Redundancy Analysis and Structured Control Design for Dynamic Systems Using Set-Theoretical Methods

Vom Fachbereich
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

Philipp Leander Schaub, M.Sc.

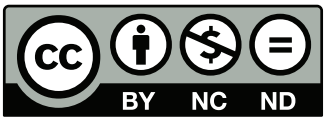
geboren in Darmstadt

Referent:	Prof. Dr.-Ing. Ulrich Konigorski
Korreferent:	Prof. Dr. Ing. Jürgen Adamy

Darmstadt 2024

Redundancy Analysis and Structured Control Design for Dynamic Systems Using Set-Theoretical Methods

Genehmigte Dissertation von Philipp Leander Schaub, M.Sc.
Darmstadt, Technische Universität Darmstadt
Tag der mündlichen Prüfung: 24. September 2024



Preface

The content of this thesis has been developed during my time at the Institute of Automatic Control and Mechatronics at the Technical University of Darmstadt, Germany. Although started at the dawn of the Covid-19 pandemic characterized by a significant reduction of social and scientific exchange, it was a time of exciting findings, not only on the scientific level, but also about myself and my attitude in view of mentally demanding situations. However, I will equally remember the cohesive team I was a part of, providing mutual support and a good working atmosphere. Our joint events, such as group retreats, after-work activities, or traveling around conferences were always a great pleasure. There is a group of outstanding people who have contributed to bringing this work to a successful conclusion and that I would like to mention. I sincerely thank my supervisor Prof. Dr.-Ing. U. Konigorski for his trust and his interest in my work. His words, intended or not, have always motivated and encouraged me in the most positive way. Prof. Dr.-Ing. J. Adamy who thankfully took the task of the second reviewer is the person encouraging me to pursue Ph.D. studies in the first place.

Fully aware of the fact that the following list is incomplete, I would like to explicitly mention and thank: P. Vogt and J. Hermann for the support during the work on my Master thesis which builds an important foundation for some of the results presented in this thesis; L. Herber for his dedicated cooperation on the field of LQRLC; F. Meiners for the periodic, deep, fruitful, and contentual discussions; A. Gross for many inspiring and exceptionally trustful conversations; C. Fischer and I. Brauer for keeping the group together and always fostering a good atmosphere; my colleagues from Office 510, especially J. Schaeffer and J. Pohlodek, who were the reason for preferring office over home-office; N. Hohmann, with whom I already went through the Master studies and whose ear I always had; my parents J. and R. Schaub who have always fully supported my educational path (and of course much more); my brother M. Maier, as well as R. Schurig, for the many mathematical discussions whose results are included at several points in this thesis; J. Andrae for the encouragement to always push the envelope[1]; my loving wife Isabel whose outstanding perservance has been a great inspiration for carrying on even in challenging times.

Darmstadt, October 2024
Philipp Schaub

[1] "Niemals dünne Bretter bohren!"

