
$$\mathbf{y} = \mathbf{C}(\mathbf{p}) \cdot \mathbf{x} + \mathbf{D}(\mathbf{p}) \cdot \mathbf{u}, \tag{8.3}$$

where $\mathbf{A}(\mathbf{p})$, $\mathbf{B}(\mathbf{p})$, $\mathbf{C}(\mathbf{p})$ and $\mathbf{D}(\mathbf{p})$ depend on uncertain parameters $\mathbf{p}$. Again, with implementing a feedback controller of the form $\mathbf{u} = -\mathbf{K}(\mathbf{x}) \cdot \mathbf{x}$ (analogously to Eq. (3.2)) with vanishing feed-forward action, the closed-loop system becomes asymptotically stable. As already mentioned, then, the desired steady-state operating point is assumed to be $\mathbf{x} = \mathbf{x}_\mathrm{s} = \mathbf{0}$ for $\mathbf{u} = \mathbf{u}_\mathrm{s} = \mathbf{0}$. When considering commensurate FOS — with equal order ν for all states — this is very similar to integer-order systems. However, it still leaves a difficulty regarding the stability region of FOS. This region differs not only from integer-order systems but also whether fractional orders of $1 < \nu < 2$ or $0 < \nu < 1$ are considered. For the latter, the stable region extends into the right half plane according to Fig. 8.1(a), but systems with fractional orders of $1 < \nu < 2$ have smaller regions in contrast to integer-order systems and correspond to the stability region in Fig. 8.1(b).

8.1 LMI-Based Robust Control

Since the stability region is not the same as for standard dynamical systems, the LMI-based control of Sec. 3.1 needs to be adapted. In [16], a suitable formulation was published, which shall be used in the following. Basically, the idea was to treat the stability domain as a union of two half planes, resulting from rotating the left half plane with angles of $\varphi = \pm(1 - \nu)\frac{\pi}{2}$ for the case of $0 < \nu < 1$ and with $\varphi = \pm(\nu - 1)\frac{\pi}{2}$ for $1 \leq \nu < 2$, respectively. Applying the polytopic model (3.13) for each of the vertices (3.15) including the vector of independent parameters, Eq. (3.11) and, hence, Eq. (3.24) are expressed as

$$(r\mathbf{X} + r^*\mathbf{X}^*)^T\mathbf{A}_\psi^T + \mathbf{A}_\psi(r\mathbf{X} + r^*\mathbf{X}^*) - \mathbf{Y}^T\mathbf{B}_\psi^T - \mathbf{B}_\psi\mathbf{Y} \prec 0 \tag{8.4}$$



(a) $0 < \nu < 1$ **(b)** $1 \leq \nu < 2$

Figure 8.1: Stability regions of a fractional-order system.

or including user-defined specifications of the Γ-region as

$$\mathbf{D}_0 \otimes (r\mathbf{X} + r^*\mathbf{X}^*) + \mathbf{D}_1 \otimes ((r\mathbf{X} + r^*\mathbf{X}^*)^T \mathbf{A}_\psi^T - \mathbf{Y}^T \mathbf{B}_\psi^T) + \mathbf{D}_1^T \otimes (\mathbf{A}_\psi (r\mathbf{X} + r^*\mathbf{X}^*) - \mathbf{B}_\psi \mathbf{Y}) \prec 0 \ . \quad (8.5)$$

Again, this needs to be evaluated simultaneously for all $\psi \in \{1, \ldots, n_\psi\}$. Note that the matrix $\mathbf{X} = \mathbf{X}^H$ is Hermitian which means that

$$\mathbf{X} = \mathbf{X}_R + j\mathbf{X}_I = \mathbf{X}_R^T - j\mathbf{X}_I^T \quad (8.6)$$

holds for $\mathbf{X} \succ 0$. Here, $\mathbf{X}_R$ and $\mathbf{X}_I$ represent the symmetric real and skew-symmetric imaginary parts, respectively, and

$$r = \begin{cases} e^{j(1-\nu)\frac{\pi}{2}} & \text{if } 0 < \nu < 1 \\ e^{j(\nu-1)\frac{\pi}{2}} & \text{if } 1 \leq \nu < 2 \end{cases} \quad (8.7)$$

as well as r^* and $\mathbf{X}^*$ as their respective conjugate-complex. Analogously to Eq. (3.10), the controller gain matrix is calculated by

$$\mathbf{K} = \mathbf{Y}(r\mathbf{X} + r^*\mathbf{X}^*)^{-1} \ . \quad (8.8)$$

The reader is referred to [16] for further information.

8.2 Application Scenario: Battery

As mentioned, a real-life application of fractional-order models is e.g. a battery system, which is analyzed in the following. The structure of the equivalent circuit representation of this battery system, which was derived based on [64], is shown in Fig. 8.2. Here, the base is given by a first-order *RC*-model, where the commonly used standard capacitor is replaced by a constant phase element (CPE, see C.2). Two energy storage elements describe the battery's dynamics, namely the dynamic behavior of the CPE voltage given by

$$u_{\mathrm{CPE}}^{(\nu)}(t) = -\frac{1}{RC} u_{\mathrm{CPE}}(t) + \frac{1}{C} i_0(t) \ , \quad (8.9)$$

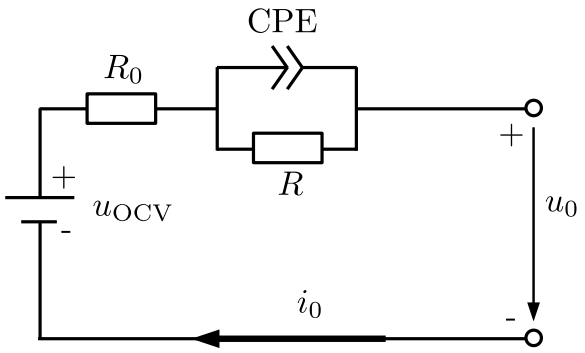


Figure 8.2: Battery structure with a constant phase element (CPE), the open-circuit voltage u_{OCV}, and the terminal current of the battery i_0.

where R and C are the element parameters and i_0 is the terminal current of the battery. Here, the fractional order is set to $\nu = 0.5$, which is close to the actual value identified by [64]. Secondly, the state of charge (SOC) σ can be described as the integer-order ODE

$$\dot{\sigma}(t) = -\frac{\eta}{Q_n} i_0(t) \tag{8.10}$$

with the charging/discharging efficiency η and the nominal battery capacity Q_n. This would lead to a state vector with two elements, i.e., the CPE voltage and the SOC. However, an incommensurate system would result with different differentiation orders. A remedy comes in terms of an implemented auxiliary relationship with $\sigma^{(0.5)} = \sigma_{\mathrm{f}}$, obtaining

$$\sigma^{(0.5)^{(0.5)}} = \sigma_{\mathrm{f}}^{(0.5)} = \dot{\sigma}. \tag{8.11}$$

As a general note, the number of auxiliary variables that have to be introduced to change an incommensurate system into a commensurate one relates to finding the greatest common divisor of all orders in the common fractional order. In the presented example, $\nu = 0.5$ already presents the greatest common divisor of the orders in Eqs. (8.9) and (8.10). Note that in this case $0 < \nu < 1$ holds, which means that we will refer to those specifications as given before, e.g. Fig. 8.1(a). The resulting state-space representation is

$$\mathbf{x}^{(0.5)} = \begin{bmatrix} -\frac{1}{RC} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} \frac{1}{C} \\ 0 \\ -\frac{\eta}{Q_n} \end{bmatrix} u = \mathbf{A}(p)\mathbf{x} + \mathbf{b}u \ , \tag{8.12}$$

with the state vector

$$\mathbf{x} = \begin{bmatrix} V_{\mathrm{CPE}} \\ \sigma \\ \sigma_{\mathrm{f}} \end{bmatrix} \tag{8.13}$$

and the system input $u = i_0$. The output equation is omitted here because it has no direct effect on the presented methods. The parameter values, which where identified by [64], are listed in Table 8.1. To additionally account for aging, an uncertainty is added in the Ohmic resistance of the CPE. In the given example, both transformation methods presented in Chapter 4 can be applied with a small modification regarding the control design. This will be explained in greater detail in the next subsections, before both simulation results are compared to each other.

Table 8.1: Parameters of the battery.