Contents

Symbols, Notation and Abbreviations

Latin Symbols

$\mathbf{0}_a$, $\mathbf{0}_{a\times b}$	a-dimensional vector or $(a \times b)$-dimensional matrix of zeros
$\mathbf{1}_a$	a-dimensional vector of ones
$\mathbf{A}$, $\bar{\mathbf{A}}$	continuous-time or discrete-time system matrix
$\mathbf{A}_\mathrm{F}$	closed-loop system matrix
$\mathbf{B}$, $\bar{\mathbf{B}}$	continuous-time or discrete-time input matrix
$\boldsymbol{c}_i^\top$	i-th row of output matrix $\mathbf{C}$
$\mathbf{C}$, $\bar{\mathbf{C}}$	continuous-time or discrete-time output matrix
$\mathbf{C}_\mathrm{m}$	measurement output matrix
$\mathbb{C}$	field of complex numbers
$\boldsymbol{d}_i^\top$	i-th row of feedthrough matrix $\mathbf{D}$
$\mathbf{D}$, $\bar{\mathbf{D}}$	continuous-time or discrete-time feedthrough matrix
$\mathbf{D}^*$	decoupling matrix
$\boldsymbol{e}$	error vector
$\boldsymbol{f}$	right side of ordinary differential equation
$\boldsymbol{f}_{\boldsymbol{x}}$	vectorized parameters of the state feedback matrix $\mathbf{F}_{\boldsymbol{x}}$, i.e. $\boldsymbol{f}_{\boldsymbol{x}} = \mathrm{vec}(\mathbf{F}_{\boldsymbol{x}})$
$\boldsymbol{f}_{\boldsymbol{y}}$	vectorized parameters of the output feedback matrix $\mathbf{F}_{\boldsymbol{y}}$, i.e. $\boldsymbol{f}_{\boldsymbol{y}} = \mathrm{vec}(\mathbf{F}_{\boldsymbol{y}})$
$\boldsymbol{f}_z$	right side of ordinary differential equation for fault scenario z
$\bar{\boldsymbol{f}}_{z,k}$	right side of discretized ordinary differential equation for fault scenario z
$\mathbf{F}_{\boldsymbol{x}}$	state feedback matrix
$\mathbf{F}_{\boldsymbol{y}}$	output feedback matrix
$g_{\mathrm{w}ij}$	element i, j of closed-loop transfer matrix $\mathbf{G}_\mathrm{w}$
$\boldsymbol{g}$	right side of continuous-time output equation
$\boldsymbol{g}_{\mathrm{w}j}$	column j of closed-loop transfer matrix $\mathbf{G}_\mathrm{w}$
$\bar{\boldsymbol{g}}_{z,k}$	right side of output equation of discrete-time system for fault scenario z
$\mathbf{G}$	open loop transfer matrix
$\mathbf{G}_\mathrm{w}$	closed-loop transfer matrix
$\mathbf{G}_\mathrm{w}^*$	decoupling structure of closed-loop transfer matrix
$\boldsymbol{h}$	right side of inequality in polytope $\mathcal{H}$-representation
$\mathbf{H}$	left side of inequality in polytope $\mathcal{H}$-representation
$\mathbf{I}_n$	$(n \times n)$-dimensional identity matrix
$\mathcal{I}_{a,b}$	index set $\{a, \ldots, b\}$ with $a, b \in \mathbb{Z}$ and $a < b$
J	objective function
$\mathbf{K}$	permutation matrix satisfying $\mathrm{vec}(\mathbf{M}^\top) = \mathbf{K}^{(m,n)}\,\mathrm{vec}(\mathbf{M})$ for some $\mathbf{M} \in \mathbb{R}^{m\times n}$
$\mathcal{K}$	cone
$\boldsymbol{\ell}$	vectorized parameters of prefilter matrix $\mathbf{L}$, i.e. $\boldsymbol{\ell} = \mathrm{vec}(\mathbf{L})$
$\mathbf{L}$	prefilter matrix

m	number of inputs
$\mathcal{M}$	index set of actuators considered in the redundancy analysis
n	number of states
$\mathbb{N}$	set of natural numbers
$\mathbb{N}_0$	set of natural numbers including zero
$\mathcal{O}$	orthant
p	number of outputs
P	characteristic polynomial
$\boldsymbol{p}$	vectorized parameters of Riccati matrix $\mathbf{P}$, i.e. $\boldsymbol{p} = \mathrm{vec}(\mathbf{P})$
$\mathbf{P}$	Riccati matrix or Rosenbrock system matrix
$\mathcal{P}$	preimage set
$\boldsymbol{q}$	vector containing degrees of freedom for solving a linear system of equations or polynomial variables
$\mathbf{Q}$	state weights for LQR
$\mathcal{Q}$	redundantly reachable set or solution set of polynomial inequalities
r	redundancy measure
$\mathbf{R}$	input weights for LQR
$\mathbb{R}$	field of real numbers
$\mathbb{R}_{\geq 0}$	set of positive real numbers including zero
$\mathcal{R}$	reachable set
s	Laplace variable
$\mathcal{S}$	set of feasible trajectories or conditioned invariant subspace or selection matrix
t	time
t_f	time of failure
T_s	sampling time
$\mathbf{T}$	transformation matrix or basis of a subspace
$\boldsymbol{u}$	control input
$\mathbf{U}$	matrix containing left singular vectors
$\mathcal{U}$	actuator constraint set
$\mathbf{V}$	matrix containing right singular vectors or basis of control invariant subspace
$\mathbf{V}_\Lambda^{-1}$	eigenvector matrix
$\mathcal{V}$	control invariant set or subspace
$\boldsymbol{w}$	reference input
$\mathcal{W}$	controllable subspace
$\boldsymbol{x}$	system state
$\mathcal{X}$	state constraint set
$\boldsymbol{y}$	system output
$\boldsymbol{y}_\mathrm{m}$	measurement output
$\boldsymbol{y}_\mathrm{ref}$	reference point in the output space
z	fault scenario
$\boldsymbol{z}$	constraint coefficient vector
$\mathbf{Z}$	constraint coefficient matrix
$\mathbb{Z}$	set of whole numbers

$\mathcal{Z}$ set of fault scenarios for redundancy analysis

Greek Symbols

$\boldsymbol{\alpha}_j$ canonical unit vector in j-th coordinate

γ Heaviside function: $\gamma(t) = \begin{cases} 0 & t < 0 \\ 1 & t \geq 0 \end{cases}$

Γ pole region

$\mathbf{\Gamma}$ design matrix for LQRLC

δ relative degree

$\boldsymbol{\theta}$ vector of uncertain parameters in a plant model

Θ domain of uncertain plant parameters

λ eigenvalue or parameter

$\mathbf{\Lambda}$ eigenvalue matrix

μ Lebesgue measure

π one-dimensional polynomial

$\boldsymbol{\pi}$ multi-dimensional polynomial

σ eigenvalue spectrum

$\mathbf{\Sigma}$ singular value matrix

$\mathbf{\Phi}$ time solution of plant model

Notation

$\mathbf{A}\mathcal{X}$ Given a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and a subset $\mathcal{X} \subseteq \mathbb{R}^n$, the product of $\mathbf{A}$ and $\mathcal{X}$ is defined as $\mathbf{A}\mathcal{X} = \{\mathbf{A}\boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{X}\}$.

Abbreviations

MIMO multiple-input-multiple-output

LQR linear quadratic regulator

LQRLC linear quadratic regulator with linear constraints

SISO single-input-single-output

Abstract

Automation advances further and further to fields in which highly complex and safety-critical system are operated autonomously. This is motivated by an enhanced quality of produced goods, an increase in profitability, or the demographic development in many industrialized countries. Systems must therefore be designed to operate autonomously, even under fault conditions. That is, faults must be diagnosed and controllers must be adapted automatically. This effort is, however, only fruitful if the underlying dynamic system is physically capable of maintaining a minimum level of operability in every possible fault scenario.

This thesis approaches this exact issue by developing a theoretic framework for analyzing dynamic systems for their redundancy properties. It is based on a newly formulated definition of redundancy merging intuitions with system theoretic terminology. Based on a reachability analysis, two types of redundantly reachable sets are introduced for redundancy analysis. Mathematical properties, applicability to system classes, and their computation are discussed in detail. A redundancy measure comprising two scalars is introduced for condensing possibly high dimensional reachability information in a manageable format. Conditions for redundant stabilizability of given reference points are formulated and verified. A respective algorithm for checking the conditions is presented.

A second contribution of the thesis is made in the field of structured control design. Imposing structural requirements on the closed-loop transfer behavior of multiple-input-multiple-output systems is necessary if certain signal paths from input to output must be decoupled from each other. Set theoretic methods are used to transform the structural design problem into equality constraints that must be satisfied by the controller and prefilter parameters of a time-invariant state feedback law. On the one hand, these constraints can be used to investigate the closed loop structure for robustness properties with respect to parameter uncertainties of the underlying plant. On the other hand, the approach enables a strictly structured method to design controllers establishing a desired closed loop structure. Combined with pole region assignment, an almost automatic structured control design is enabled.

In extension to the previous, the well-established method of linear regulator design (LQR) is investigated in the context of structured controllers. That is, the weighting matrices of the standard LQR are sought such that the resulting optimal controller fulfills the imposed equality constraints. The developed procedure makes use of results in the field of inverse optimal control and serves as an alternative for populating the degrees of freedom of structured state feedback controllers.

Zusammenfassung

Die Automatisierung dringt immer weiter in Felder vor, in denen hochkomplexe und sicherheitskritische Systeme autonom betrieben werden. Die Gründe hierfür liegen in einer verbesserten Produktqualität, einer Erhöhung der Profitabilität, oder aber der demographischen Entwicklung in vielen Industrienationen. Systeme müssen deshalb so entworfen werden, dass sie auch unter fehlerhaften Bedingungen autonom operieren können. Dazu müssen Fehler automatisch diagnostiziert werden. Ebenso müssen sich Regler automatisiert adaptieren. Dies kann jedoch nur dann erfolgreich sein, wenn das zugrundeliegende dynamische System physikalisch in der Lage ist, ein Mindestniveau an Operabilität in jedem möglichen Fehlerszenario aufrecht zu erhalten.

Diese Arbeit behandelt ebendieses Thema durch die Entwicklung eines theoretischen Rahmens für die Analyse dynamischer Systeme bezüglich ihrer Redundanzeigenschaften. Er basiert auf einer neu formulierten Definition des Redundanzbegriffs, die verbreitete Intuitionen zu diesem Begriff mit systemtheoretischen Betrachtungen in Einklang bringt. Basierend auf einer Erreichbarkeitsanalyse werden zwei Typen von redundanten Erreichbarkeitsmengen zur Redundanzanalyse eingeführt. Deren mathematische Eigenschaften, die Anwendbarkeit für verschiedene Systemklassen sowie deren Berechnung werden detailliert diskutiert. Ein Redundanzmaß, welches aus zwei Skalaren besteht, wird eingeführt. Dieses überführt die Information aus einer potenziell hochdimensionalen Erreichbarkeitsanalyse in ein übersichtliches Format. Schließlich werden noch Bedingungen für redundante Stabilisierbarkeit gegebener Referenzpunkte formuliert und bewiesen. Ein entsprechender Algorithmus zu deren Prüfung wird vorgestellt.