Variable	Unit/Value
C	615.93 F
η	0.74
Q_n	$18 \cdot 10^4$ C
R	$[0.51\ ;\ 1.51]\ \Omega$

8.2.1 Transformation on the basis of conjugate-complex eigenvalues

Applying the control design explained above, the controller gain vector

$$\mathbf{k}^T = \begin{bmatrix} 0.0067 & -0.0066 & -2.0008 \end{bmatrix} \cdot 10^4 \tag{8.14}$$

is calculated by Eq. (8.4) without any further requirements on the Γ-region, fulfilling the stability requirement of the position of eigenvalues complying to Fig. 8.1(a). Hence, the controlled system matrix becomes

$$[\mathbf{A}]_{\mathrm{C}} = \begin{bmatrix} [a_1] & 0.1077 & 32.4842 \\ 0 & 0 & 1 \\ 0.0003 & -0.0030 & -0.0823 \end{bmatrix} \tag{8.15}$$

with $[a_1] = [-0.1121;\ \ -0.1099]$. Its eigenvalues result in one real eigenvalue and a conjugate-complex pair. Incontrovertible, the transformation is done with the method presented in Sec. 4.3. After this transformation, the new system matrix results in

$$[\mathbf{N}] = \begin{bmatrix} [a_1] & 0 & 0 \\ 0 & [a_2] & 0 \\ 0 & 0 & [a_2] \end{bmatrix} \quad \text{with} \tag{8.16}$$

$$\begin{aligned} [a_1] &= [-0.1916;\ \ -0.1903] \\ [a_2] &= [-0.0014;\ \ -0.0009]\ , \end{aligned}$$

which is Metzler while still being asymptotically stable according to the design requirements (8.4). As mentioned, this approach can result in larger overestimation for the real eigenvalue. As a countermeasure, the specialized transformation for purely real eigenvalues shall be applied.

8.2.2 Transformation on the basis of real eigenvalues

Obviously, all eigenvalues need to be shifted into a real form to apply this transformation. A closer look at the system (8.12) reveals that the purely real eigenvalue is connected to the first state leaving only the other two to be shifted. As a first and simple choice, the respective gain for the first state is set to $k_1 = 0$ resulting in a controller gain

$$\mathbf{k}^T = \begin{bmatrix} 0 & -0.0066 & -2.0008 \end{bmatrix} \cdot 10^4 \tag{8.17}$$

of the output feedback law instead of Eq. (8.14). Of course, the stability has to be proven again for the changed, new controlled system

$$[\mathbf{A}]_{\mathrm{C}} = \begin{bmatrix} [a]_1 & 0.1077 & 32.4842 \\ 0 & 0 & 1 \\ 0 & -0.0030 & -0.0824 \end{bmatrix} \tag{8.18}$$

with $[a]_1 = [-0.0032; \quad -0.0010]$. This is done by reducing Eq. (8.4) to

$$(r\mathbf{X} + r^*\mathbf{X}^*)^T\mathbf{A}_\psi^T + \mathbf{A}_\psi(r\mathbf{X} + r^*\mathbf{X}^*) \prec 0 \ , \quad \mathbf{X} \succ 0 \ , \tag{8.19}$$

where the vertex matrices under application of the partial state feedback controller are denoted by $\mathbf{A}_\psi$. An analysis shows that the system is not only still asymptotically stable but also all eigenvalues are purely real. Hence, the approach of Sec. 4.2 is applied and results in the transformed system matrix

$$[\mathbf{N}] = \begin{bmatrix} [a]_1 & 0 & 0 \\ 0 & [a]_2 & 0 \\ 0 & 0 & [a]_3 \end{bmatrix} \quad \text{with} \tag{8.20}$$

$$\begin{aligned} [a]_1 &= [-0.0016; \quad -0.0016] \\ [a]_2 &= [-0.0003; \quad -0.0003] \\ [a]_3 &= [-0.0787; \quad -0.0787] \ . \end{aligned}$$

Noticeable, the interval widths, which are nearly given by point intervals with an interval width smaller than $1 \cdot 10^{-14}$, definitely undercut the larger ones for the first approach with mixed eigenvalues. This only highlights the conservatism of the first approach, which is visualized in the next section.

8.2.3 Simulation results

For this example, the simulations are only used to verify the presented approaches. Nevertheless, they are equally suitable for already mentioned tasks such as, e.g. state prediction in order to optimize the control or observer design by a simulative feasibility analysis or to perform an analysis of the sets of reachable states. A simulation of fractional-order systems is commonly done with a Grünwald-Letnikov definition, see [41], which is also the basis of the discretized implementation used in the presented case. Here, for $k = 0$

$$\mathbf{x}(k+1) = [\Delta T^\nu \mathbf{A} + \mathrm{diag}(\nu)\mathbf{I}]\,\mathbf{x}(k) + \Delta T^\nu\,\mathbf{B}u(k) \tag{8.21}$$

and for $k \geq 1$

$$\begin{aligned} \mathbf{x}(k+1) = &[\Delta T^\nu \mathbf{A} + \mathrm{diag}(\nu)\mathbf{I}]\,\mathbf{x}(k) \\ &- \sum_{i=2}^{N+1} (-1)^i \binom{\nu}{i} \mathbf{x}(k+1-i) + \Delta T^\nu\,\mathbf{B}u(k) \end{aligned} \tag{8.22}$$

holds. Still, ν is the fractional order, whereas the sampling time is given by ΔT and $\binom{\nu}{i}$ corresponds to the Newton binominal coefficient

$$\binom{\nu}{i} = \frac{\Gamma(\nu+1)}{\Gamma(i+1)\Gamma(\nu-i+1)} \tag{8.23}$$

generalized to real numbers with the Gamma function [38]

$$\Gamma(\nu) = \int_0^\infty \xi^{(\nu-1)} e^{-\xi}\, d\xi \ . \tag{8.24}$$

Any occurring discretization errors were not subject of this work and are, therefore, not further analyzed.[1] A comparison of both presented methods is done for the relevant states, the voltage over the CPE and the SOC. The initial state vector was set to $\mathbf{x}_0 = \begin{bmatrix} 0 & 0.75 & 0 \end{bmatrix}^T$ and a discharge process was enabled for 100 s to an SOC $\sigma = 0.4$.

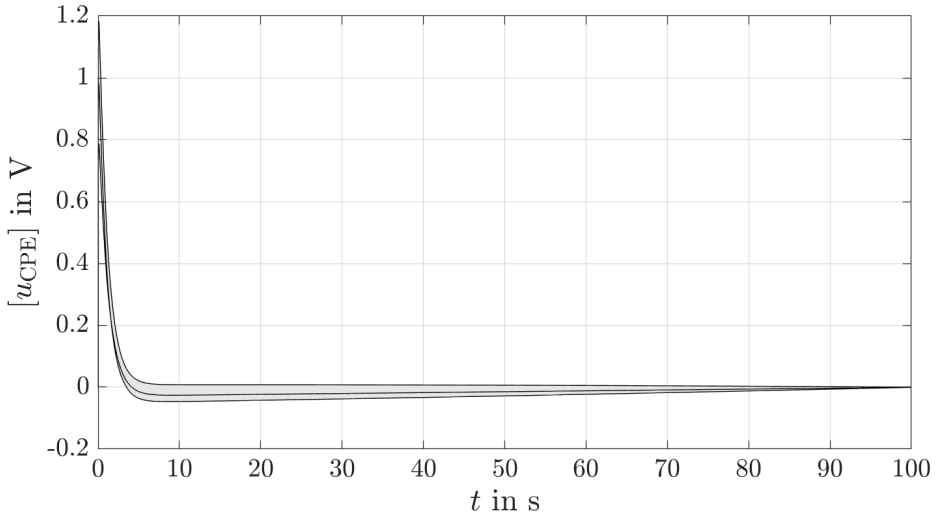


Figure 8.3: Simulation of the voltage over the constant phase element (CPE) for $t \in [0;\ 100]$ s — gray: the Jordan-form 4.3, black: transformation with purely real eigenvalues.

In Fig. 8.3, the first state, namely the voltage of the constant phase element is shown. The behavior for the system with purely real eigenvalues is given in black and results in a line coinciding with the nearly point interval-valued matrix with very small variations of the upper vs. lower bound of the interval. The Jordan form approach in Sec. 4.3 is given as a gray area. The same color code holds for the next Fig. 8.4, where the SOC is presented.