Ein zweiter Beitrag der Arbeit wird auf dem Feld des strukturierten Reglerentwurfs erbracht. Das Auferlegen von Strukturbeschränkungen auf das Übertragungsverhalten des geschlossenen Regelkreises von Mehrgrößensystemen ist dann erforderlich, wenn verschiedene Signalpfade zwischen Ein- und Ausgängen voneinander entkoppelt werden müssen. Mengentheoretische Methoden werden genutzt, um das Problem des Strukturentwurfs in Gleichungsbeschränkungen in den Regler- und Vorfilterparametern eines zeitinvarianten Zustandsrückführungsgesetzes zu überführen. Einerseits können diese verwendet werden, um die Übertragungsstruktur des geschlossenen Regelkreises auf ihre Robustheit bezüglich Parameterunsicherheiten der Regelstrecke zu untersuchen. Andererseits ermöglicht dieser Ansatz einen streng strukturierten Entwurf von Reglern, die eine gewünschte Übertragungsstruktur realisieren. Wenn er mit dem Verfahren der Polbereichsvorgabe kombiniert wird, entsteht ein nahezu automatisches Entwurfsverfahren strukturierter Regelungen.

Als Weiterführung der vorangegangenen Überlegungen wird der Entwurf linear-quadratischer Regelungen (LQR) im Kontext strukturierter Regler untersucht. Hierzu werden Gewichtungsmatrizen des Standard-LQR-Problems gesucht, die dazu führen, dass der resultierende optimale Regler die auferlegten Gleichungsbeschränkungen erfüllt. Die entwickelte Methodik greift auf Ergebnisse auf dem Gebiet der inversen Optimalsteuerungsprobleme zurück und dient als Alternative für die Belegung der Freiheitsgrade beim Entwurf strukturierter Zustandsrückführungen.

1 Introduction

In the second half of the 20th and the early 21st century, control engineering has played a significant role in developing assistive devices which are able to keep away many tasks from users. Automatic cruise control in motor vehicles or autopilot systems in aviation are well-known examples how such devices can relieve people not only at work, but also in their private life. By taking over tasks which have priorly bound the users' concentration and time, people can now focus on more important tasks, or, for private life, more comforting activities.

Automation of various tasks in private and professional life is, however, not only motivated by minimizing the amount of non-satisfactory activities. It furthermore has a major economic impact in the following fields:

Quality Using the example of an operator in a large chemical plant, automation helps to improve the quality of produced goods while minimizing the consumption of resources. An operator needs to oversee hundreds of quantities displayed in human machine interfaces. The data has to be absorbed and processed before the operator can deduce a sensible control action from it. Repeating the previously mentioned procedure, the human operator serves as a controller in the chemical control loop who suffers from some shortcomings. The process model, upon which control actions are taken, is merely qualitative, but not quantitative. Noisy measurement data might be hard to interpret. Significant outliers in the measurement data might thus be missed by the operator. This short and incomplete list of challenges faced by a human operator of a complex dynamic system motivates the use of an automated controller that is more powerful in processing data, predicting the future behavior of the plant and computing optimal control actions. This increases and perpetuates the quality of the produced goods.

Finance Improving production quality, and perhaps also quantity, through the use of automated control systems increases the overall value of products. This justifies higher prices which directly affects a company's revenue. Furthermore, the number of employees can be lowered, or, they can focus on more strategic tasks like future products and markets of the company. These items either lower the company's spendings or make them more efficient, hence, increase the revenue per invested money unit.

Demography Demography describes how populations of humans evolve over time [43, §101]. In many industrialized countries, the number of available professionals is decreasing because people have less children as the society's wealth increases. For Germany, it is expected that, in the twenties of the 21st century, the number of retiring professionals will exceed the number of young people entering the job market by 300000 each year [29]. This situation can of course not be

changed by increasing the amount of automatic devices. But it is one of the most important reasons why automation is truly needed. The plant operator mentioned in one of the prior paragraphs might simply not exist in the future and must be substitued by an autonomous control system.

Safety In modern society, safety is *the* key aspect of technical devices outweighing productivity and financial success. A technical system must be designed such that it does not put anyone or anything in danger. Again, automation can help improve safety properties of such systems by incorporating automatic fault detection units or fault-tolerant control strategies.