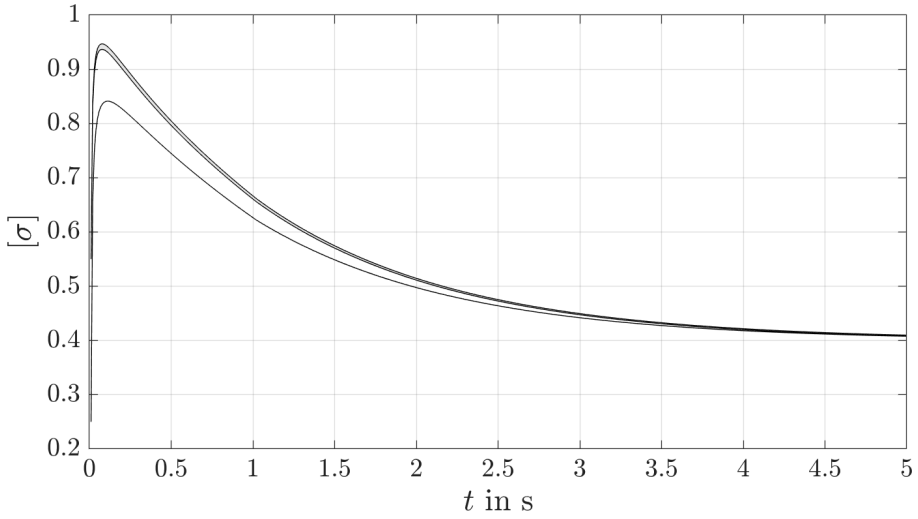


Figure 8.4: Simulation of the SOC σ, zoomed in for $t \in [0;\ 5]$ s — gray: the Jordan-form 4.3, black: transformation with purely real eigenvalues.

Here, the time horizon was limited to a few seconds in the beginning while it was also simulated for 100 s. This is done to highlight the differences of both approaches, because this would be

[1] For additional information, the reader is referred to [56,64] as well as to [34] and [45]. Here, those discretization effects were investigated on the basis of a verified interval analysis by employing a Picard iteration scheme directly on the fractional-order system.

hard to see in the overall time horizon since the intervals are very tiny.
Keeping in mind the different controller gains, the different outcomes are expected. One can see, that with (8.17) the system overshoots a little less than with (8.14). However, both stabilize the system at the desired point in a decent time horizon. Again, the transformation of purely real eigenvalue systems yields the smaller interval diameters.

This shows that with small modifications, the presented methods in this work for integer-order systems are equally applicable to fractional-order systems. The two main concerns, here, are to make the state-space system commensurate and adapt the stability region for the robust control and/or observer design. The simulation of a real-life application scenario validates these statements.

9 Alternative Computation of Interval Enclosures

As already mentioned in Chapter 4, the cooperativity-based calculation of interval enclosures is limited in its applicability. An obvious reason occurs, when the evaluation of Eq. (4.36) leads to excessively wide bounds for $\tilde{\mathbf{T}} \in \left[\tilde{\mathbf{T}}\right]$. For such too wide bounds, an interval-valued inverse $\left[\tilde{\mathbf{T}}\right]^{-1}$ of $\left[\tilde{\mathbf{T}}\right]$ does not exist or induces an excessive blow-up of the bounds due to the wrapping effect. This may happen after using the union over submatrices $\left[\tilde{\mathbf{T}}\right] = \bigcup_{\mathcal{I}} \left[\tilde{\mathbf{T}}_{\mathcal{I}}\right]$ resulting from a domain splitting to enclose the inverse $\tilde{\mathbf{T}}^{-1} \in \bigcup_{\mathcal{I}} \left[\tilde{\mathbf{T}}_{\mathcal{I}}^{-1}\right]$ with less overestimation. Additionally, a numerical instability in terms of a blow-up of interval enclosures may happen due to this conservative calculation of $\left[\tilde{\mathbf{T}}\right]$. Hence, an alternative to the computation of interval enclosures is given in the following based on findings of [27].

9.1 Exponential Interval Enclosure Technique

While an exponential state enclosure technique was already developed in [54], a combination with LMI-based approaches for robust control parameterization was added in [26]. Here, an exponential interval enclosure for the true solution $\mathbf{x}^*(t)$ to an IVP with $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^n$ and $t_0 = 0$ is defined as

$$\mathbf{x}^*(t) \in [\mathbf{x}_e](t) := \exp([\mathbf{\Lambda}] \cdot t) \cdot [\mathbf{x}_e](0) \ , \ [\mathbf{x}_e](0) = [\mathbf{x}_0] \tag{9.1}$$

with

$$[\mathbf{\Lambda}] := \mathrm{diag}[\lambda_i] \ , \ i = \{1, \ldots, n\} \ . \tag{9.2}$$

In this, the coefficients $\lambda_i \in \mathbb{R}$ introduced in Eqs. (9.1) and (9.2) are calculated using an iteration scheme in the context of an extended version of ValEncIA-IVP presented in [54, 55]. This scheme is based on a Picard iteration

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]^{(\kappa+1)} := [\mathbf{x}_e](0) + \int_0^t \mathbf{f}\left([\mathbf{x}_e]^{(\kappa)}(s)\right) \mathrm{d}s \ , \tag{9.3}$$

in which the exact solution is substituted by the exponential state enclosures (9.1). The resulting formula is differentiated with respect to time leading to

$$\dot{\mathbf{x}}^*(t) \in [\mathbf{\Lambda}]^{(\kappa+1)} \cdot \exp\left([\mathbf{\Lambda}]^{(\kappa+1)} \cdot t\right) \cdot [\mathbf{x}_e](0) = \mathbf{f}\left(\exp\left([\mathbf{\Lambda}]^{(\kappa)} \cdot t\right) \cdot [\mathbf{x}_e](0)\right) \tag{9.4}$$

as a fixed-point iteration scheme. To account for the complete time interval $t \in [t] = [0 \ ; \ T]$, the expression

$$\begin{aligned} \dot{\mathbf{x}}^*([t]) \in [\mathbf{\Lambda}]^{(\kappa+1)} \cdot \exp\left([\mathbf{\Lambda}]^{(\kappa+1)} \cdot [t]\right) \cdot [\mathbf{x}_e](0) \\ = \mathbf{f}\left(\exp\left([\mathbf{\Lambda}]^{(\kappa)} \cdot [t]\right) \cdot [\mathbf{x}_e](0)\right) \end{aligned} \tag{9.5}$$

replaces the evaluation of (9.4). If the iteration processes of Eqs. (9.4) and (9.5) converge,

$$[\lambda_i]^{(\kappa+1)} \subseteq [\lambda_i]^{(\kappa)} \quad \text{and} \quad [\mathbf{\Lambda}]^{(\kappa+1)} \subseteq [\mathbf{\Lambda}]^{(\kappa)} \tag{9.6}$$

as well as

$$\exp\Big([\mathbf{\Lambda}]^{(\kappa+1)}\,[t]\Big) \subseteq \exp\Big([\mathbf{\Lambda}]^{(\kappa)} \cdot [t]\Big) \tag{9.7}$$

hold. Following [54, 55], the iteration formula

$$[\lambda_i]^{(\kappa+1)} := \frac{f_i\left(\exp\left([\mathbf{\Lambda}]^{(\kappa)} \cdot [t]\right) \cdot [\mathbf{x}_e]\,(0)\right)}{\exp\left([\lambda_i]^{(\kappa)} \cdot [t]\right) \cdot [x_{e,i}]\,(0)}\ ,\ i \in \{1, \ldots, n\}, \tag{9.8}$$

is obtained for the interval parameter $[\lambda_i]$ of the desired state enclosure by applying further reformulations of Eq. (9.5) as well as the convergence properties in (9.6) and (9.7). However, the involved division only holds for $0 \notin [x_{e,i}]$, which is why the value 0 has to be excluded from the state enclosure or handled in another way in the computation of Eq. (9.8). Finally, the result of the iteration in (9.8)

$$\mathbf{x}^*(t) \in [\mathbf{x}_e]\,(t) := \exp\left([\mathbf{\Lambda}] \cdot T\right) \cdot [\mathbf{x}_e]\,(0) \tag{9.9}$$