1.1 Motivation and Research Questions

The automatic handling of fully unforeseen events remains challenging. Establishing safety *towards* external entities and resilience *against* external influences increases system complexity and, therefore, development costs. Hence, safety aspects and financial aspects clearly contradict each other. This makes it desirable to analyze a system's safety properties as precisely as possible such that backup systems are designed as complex as necessary, but as simple and cost efficient as possible.

System design does not only include the physical construction, but also control design. Both disciplines need to work together in order to achieve an optimal result. This thesis aims at supporting such an *integrated* approach to system synthesis by studying the following two main research questions:

1. How can we assess the safety level of a given dynamic control system?
2. Can we identify robustness properties of a desired control structure with respect to uncertainties or fault scenarios?

The research fields that are relevant for this study are introduced and discussed in the following sections.

1.2 Redundancy

In a technical environment, the term *redundancy* describes "a part in a machine, system, etc., that has the same function as another part and that exists so that the entire machine, system, etc., will not fail if the main part fails" [24]. It is widely used in the language of control engineering and is often associated with the presence of backup systems that can keep a certain level of functioning in the event of a fault acting on the system. However, also other disciplines are familiar with this term and use it in a similar way. If the inactivation of genes does not or only marginally affect a biological phenotype, biologists speak of genetic redundancy [115]. Koren and Krishna [85] call

the ability to cope with errors that occur during transmission of data "information redundancy". As the frequency of extreme weather scenarios rises as a consequence of climate change, geographic redundancy, e.g., in telecommunication systems, is becoming more important. This term relates to the presence of duplicated server farms in different geographic areas [66].

Another term which is prominent in today's discussions is *resilience*. It describes "the ability to become strong, healthy, or successful again after something bad happens" [25].

In system design, it is essential, that the technical design goals are stated clearly and are aligned with the needs formulated by the stakeholders. Focusing on the definitions given above, we can interpret a wish for resilience of a system as a wish to recover from an unforeseen event and to return to its normal functions. This design goal goes beyond what could intuitively be interpreted as a wish for system safety. What is intuitively desired in automatic systems is that faults that may occur do not lead to an immediate failure of the system. Hence, the system must, in every case, be capable of reaching or maintaining a safe state of operation. In worst-case scenarios, there is no priority on the efficiency of the control system, like it is demanded of a resilient system. It is only required to keep away damage from humans, animals, and the environment.

The above discussion is the reason why we examine redundancy of dynamic systems in this thesis. The goal is to establish a systematic approach to revealing redundant structures in a given system dynamics. Classically, the design of a redundant control system requires the installation of backup systems. Depending on the severity of fault scenarios, systems are provided twice, thrice, or even more often, such that functioning of the system can be guaranteed with a sufficiently high probability. As already mentioned, designing systems in this way is costly. The aim of this thesis is to give an estimate from a control perspective on how urgent the need for such backup systems is.

1.2.1 Literature Overview

A large amount of research has been conducted in the field of fault-tolerant control which is clearly a field of major relevance when designing safe systems. We additionally state research results from the connected field of over-actuated control, which is inherently less focused on the occurrence of faults. A brief overview of existing system theoretic definitions for redundancy is presented.

Over-Actuated Control Developing design methods for so-called over-actuated systems originates mainly from the aviation sector (Buffington [26], Buffington and Enns [27], Buffington et al. [28], and Oppenheimer et al. [120]). Over-actuation is understood as the presence of degrees of freedom in the control variables that do not affect a given trajectory of the system. Important contributions on the choice of these degrees of freedom, the *control allocation*, have been given by Durham [45], Härkegård [59], Härkegård and Glad [60], Johansen [69], Johansen and Fossen [70], and Zaccarian [179]. The algorithms presented rely on the idea of allocating a desired force or moment which is calculated by a suitable control law to a given set of actuators. The simplest approaches use unconstrained optimization [70], whereas later results also consider input position and rate constraints [120]. A detailed compilation of the most important algorithms is given in the

survey [70]. The results have in common that they provide means to distribute the control action if there exist degrees of freedom. In this branch of research, it is, however, of reduced interest what this freedom implies for the system dynamics under the occurrence of fault scenarios.