is the solution of all reachable states at the end of the considered integration horizon $t = T$. Here, $[\mathbf{\Lambda}]$ is composed of the result obtained in the final iteration step. In [54], it is stated that requirements for a maximization of efficiency of the exponential enclosure approach include a domination by asymptotically stable, linear dynamics and a decoupling of the state equations. This means, that for systems with conjugate complex eigenvalues a simple transformation into *real* Jordan canonical form does not meet the latter requirement. Nevertheless, [27] proposes a solution in using the *complex* Jordan canonical form. The point being that a transformation of a point-valued realization embedded in the uncertain system model into this form maintains the decoupling properties approximately. This holds for both linear and nonlinear systems with uncertain parameters, which can be decoupled approximately if the matrix of the eigenvectors of the system's Jacobian, evaluated at the corresponding interval midpoints is used to perform the coordinate transformation before the application of Eq. (9.8). Note that the transformation has to be performed backwards into the original coordinates after the computation of the complex-valued state enclosures. In [26, 49, 54, 55] further information is given. Although the basic simulation routine according to Eq. (9.1)–(9.9) published in [54, 55] was done with a fixed, time- and state-independent integration step size, [26] extended the method by a simple step-size control strategy, guaranteeing numerical efficiency of the exponential enclosure technique. Said step-size control strategy determines the most appropriate step size $T = T_k$ according to

$$T_k = \max\left\{t_{\min}, \frac{1}{10} \cdot \min_{i \in \mathcal{I}^*}\left\{\inf\left\{\frac{-1}{\Re\left([\lambda_i]\right)}\right\}\right\}\right\}\ , \tag{9.10}$$

for $k \in \{2, 3, \ldots\}$, while the value T_1 is set to the user-defined value $T_1 = t_{\min}$. The time instants, where the controller should change, are calculated on the basis of the step size T_k from Eq. (9.10) by

$$i^* = \left\lceil \frac{L}{\zeta_{\text{end}}} \right\rceil\ , \tag{9.11}$$

where L is the number of all discretization steps from before and ζ_{end} is the number of user-defined controller steps, see Sec. 3.3.2. Furthermore, $\mathcal{I}^*$ denotes the index set for all states and respective parameter enclosures $i \in \{1, \ldots, n\}$ for which the relation $0 \notin [\lambda_i]$ holds. Here, $[\lambda_i]$

are in the decoupled linear case the enclosures of the eigenvalues of the system and generally correspond to the results of the iteration that was performed during the evaluation of the state enclosures according to (9.8) for the last temporal discretization slice $[t] = [0\ ;\ T_{k-1}]$. Here, the initial point of time of each slice is shifted to zero w.l.o.g. for time-invariant ODEs, cf. (9.8). The adaptation of the integration step size T results in a sequence $\{T_1, T_2, \ldots\}$. Instead of an equidistant grid $t_k = k \cdot T$ that would have to be adjusted to the fastest time constant for the complete simulation time horizon,

$$t_k = \sum_{j=1}^{k} T_j \tag{9.12}$$

determines the points of time $t = t_k$, where state enclosures are computed.

9.2 Application Scenario: Inverted Pendulum

For this method, the stabilization of an inverted pendulum in its upright position shall be the respective benchmark application as depicted in Fig. 9.1.

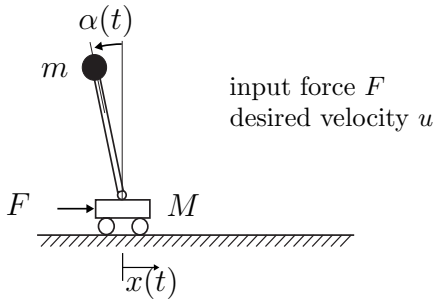


Figure 9.1: Control of an inverted pendulum on a moving carriage.

A pendulum of length $a = 0.2\,\mathrm{m}$ is mounted at the horizontal foot-point position x on a carriage of the mass M, which itself moves along a track. The deflection of the pendulum from its unstable upright equilibrium is denoted by the angle α. A massless rod with its mass m located in the tip of the pendulum describes the pendulum in good accuracy. Following this, the system can be described by two nonlinear autonomous second-order differential equations, namely

$$ma^2 \cdot \ddot{\alpha} - ma \cdot \cos(\alpha) \cdot \ddot{x} - mga \cdot \sin(\alpha) = 0, \tag{9.13}$$

with the gravitational acceleration $g = 9.81\,\frac{\mathrm{m}}{\mathrm{s}^2}$, and

$$(M + m) \cdot \ddot{x} - ma \cdot \cos(\alpha) \cdot \ddot{\alpha} + ma \cdot \sin(\alpha) \cdot \dot{\alpha} = F\ , \tag{9.14}$$

where F is the actuation force applied to the carriage into the positive direction of motion x. An underlying velocity control for the carriage is implemented in the form of a first-order lag behavior with the time constant $T_1 = 0.05\,\mathrm{s}$. In contrast to the high-bay rack feeder in Sec. 7.1, this describes the presented pendulum system well since reaction forces of the pendulum dynamics on the carriage can be neglected. Hence, the expression (9.14) can be replaced fully by

$$T_1 \cdot \ddot{x} + \dot{x} = u\ , \tag{9.15}$$

where u and $\dot{x}$ represent the desired and actual carriage velocities, respectively. A quasi-linear state-space representation

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{g \cdot \mathrm{si}(\alpha)}{a} & 0 & 0 & -\frac{\cos(\alpha)}{T_1 a} \\ 0 & 0 & 0 & -\frac{1}{T_1} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ \frac{\cos(\alpha)}{T_1 a} \\ \frac{1}{T_1} \end{bmatrix} u \ , \qquad (9.16)$$
$$y = \begin{bmatrix} -a \cdot \mathrm{si}(\alpha) & 1 & 0 & 0 \end{bmatrix} \mathbf{x} \ , \quad \mathrm{si}\,(\alpha) = \frac{\sin(\alpha)}{\alpha} \ ,$$

with the state vector $\mathbf{x} = \begin{bmatrix} \alpha & x & \dot{\alpha} & \dot{x} \end{bmatrix}^T$ and the system input u is used to express the overall system dynamics.

9.2.1 Control design

Similar to Sec. 7.2, two independent parameters

$$p_1 = \frac{g \cdot \mathrm{si}(\alpha)}{a} \quad \text{and} \quad p_2 = \frac{\cos(\alpha)}{T_1 \cdot a} \qquad (9.17)$$

are introduced to replace the matrix entries of (9.16) depending on the pendulum angle α, which also represents the uncertainty, again as a state dependency of the system. With this, a polytopic uncertainty representation (3.13) is obtained to find a convex combination of extremal system models. This results in a parameter-dependent system matrix and input vector according to

$$\mathbf{A}(\mathbf{p}) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ p_1 & 0 & 0 & -p_2 \\ 0 & 0 & 0 & -\frac{1}{T_1} \end{bmatrix} \quad \text{and} \quad \mathbf{b}(\mathbf{p}) = \begin{bmatrix} 0 \\ 0 \\ p_2 \\ \frac{1}{T_1} \end{bmatrix} \ . \qquad (9.18)$$

Initially, the control should hold for the interval $[\alpha] = \left[-\frac{\pi}{2} + \epsilon \ ; \ \frac{\pi}{2} - \epsilon\right]$ leading to the initial state interval

$$[\mathbf{x}](0) = \begin{bmatrix} [\alpha](0) \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (9.19)$$

with $[\alpha](0) = [\underline{\alpha}(0) \ ; \ \overline{\alpha}(0)]$. Note that for $\alpha = \frac{\pi}{2}$ as well as for $\alpha = -\frac{\pi}{2}$ controllability is lost, which is the reason why a small $\epsilon > 0$ is introduced.

In the following, both control procedures (i) a constant gain over the whole time horizon as well as (ii) a gain scheduling design over temporal subslices shall be implemented. Since both control designs rely on a tight enclosure of predicted state intervals, a verified computation of those is needed. However, for the presented example, a transformation into a cooperative form is not possible because of too wide intervals and a resulting non-invertible transformation matrix. Therefore, the exponential enclosure technique is used to determine the interval enclosures.

Approach 1: Constant gain with robustness over the whole time horizon

The result of applying the control design of Approach 1 presented in Sec. 3.3.1 is shown in Fig. 9.2, exemplary for the third run ($N = 3$).

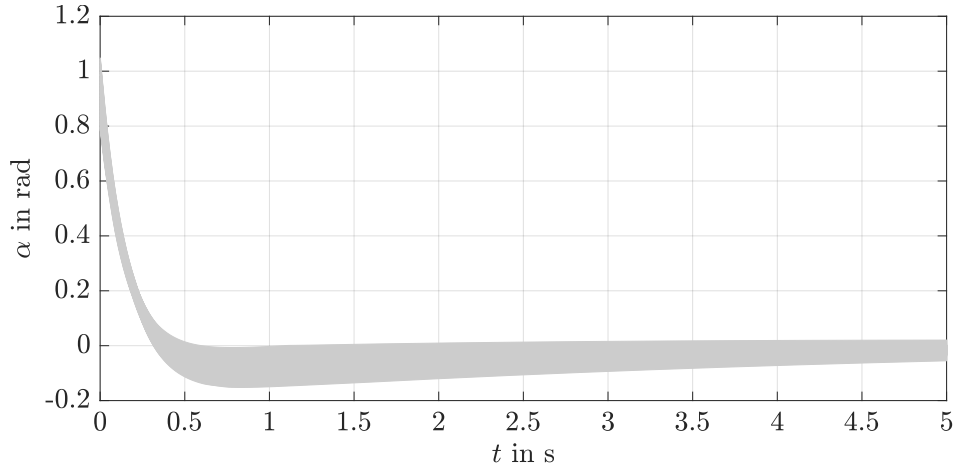


Figure 9.2: Interval enclosure for the pendulum angle α for all $t \in [0\ ;\ t_\mathrm{f}]$ — **Approach 1**.

The system's dynamics are not only successfully stabilized in its operating point by the controller, but also the interval diameter is decreasing once the operating point has been reached. Table 9.1 gives numerical results for the sequence of controller gains and the hull over the angle intervals for $t \in [0\ ;\ t_\mathrm{f}]$.

run i	K				$[\mathcal{X}^{(i)}]_1$	
	k_1	k_2	k_3	k_4	inf	sup
1	96.54707	15.3023677	-0.472897180	-4.99555847	-0.1521028	1.0481524366
2	94.68120	15.0066626	-0.463765567	-4.89774783	-0.1721871	1.0481472189
3	94.68119	15.0066613	-0.463765526	-4.89774740	-0.1721838	1.0481472163

Table 9.1: Simulation results for **Approach 1**: Controller gains and enclosure of the first state variable, where underlined digits highlight the values identical between two successive iterations.

The reader is reminded that this approach's main part is based on the repetition of controller gain calculations for the complete time horizon, see Fig. 3.6. The first run is, hence, done with the initial state interval resulting in the most conservative controller gains. The interval box describing the interval enclosure over the complete time horizon appearing in the first run $[\mathcal{X}^{(i)}]_1$ is used to calculate the second run's controller gains. Due to this less conservative interval box, the controller gains decrease, which also happens in the third run. For this scenario, a further reduction is not possible, which means that the optimal solution has been found.

Approach 2: Gain scheduling design over temporal subslices

The second approach, described in Sec. 3.3.2 is now analyzed, see Fig. 3.9. For this, Fig. 9.3 shows the results of the simulation. Here, the procedure also makes use of the step-size control strategy given in Eq. (9.10), where the calculated time slices are highlighted by dashed lines in Fig. 9.3.

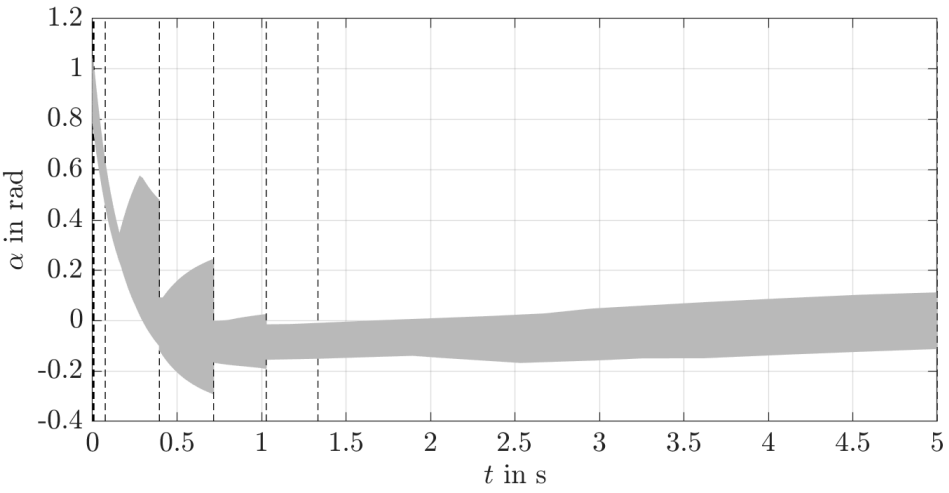
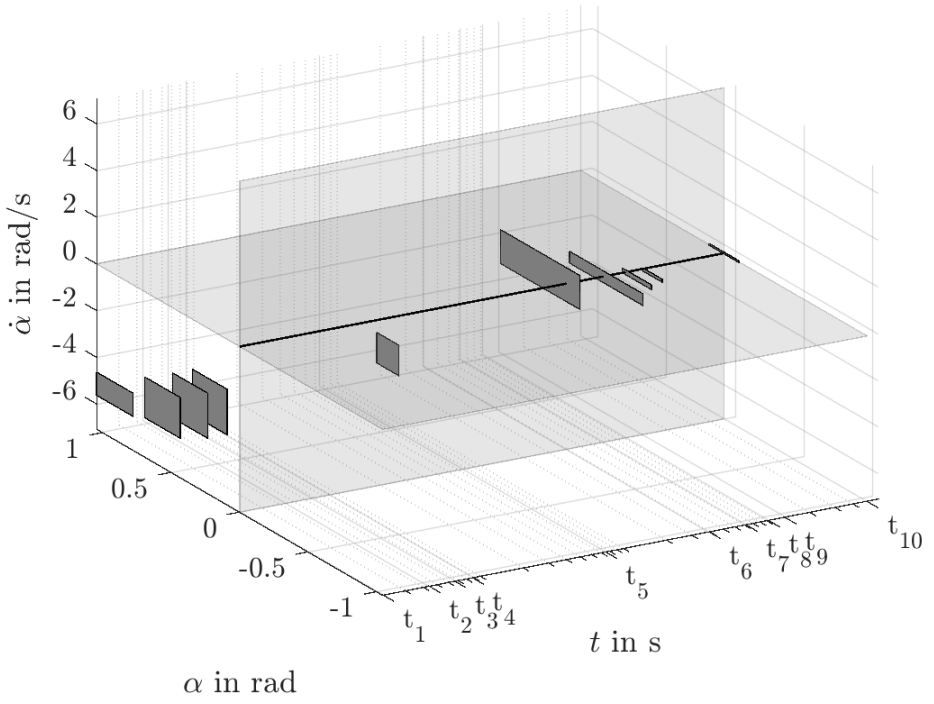


Figure 9.3: Interval enclosure for the pendulum angle α for all $t \in [0\ ;\ t_\mathrm{f}]$ — **Approach 2**.

A comparison with Fig. 9.2 confirms that the time instants of the step-size control strategy are generally well chosen. One can see, that up until approximately 1 s the hull over all possible state intervals is very wide. Hence, the control strategy divides the time slices in this section more often than afterwards, because the dynamics, here, is more stiff than close to the steady state. Since in the last time part after reaching the operating point, the state does not really change much, this section is not subdivided again. Unfortunately, **Approach 2** does not provide an improved control accuracy for the presented application scenario, which can be seen in Fig. 9.3 in the widened intervals in contrast to Fig. 9.2.
However, additional information can be received in terms of a simulation-based verification of the contraction property towards the asymptotically stable equilibrium. For this, Fig. 9.4 shows, when certain components of the state vector are mapped into themselves.



t_1	0.002
t_2	0.004
t_3	0.007
t_4	0.009
t_5	0.075
t_6	0.395
t_7	0.716
t_8	1.028
t_9	1.334
t_{10}	5.000

Figure 9.4: Approach 2: Interval boxes for α and $\dot{\alpha}$ for each predefined time step t_ζ (depicted on a logarithmic temporal axis).

Here, the interval boxes are depicted exemplarily for α and $\dot{\alpha}$ over the time horizon t_f. Obviously, due to the step-size control strategy, comparable enclosures are only available for the fixed time steps t_ζ, see Eqs. (9.10)–(9.11), which are the points of time where the controller gains change, see the table in Fig. 9.4. Those time steps t_ζ are featured logarithmically for simplified reading, because — as already mentioned — much denser time steps where needed for the beginning phase. Herein may also lie the problem of using this approach for the presented application scenario. For small time steps — as it is the case here —, the exponential enclosure is unnecessary conservative. This happens due to the involved transformation of the model into an approximately decoupled form and results in the large intervals, especially for the starting phase in Fig. 9.3. Still, the simulation shows a stabilizing behavior of the system. Additionally, interval enclosures are getting tighter by exploiting the step size selection according to Eq. (9.10) once the system gets close to the equilibrium.