Fault-Tolerant Control The degrees of freedom which are present when designing control strategies for over-actuated systems can be lost in the event of a fault scenario. Hence, researchers have studied how to reconfigure the controller in the presence of such situations. Oftentimes, the results are strictly linked to specific applications because the system architecture is explicitly exploited when designing the fault tolerant control scheme, see Osmic et al. [121] and Shaffer and Ross [152] who develop specific solutions for unmanned aerial vehicles. More generic approaches to the topic are given by Casavola and Garone [30] and Cristofaro et al. [39]. Results which take into account the system's reachability capabilities more explicitly include [22, 124, 126]. Paredis and Khosla [124] develop an algorithm such that the joint trajectories of a manipulator are chosen such that the system remains maneuverable in scenarios with erratically locked joints. Bouvier and Ornik [22] introduce the concept of resilient reachability. It enables the analysis of target states for their reachability under undesirable inputs. Paulson et al. [126] propose a model predictive control scheme incorporating path constraints that are based on reachable sets computed for all considered fault scenarios. This way, feasibility of the control scheme is preserved, even under fault conditions.

Redundancy Definitions When researching in the field of over-actuated control, it becomes apparent that there exist several definitions of what *over-actuation* or *redundancy* means (Johansen and Fossen [70] and Kreiss and Trégouët [89]). Serrani [151] relates over-actuation to the number of linearly independent inputs and outputs. Zaccarian [179] also uses kernel properties of the steady-state transfer matrix for assigning redundancy to the system. Galeani et al. [50] introduces the system's left-invertibility in the discussion about the presence of redundancy. As can be seen, there is no unique definition of redundancy for dynamic systems yet. Additionally, the works introduced all focus on linear time-invariant systems.

Open Questions As the above discussion shows, there are two main points which still need to be addressed, regarding the definition of redundancy. These are, finding a definition for redundancy that will be uniquely applied by researchers and which is seamlessly applicable to nonlinear systems. For the first point, it is necessary to map human intuition to a practical mathematical framework. A definition needs to be congruent with intuitions as well as being applicable in system theory. Secondly, applicability to nonlinear systems demands for concepts which are defined in both domains, linear and nonlinear. Hence, approaches to redundancy that fully rely on linear algebraic concepts are less promising than more general concepts like left-invertibility, see Galeani et al. [50].

1.3 Structured Control

When designing state feedback for linear MIMO plants (multiple-input-multiple-output), the number of degrees of freedom is significantly higher than for SISO plants (single-input-single-output). While for SISO plants, we can only assign the closed-loop dynamics using the state controller, we can additionally manipulate the system's structure in the MIMO case. We hereby mean the structure of the closed loop transfer matrix which is comprised of multiple transfer functions that characterize the transient behavior from one specific input or initial state to an output of the system. Often it is desirable to have certain *decoupling* properties such that manipulation of an input (or initial state) only affects a predefined subset of outputs – all other outputs should remain unchanged.

To achieve such decoupling structures in control design, special algorithms have been developed over the years and there exists a rich theory on how to design controllers enforcing such structures. Much attention has been payed to the robustness of controllers in the presence of parametric uncertainties of plants with respect to stability and performance. Stability is the most important property of a well-functioning control system. An automatic system must always be designed such that it can return to a stable equilibrium point that is desired and safe for all surrounding humans and objects. However, when speaking of performance, requirements are not formulated as concisely. This is because it depends on the specific application of the system, what *performance* exactly means.

In this thesis, we study the robustness of control structures with respect to parametric uncertainties. To this end, it is necessary to identify which parts of the control law are responsible for establishing the closed-loop structure and the closed loop dynamics. By separating these parts in the control design step, we can make statements about

- which plant parameters may change without changing the closed-loop structure,
- which control parameters are essential for establishing the desired control structure,
- which signal paths are not needed for establishing the control structure or stability.

Hence, the analysis can be used to specify which parts, i.e. parameters, of the control system must be kept free from uncertainty in order to maintain a desired transfer structure. Furthermore, it provides structural statements regarding the communication architecture. This means, that we can say which measurements and which control signals must be ensured to be transferred correctly in order to guarantee the desired plant operation. However, unnecessary communication paths can also be revealed, i.e., signal paths that do not contribute to the system structure.

The analysis tool developed in this thesis is constructive. Hence, the results may directly be used in a control design step. However, usage in the physical construction of the plant is also possible. As we will see, the nature of the design algorithm is applicable to the class of all state feedback control laws that satisfy linear equality constraints in the control parameters. Hence, we will not

limit ourselves to eigenstructure assignment in the latter, but also integrate output or decentralized control laws.

1.3.1 Literature Overview

Eigenstructure assignment has raised the interest of researchers in the 20th century, but there exists an ongoing stream of research. Early contributions include the famous publication by Falb and Wolovich [49] who were able to find necessary and sufficient conditions for the solvability of the diagonal decoupling problem. Furthermore, they introduced a synthesis formula for a decoupling control law which is simple to use and has therefore achieved great importance in the control community.