10 Conclusions and Outlook

A lot of dynamical systems are subject to uncertainty whether it be willingly due to e.g. approximations made by the practitioner or be it unwillingly due to e.g. measurement noise. Either way, it has to be distinguished if those uncertainties have to be considered or if they can be retained in further approximations, e.g. if the control is robust enough. However, robust design can be cautious or too conservative, which is why it needs to be balanced with theoretical considerations and simulation tests. Nevertheless, there are systems, where experimental tests are costly, both economically and/or timely. Especially, for systems with big time constants, where the duration of experiments exceeds practicable solutions, it makes sense to calculate the dynamics beforehand. If this means including said uncertainties, those computations may be difficult.

10.1 Conclusions

In the presented thesis, uncertainties were considered to be bounded and, hence, given by intervals. Since computing the worst-case enclosures of the state intervals is a difficult task mainly because of the appearance of the wrapping effect which produces numerical overestimation, the main part of this thesis focused on finding solutions for that. It was noted that cooperativity may present a powerful tool to compute the worst-case bounds as two separate crisp parameter systems. However, since not all system models are naturally cooperative when derived by the first principle, a possible solution presents itself by finding a suitable transformation for these originally non-cooperative systems which fulfills the properties in another coordinate system. Literature on this is clear about the fact that possible approaches need to distinguish between systems with purely real eigenvalues and systems containing conjugate-complex ones, rendering the transformation matrix time-invariant or time-varying, respectively. The latter can be further differentiated into real-valued and complex-valued matrices. Each presenting a unique possibility to find the best transformation for a specific application scenario.
Additionally, this work presented robust control strategies. Basically, all controller designs make use of an LMI approach taking into account the worst-case bounds given by a polytopic model. Normally, this model includes worst-case bounds given by axis-aligned parameter boxes. However, methods to reduce the overestimation of these models were presented. A first control approach aims at reducing the control effort by subsequently decreasing the initial interval bounds. The idea stems from the fact that a physically possible state interval may be reduced if a control is applied and hence, the worst cases come closer together. A second control design can be seen as an extension to this, where controller gains were calculated not for the whole simulation horizon but for shorter time spans. In such a way, the control effort may be reduced resulting in tighter state intervals especially for the stationary phase.
Finally, an estimator was introduced to realize a fault diagnosis for the given systems. Here, it was assumed that measurements were fully available to the controller but a parallel model was implemented to check those measurements for faults. For this, two approaches were considered. At first, a cooperativity-preserving structure was introduced were the observer can be directly applied into the controlled and transformed system. A second observer design makes use of

the duality principle and is introduced as a parallel model computing possible state enclosures — again with a transformation into a cooperative system — which then can be used to detect system as well as measurement faults. However, the results of those observers could also be used in a closed-loop system like in [46].
All in all, the presented work considers those theoretical aspects and underlines the findings with suitable application scenarios. Note that those scenarios are chosen to be simple, academically systems to give the reader a number of different examples to validate the theoretical approaches. None of the given exemplary applications would be impossible to handle purely based on standard non-interval techniques, neither for cost decisions nor for safety reasons. However, they give a good overview on the possibilities and try to name a number of necessary adaptations for implementing the approaches on further real-life applications.

10.2 Outlook

In future work, the presented methods could be extended, e.g. to include a further reduction of overestimation in a polytopic model regarding complex systems, where a lot of uncertain parameters need to be taken into account. For this, [11] presented an approach, which is based on the assumption that the analyzed entries are related linear but an affine-linear approach will be very difficult due to the complexity of the system. The general idea is that there exists a coordinate system, where an axis-aligned box represents a tight enclosure of all possible combinations. The aim is, hence, to find this coordinate system, build the convex hull and then transform the whole system back into its original coordinates.
Another addition may be the already mentioned combination of the Taylor series expansion and the presented cooperativity-based method. This could help in reducing the overestimation especially of the starting phase of the cooperativity-based approach.
Finally, the presented methods could be used as a form of reachability analysis regarding both, the controller and the observer. With that, a realization as an optimal trajectory planer becomes possible and it could be further used as a parameter identification for uncertain systems. Additionally, the findings could be used to implement a predictive control reworking the methods in [44] with the new cooperativity-based approach.

A Mathematical Explanations

A.1 Calculation Rules for Partial Derivations

$$\frac{\partial \mathbf{A}}{\partial \mathbf{B}} = \sum_{ij} \mathbf{E}_{ij}^{(r\times s)} \otimes \frac{\partial \mathbf{A}}{\partial B_{ij}} \tag{A.1}$$

with $\mathbf{A} \in \mathbb{R}^{(n\times m)}$ and $\mathbf{B} \in \mathbb{R}^{(r\times s)}$, see [65].

A.2 Kronecker Product

Definition A.2.1 (Kronecker product)
For two matrices $\mathbf{A} \in \mathbb{R}^{m\times n}$ *and* $\mathbf{B} \in \mathbb{R}^{p\times q}$ *the direct (tensor) Kronecker product, written* $\mathbf{A} \otimes \mathbf{B}$*, is defined to be the partioned matrix*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots & a_{1n}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots & a_{2n}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & a_{m2}\mathbf{B} & \dots & a_{mn}\mathbf{B} \end{bmatrix} = [a_{ij}\mathbf{B}]_{i,j=1}^{m,n} \in \mathbb{R}^{mp\times nq} \ . \tag{A.2}$$

Basic calculation rules are

$$(\mathbf{A} \otimes \mathbf{B})\,(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD}) \tag{A.3}$$

$$(\mathbf{A} \otimes \mathbf{B})^T = \mathbf{A}^T \otimes \mathbf{B}^T \ . \tag{A.4}$$

For further information see [42].

A.3 Schur Complement

Definition A.3.1 (Schur complement)
Nonlinear (quadratic) matrix inequalities may be converted into an LMI form using the Schur complement, where

$$\mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^T \succ \mathbf{0} \qquad \textit{with} \quad \mathbf{C} \succ \mathbf{0} \tag{A.5}$$

can be reformulated into

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} \succ 0, \qquad \mathbf{A} = \mathbf{A}^T, \quad \mathbf{C} = \mathbf{C}^T \ . \tag{A.6}$$

A.4 Hadamard Product

Definition A.4.1 (Hadamard product)
For two matrices $\mathbf{A} \in \mathbb{R}^{m \times n}$ *and* $\mathbf{B} \in \mathbb{R}^{m \times n}$ *the Hadamard product is defined as*

$$\mathbf{A} \circ \mathbf{B} = (a_{ij} \cdot b_{ij}) = \begin{bmatrix} a_{11} \cdot b_{11} & a_{12} \cdot b_{12} & \dots & a_{1n} \cdot b_{1n} \\ a_{21} \cdot b_{21} & a_{22} \cdot b_{22} & \dots & a_{21} \cdot b_{21} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} \cdot b_{m1} & a_{m2} \cdot b_{m2} & \dots & a_{mn} \cdot b_{mn} \end{bmatrix} \in \mathbb{R}^{m \times n} \ . \tag{A.7}$$

Fundamental calculation rules are

$$\mathbf{A} \circ (\mathbf{B} \circ \mathbf{C}) = (\mathbf{A} \circ \mathbf{B}) \circ \mathbf{C} \tag{A.8}$$

$$a\,(\mathbf{A} \circ \mathbf{B}) = (a\mathbf{A}) \circ \mathbf{B} = \mathbf{A} \circ (a\mathbf{B}) \tag{A.9}$$

$$\mathbf{A} \circ \mathbf{B} = \mathbf{B} \circ \mathbf{A} \tag{A.10}$$

$$(\mathbf{A} + \mathbf{B}) \circ \mathbf{C} = \mathbf{A} \circ \mathbf{C} + \mathbf{B} \circ \mathbf{C} \quad \text{as well as} \quad \mathbf{A} \circ (\mathbf{B} + \mathbf{C}) = \mathbf{A} \circ \mathbf{B} + \mathbf{A} \circ \mathbf{C} \tag{A.11}$$

$$(\mathbf{A} \circ \mathbf{B})^T = \mathbf{A}^T \circ \mathbf{B}^T \ . \tag{A.12}$$

B Basics of Control Engineering

B.1 Kalman Controllability Criterion

Definition B.1.1 (Kalman controllability criterion)
The system $(\mathbf{A}, \boldsymbol{B})$ *of order* n *is fully controllable according to Kalman, if the matrix*

$$\mathbf{Q}_{\mathrm{C}} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} \tag{B.1}$$

has full rank, where

$$\operatorname{rank}(\mathbf{Q}_{\mathrm{C}}) = n \tag{B.2}$$

holds. For a single input system, it is sufficient to examine $\det(\mathbf{Q}_{\mathrm{C}}) \neq 0$ *for proving full controllability. Note that this also holds for* $\mathbf{B}(\mathbf{p})$ *and* $\mathbf{A}(\mathbf{p})$*, if* $\mathbf{p} = const.$

B.2 Kalman Observability Criterion

Definition B.2.1 (Kalman observability criterion)
The system $(\mathbf{A}, \boldsymbol{C})$ *of order* n *is controllable according to Kalman, if the matrix*

$$\mathbf{Q}_{\mathrm{O}} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{B.3}$$

has full rank, where

$$\operatorname{rank}(\mathbf{Q}_{\mathrm{O}}) = n \tag{B.4}$$

holds. For a single output system, it is sufficient to examine $\det(\mathbf{Q}_{\mathrm{O}}) \neq 0$ *for proving full observability. Note that this also holds for* $\mathbf{B}(\mathbf{p})$ *and* $\mathbf{A}(\mathbf{p})$*, if* $\mathbf{p} = const.$

C Physical Assumptions

C.1 Replacing an Inductivity by a Gyrator

It is sometimes desirable to implement high inductivities in electrical circuits with low voltages/currents/energies. However, it may be difficult to find such suitable inductivities. If that is the case, it is possible to replace a pure inductivity by a gyrator see [5], as shown in Fig. C.1.

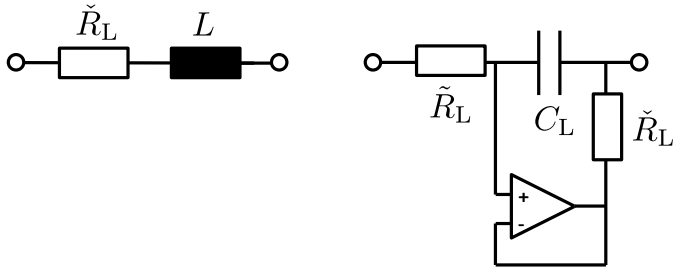


Figure C.1: Replacement of a real inductivity by a gyrator.

Here, the mathematical relation is given by

$$L = \tilde{R}_\mathrm{C} \cdot C_\mathrm{L} \cdot \check{R}_\mathrm{L} \ . \tag{C.1}$$

This means, that instead of an inductivity, two capacities and an Ohmic resistance are added into the electrical circuit.

C.2 Constant Phase Element

A constant phase element is an equivalent electrical circuit component, which models the behavior of an imperfect capacitor by a double layer. It's mathematical definition is given by

$$Z_Q = \frac{1}{Q_0(j\omega)^n} \ . \tag{C.2}$$

Here, an ideal capacity would be presented by setting $n = 1$ and $Q_0 = C$, whereas with a constant phase element $0 < n < 1$ holds. This is done in order to include, e.g. aging effects, rough surfaces, or a distribution of reaction rates. For more information see [8].

Bibliography

[1] Jürgen Ackermann. *Robust Control - The Parameter Space Approach*. Springer–Verlag, 2002.

[2] Harald Aschemann, Dominik Schindele, and Jöran Ritzke. State and Disturbance Estimation for Robust Control of Fast Flexible Rack Feeders. In A. Rauh and E. Auer, editors, *Modeling, Design, and Simulation of Systems with Uncertainties*, Mathematical Engineering, pages 333–351. Springer, Berlin, Heidelberg, 2011.

[3] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The Quickhull Algorithm for Convex Hulls. *ACM Transactions in Mathematical Software*, 22(4):469–483, December 1996.

[4] William Baumann and Wilson J. Rugh. Feedback Control of Nonlinear Systems by Extended Linearization. *IEEE Transactions on Automatic Control*, 31(1):40–46, 1986.

[5] Dale F. Berndt and Suhash C. Dutta Roy. Inductor Simulation Using a Single Unity Gain Amplifier. *IEEE Journal of Solid-State Circuits*, 4(3):161–162, 1969.

[6] Kurt Binder and Dieter Heermann. *Monte Carlo Simulation in Statistical Physics*. Springer–Verlag, Berlin, Heidelberg, 2010.

[7] Stephen P. Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[8] G.J. Brug, Ad L.G. van den Eeden, Margaretha Sluyters-Rehbach, and Jan H. Sluyters. The Analysis of Electrode Impedances Complicated by the Presence of a Constant Phase Element. *Journal of Electroanalytical Chemistry and Interfacial Electrochemistry*, 176(1):275–295, 1984.

[9] Wei Chen and Sang Hoon Lee. A Comparative Study of Uncertainty Propagation Methods for Black-Box-Type Problems. *Structural and Multidisciplinary Optimization*, 37(239), 2008.

[10] Claudio Cobelli, Ludwik Finkelstein, and Ewart R. Carsen. Mathematical Modelling of Endocrine and Metabolic Systems: Model Formulation, Identification and Validation. *Mathematics and Computers in Simulation*, 24(6):442–451, 1982.

[11] Noel Cont, Wiebke Frenkel, Julia Kersten, Andreas Rauh, and Harald Aschemann. Interval-Based Modeling of High-Temperature Fuel Cells for a Real-Time Control Implementation Under State Constraints. In *Proc. of 21st IFAC World Congress*, Berlin, Germany, 2020.

[12] Jb Copp. *The COST Simulation Benchmark — Description and Simulator Manual*. COST (European Cooperation in the Field of Scientific and Technical Research), Brussels, Belgium, 2001.

[13] Robert Dehnert. *Entwurf robuster Regler mit Ausgangsrückführung für zeitdiskrete Mehrgrößensysteme*. Springer–Vieweg, Wiesbaden, 2020.

[14] Yves Deville, Micha Janssen, and Pascal Van Hentenryck. Consistency Techniques for Ordinary Differential Equations. In *Proc. of the International Conference on Principles and Practice of Constraint Programming*, pages 162–176, Pisa, Italy, 1998. Springer–Verlag.

[15] Denis Efimov, Tarek Raïssi, Stanislav Chebotarev, and Ali Zolghadri. Interval State Observer for Nonlinear Time Varying Systems. *Automatica*, 49(1):200–205, 2013.

[16] Christoph Farges, Mathieu Moze, and Jocelyn Sabatier. Pseudo-State Feedback Stabilization of Commensurate Fractional-Order Systems. *Automatica*, 46(10):1730–1734, 2010.

[17] Thomas E. Fortmann and Konrad L. Hitz. *An Introduction to Linear Control Systems*. Marcel Dekker, Inc., New York, 1977.

[18] Mareile Freihold and Eberhard P. Hofer. Derivation of Physically Motivated Constraints for Efficient Interval Simulations Applied to the Analysis of Uncertain Dynamical Systems. *Applied Mathematics and Computer Science*, 19:485–499, 2009.

[19] Eberhard P. Hofer, Bernd Tibken, and Theodor M. Fliedner. Modern Control Theory as a Tool to Describe the Biomathematical Model of Granulocytopoiesis. In D P H Moller and O Richter, editors, *Analyse dynamischer Systeme in Medizin, Biologie, Okologie*, pages 33–39. Springer–Verlag, Berlin, Germany, 1991.

[20] Alberto Isidori. *Nonlinear Control Systems*. Springer–Verlag, 1995.

[21] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Éric Walter. *Applied Interval Analysis*. Springer–Verlag, London, 2001.

[22] Tadeusz Kaczorek. *Positive 1D and 2D Systems*. Springer–Verlag, London, 2002.

[23] Julia Kersten, Andreas Rauh, and Harald Aschemann. Interval Methods for the Implementation and Verification of Robust Gain Scheduling Controllers. In *Proc. of 22nd International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 791–796, Miedzyzdroje, Poland, 2017.

[24] Julia Kersten, Andreas Rauh, and Harald Aschemann. Interval Methods for Robust Gain Scheduling Controllers. *Granular Computing*, 2018.

[25] Julia Kersten, Andreas Rauh, and Harald Aschemann. State-Space Transformation of Uncertain Systems with Purely Real and Conjugate-Complex Eigenvalues Into a Cooperative Form. In *Proc. of 23rd International Conference on Methods and Models in Automation and Robotics (MMAR)*, Miedzyzdroje, Poland, 2018.

[26] Julia Kersten, Andreas Rauh, and Harald Aschemann. Application-Based Discussion of Verified Simulations of Interval Enclosure Techniques. In *Proc. of 24th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Miedzyzdroje, Poland, 2019.