Another major field of eigenstructure assignment is disturbance decoupling. The control law is designed such that an additive disturbance acting on the system dynamics is fully decoupled from the output. Well-known results can be found, e.g., in Basile and Marro [15] and Basile and Marro [17]. Conte and Perdon [36] studied the robustness of such control approaches by searching a control law which decouples a whole family of parametric plants. Weiland and Willems [167] and Willems [171] study the case of unsolvable disturbance decoupling and propose approximate designs.

Less demanding forms of eigenstructure assignment in the sense of solvability, such as triangular decoupling, have been studied by Morse and Wonham [110] and Wonham [173]. Coupling control, which can be interpreted as a form of *block* triangular decoupling has been investigated by Konigorski [84]. The latter uses modal synthesis methods, as also introduced by Magni [99].

Recent results in the field are, e.g., given by Garone et al. [52] who state necessary and sufficient solvability conditions for a general eigenstructure assignment problem. It differs from the one investigated in this thesis because the number and type of closed-loop modes visible at each system output must be specified before the design step. The solvability conditions relate to this specific closed-loop configuration. Schmid et al. [149] robustifies pole placement by extending an eigenstructure assignment algorithm by Moore [109]. Results on obtaining a robust eigenstructure are presented by Ntogramatzidis and Schmid [118]. Here, robustness is interpreted w.r.t. the dynamic performance, i.e. the closed-loop eigenvalue locations, under the presence of a model-plant mismatch. Pusch and Ossmann [132] develop a framework for identifying control structures in a highly actuated system for enabling a desired eigenstructure assignment for the resulting closed loop.

The previously mentioned contributions focus on studying structural design itself, i.e. the design of the closed-loop transfer behavior. However, the combination of structured control design with optimal control has received much attention as well. Investigated system structures range from classical input output decoupling (Hirzinger [64], Kučera [90], and Wu and Lu [174]) and synchronizing control (Ko and Bitmead [82]) to output feedback (Chanekar et al. [31] and Zhou et al. [180]) and decentralized feedback (Lin et al. [93], Miller [107], Miller and Davison [108], Rotkowitz and Lall [143], and Swigart and Lall [155]) or a combination thereof. For the latter, see,

e.g., Lessard and Lall [92] who combine optimal control with output feedback in a decentralized setting.

Rotkowitz and Lall [143] study the design of decentralized controllers while minimizing some norm of the closed-loop transfer matrix, e.g. the $\mathcal{H}_2$- or $\mathcal{H}_\infty$-norm. They identify a quadratic invariance property of the constraint set which is imposed by the control structure and prove that the problem at hand can be solved via convex optimization if quadratic invariance is present. A related problem is considered by Lin et al. [93], where controllers with a block sparse structure are designed. An optimization scheme alternating between searching a suitable sparsity structure and populating this structure while minimizing a given system norm is proposed. Swigart and Lall [155] split the design of decentralized controllers establishing a minimum norm solution into multiple problems for which centralized controllers are designed.

Besides optimizing system norms in constrained control settings, the minimization of *signal* norms, as in the well-known linear quadratic control problem, has been subject to a large amount of research. Input output decoupling has been covered by Hirzinger [64] who adds terms to the objective function that are designed to establish a decoupling eigenstructure by making an approximate model matching approach. A crucial property of the results is the approximating nature which is especially advantageous if the system under consideration does not fulfill the necessary and sufficient solvability conditions for input output decoupling [49]. In this case, a controller can still be designed, that might show sufficient decoupling properties for the specific application. However, an exact decoupling controller will never be achieved due to the control design formulation.

The latter can be achieved by the design introduced by Wu and Lu [174]. Here, the optimal decoupling problem is decomposed into two parts. First, a standard decoupling control law is computed. Optimality is then achieved by performing linear quadratic design for the SISO plants that are obtained by the decoupling control law. This approach is very simple, but it suffers from the fact, that only the *virtual* control inputs and only output signals of the decoupled plants are penalized in the optimization. The internal dynamics, as well as the actual control inputs, are not optimized directly. The most significant drawback of this natural approach is, however, the restriction to diagonal decoupling.

Including state equality constraints into linear quadratic design, i.e. designing synchronizing controllers, is studied by Ko [81] and Ko and Bitmead [82]. The desired control structure results from geometric invariance constraints and the set of all synchronizing controllers can be expressed in an analytic form. Inserting this synchronizing controller in the system dynamics yields a transformed system that is fed into the optimization problem. Furthermore, two approximate design strategies are investigated by [82]. One is based on using penalty terms in the objective, similar to [64]. The other one applies projection of an unconstrained optimal controller to the constraint set.