[27] Julia Kersten, Andreas Rauh, and Harald Aschemann. Verified Interval Enclosure Techniques for Robust Gain Scheduling Controllers. *Acta Cybernetica*, 24(3):467–491, 2020.

[28] Wolfgang Kühn. Rigorous Error Bounds for the Initial Value Problem Based on Defect Estimation. Technical report, 1999. Available online: \url{http://www.decatur.de/personal/papers/defect.zip}.

[29] Joseph P. LaSalle and Solomon Lefschetz. *Stability by Liapunov's Direct Method with Applications*. Elsevier, New York/London, 1. edition, 1961.

[30] Johan Löfberg. YALMIP: A Toolbox for Modeling and Optimization in MATLAB. In *Proc. of IEEE International Symposium on Computer Aided Control Systems Design*, pages 284–289, Taipei, Taiwan, 2004.

[31] Rudolf J. Lohner. Enclosing the Solutions of Ordinary Initial and Boundary Value Problems. In E. W. Kaucher, U. W. Kulisch, and C. Ullrich, editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, pages 255–286, Stuttgart, 1987. Wiley-Teubner Series in Computer Science.

[32] Rudolf J. Lohner. On the Ubiquity of the Wrapping Effect in the Computation of Error

Bounds. In A Facius U. Kulisch R. Lohner, editor, *Perspectives on Enclosure Methods*, pages 201–216. Springer–Verlag, Vienna, 2001.

[33] David Luenberger. *Introduction to Dynamic Systems: Theory, Models, and Applications*. Wiley, New York, 1979.

[34] Rainey Lyons, Aghalaya S. Vatsala, and Ross A. Chiquet. Picard's Iterative Method for Caputo Fractional Differential Equations with Numerical Results. *Mathematics*, 5(4):65, 2017.

[35] Horacio J. Marquez. *Nonlinear Control Systems: Analysis and Design*. John Wiley, 2003.

[36] Günter Mayer. *Interval Analysis*. De Gruyter, Berlin, Boston, 1. edition, 2017.

[37] Frederic Mazenc and Olivier Bernard. Asymptotically Stable Interval Observers for Planar Systems With Complex Poles. *IEEE Transactions on Automatic Control*, 55(2):523–527, 2010.

[38] Kenneth S. Miller and Ross Bertram. *An Introduction to the Fractional Calculus and Fractional Differential Equations*. John Wiley, 1993.

[39] Nedialko Nedialkov. Interval Tools for ODEs and DAEs. In *CD-Proceedings˜of the 12th GAMM-IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics SCAN 2006*, Duisburg, Germany, 2007. IEEE Computer Society.

[40] Nedialko Nedialkov, K.R. Jackson, and John D. Pryce. An Effective High-Order Interval Method for Validating Existance and Uniqueness of the Solution of an IVP for an ODE. *Reliable Computing*, 7(6):449–465, 2001.

[41] Igor Podlubny. *Fractional Differential Equations*. Academic Press, 1998.

[42] Poznyak, Alex. *Advanced Mathematical Tools Vol.1*. Elsevier, Amsterdam, 1. edition, 2008.

[43] Tarek Raïssi, Denis Efimov, and Ali Zolghadri. Interval State Estimation for a Class of Nonlinear Systems. *IEEE Transactions on Automatic Control*, 57:260–265, 2012.

[44] Andreas Rauh, Ekatarina Auer, Ramona Westphal, and Harald Aschemann. Exponential Enclosure Techniques for the Computation of Guaranteed State Enclosures in ValEncIA-IVP. *Reliable Computing: Special volume devoted to material presented at SCAN 2012*, 19(1):66–90, 2013.

[45] Andreas Rauh and Julia Kersten. Toward the Development of Iteration Procedures for the Interval-Based Simulation of Fractional-Order Systems. *Acta Cybernetica*. under review.

[46] Andreas Rauh and Julia Kersten. From Verified Parameter Identification to the Design of Interval Observers and Cooperativity-Preserving Controllers — An Experimental Case Study. *Acta Cybernetica*, 24(3):509–537, 2020.

[47] Andreas Rauh, Julia Kersten, and Harald Aschemann. An Interval Approach for Parameter Identification and Observer Design of Spatially Distributed Heating Systems. *IFAC-PapersOnLine*, 51(2), 2018.

[48] Andreas Rauh, Julia Kersten, and Harald Aschemann. Linear Matrix Inequality Techniques for the Optimization of Interval Observers for Spatially Distributed Heating Systems. In *23rd International Conference on Methods Models in Automation Robotics (MMAR)*, pages 138–143, Miedzyzdroje, Poland, 2018.

[49] Andreas Rauh, Julia Kersten, and Harald Aschemann. Techniques for Verified Reachability Analysis of Quasi-Linear Continuous-Time Systems. In *24th International Conference on Methods Models in Automation Robotics (MMAR)*, pages 18–23, Miedzyzdroje, Poland, 2019.

[50] Andreas Rauh, Julia Kersten, and Harald Aschemann. Interval and Linear Matrix Inequality Techniques for Reliable Control of Linear Continuous-Time Cooperative Systems

with Applications to Heat Transfer. *International Journal of Control*, pages 1–18, 2020. Available online.

[51] Andreas Rauh, Julia Kersten, and Harald Aschemann. Transformation of Linear Systems with Real Eigenvalues into Cooperative Form: The Case of Constant and Time-Varying Bounded Parameters. In *25th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2020. under review.

[52] Andreas Rauh, Robert Prabel, and Harald Aschemann. Oscillation attenuation for crane payloads by controlling the rope length using extended linearization techniques. In *2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 307–312, 2017.

[53] Andreas Rauh, Jöran Ritzke, and Harald Aschemann. Error Estimates for Finite-Dimensional Approximations in Control of Distributed Parameter Systems. In *Progress in Industrial Mathematics at ECMI 2010*, pages 595–602, Wuppertal, Germany, 2010.

[54] Andreas Rauh, Ramona Westphal, and Harald Aschemann. Verified Simulations of Control Systems with Interval Parameters Using an Exponential State Enclosure Technique. In *Proc. of 18th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Miedzyzdroje, Poland, 2013.

[55] Andreas Rauh, Ramona Westphal, Harald Aschemann, and Ekatarina Auer. Exponential Enclosure Techniques for Initial Value Problems with Multiple Conjugate-Complex Eigenvalues. In M. Nehmeier, J. Wolff von Gudenberg, and W. Tucker, editors, *Scientific Computing, Computer Arithmetic, and Validated Numerics*, pages 247–256, Cham, 2016. Springer International Publishing.

[56] Margarita Rivero, Sergei V. Rogosin, José A. Tenreiro Machado, and Juan Trujillo. Stability of Fractional Order Systems. *Mathematical Problems in Engineering*, 2013.

[57] Siegfried M. Rump. IntLAB— INTerval LABoratory. In T. Csendes, editor, *Developments in Reliable Computing*, pages 77–104. Kluver Academic Publishers, 1999.

[58] Carsten W. Scherer. Lineare Matrixungleichungen in der Theorie der robusten Regelung. *at - Automatisierungstechnik*, 45(7):306–318, 1997. in German.

[59] Carsten W. Scherer and Siep Weiland. Linear Matrix Inequalities in Control. In W S Levine, editor, *Control System Advanced Methods*, The Electrical Engineering Handbook Series, pages 24–30. CRC Press, Boca Raton, 2nd edition, 2011.

[60] Ahmed A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, Cambridge, 2005.

[61] Hal L. Smith. *Monotone Dynamical Systems : an Introduction to the Theory of Competitive and Cooperative Systems*. American Mathematical Society, 1995.

[62] Jos F. Sturm. Using SeDuMi 1.02, A MATLAB Toolbox for Optimization over Symmetric Cones. *Optimization Methods and Software*, 11–12(1–4):625–653, 1999.

[63] Ulrich Tietze and Christoph Schenk. *Halbleiter-Schaltungstechnik*. Springer-Verlag Berlin Heidelberg, Berlin, 9. edition, 1989. in German.

[64] Baojin Wang, Zhiyuan Liu, Shengbo Eben Li, Scott Jason Moura, and Huei Peng. State-of-Charge Estimation for Lithium-Ion Batteries Based on a Nonlinear Fractional Model. *IEEE Transactions on Control Systems Technology*, 25(1):3–11, 2017.

[65] Alexander Weinmann. *Uncertain Models and Robust Control*. Springer Vienna, Vienna, 1991.