Results for linear quadratic design in combination with output feedback have been obtained by, e.g., Chanekar et al. [31] and Zhou et al. [180]. Encoding this control structure in the objective function of the unconstrained optimization problem allows for obtaining a solution by solving an additional Lyapunov equation [180]. Chanekar et al. [31] obtained more general results suitable

for designing structured output feedback by including bilinear constraints into the optimization problem which has then to be solved numerically.

Open Questions As can be seen, the field of structured control has received a vast amount of attention during the last decades. However, the previous paragraphs show that there is still no universal solution to eigenstructure assignment with additional design constraints. The part of this thesis focusing on structured control tries to make a step towards generalizing eigenstructure assignment such that it can be easily combined with different requirements in control design, such as a decentralization of the control law or the restriction to using output signals only.

1.4 The Geometric Approach and Set-Based Control Theory

The Geometric Approach to control theory and modern set-based approaches to control will play a major role in the development of the results presented in Chapters 2 and 3. Hence, this section is devoted to the origins and the emergence of this elegant and powerful toolkit.

In the late 1950s, R. E. Kalman introduced the concept of a system *state* in the control community, see [73, 77, 78]. This simplified the analysis of dynamic systems because it offered a strictly unified description in state space representation, as we use it to this day. It enabled the development of equally unified synthesis tools for such systems. The reason for this progress was the applicability of new and well-understood mathematical methods to system theory, which was mainly linear algebra.

Shortly after the usage of the state space representation in control applications had consolidated, researchers identified the existence of certain subspaces of this state space with important properties, i.e. controllable and unobservable subspaces. Based on these findings, generic representations of the system dynamics were developed. Notable publications establishing the now well-known Kalman decomposition are Gilbert [53] and Kalman [74, 75]. Further work on results on controllability and observability was, e.g., published by Kreindler and Sarachik [88].

In 1969, the authors Basile and Marro [14] – and just shortly after, Wonham and Morse [172] in a slightly different manner – lifted the results that had been obtained so far to a next stage. In the mentioned publication, they introduced the concept of controlled and conditioned invariant subspaces to linear system theory. The most notable novelty of this contribution, besides the introduction of these important types of subspaces, was the fully coordinate-free approach to control theory. Instead of working with specific numerical matrices, the authors focused on their geometric properties, such as images and kernels, i.e. subspaces that are spanned by matrices, and their orthogonal complements. Relying on this natural geometric description of dynamic systems further generalized their representation w.r.t. the state space representation and, therefore, facilitated argumentation and the conduction of proofs. In the following years, a rich theory on linear control systems with many elegant results in analysis and design arose. These were compiled in

three widely used works by Basile and Marro [17], Trentelman [160], and Wonham [173] with a supplementary toolbox providing useful numerical algorithms in the geometric framework [101].

With the development of modern control methods such as model predictive control (MPC), researchers started using the classical concepts of, e.g., controlled invariants in a new setting. They come into play when observing stability properties of MPC schemes, e.g., when asking for a terminal constraint set to be a control invariant set [20, 134]. By considering state and input constraints of the systems, which is a natural step in optimization based controller design, the geometric approach had to be revisited for closed and bounded sets as opposed to linear spaces that are closed but not bounded. However, the *concepts* could be adopted. Hence, set-theoretic methods remain an important and powerful tool in control system theory. Large parts of the extensive theory on MPC that has been gathered can be reviewed, e.g., in Grüne and Pannek [58], Kouvaritakis and Cannon [86], Raković and Levine [136], and Rawlings et al. [139]. Toolboxes for numerical computations exist, e.g. Herceg et al. [62].

1.4.1 Literature Overview

After outlining the historical path that the geometric approach took in the previous section, we now want to highlight some interesting aspects and results in the geometric field.

Further general insights Control invariance was extended by a property called self-boundedness by Basile and Marro [16]. It characterizes subspaces contained in another control invariant subspace that cannot be left by any control law applicable to the plant. Extensions to this body of research has, e.g., been given by Ntogramatzidis [116] who extends the theory to systems that are not strictly proper. General results further supporting the geometric approach and application to tracking control are, e.g., given by Ntogramatzidis and Padula [117] and Padula et al. [123].

Robustness Researchers have put a focus on the robustification of the developed concepts quite early. Several studies have been conducted in this field and we will only state an incomplete list here. As an example, Basile and Marro [15] have introduced a robust version of control invariant subspaces. It is defined in view of plants which are subject to parametric uncertainties during operation. Its main property is that there exists a parametric linear state feedback controller that can render the subspace invariant. In this setting, the varying parameters of the plant need to be tracked precisely in order to reschedule the controller, i.e. a basic form of robustness. Conte et al. [37] provide results on how to compute this subspace. In the later publication [36], the authors approach the well-studied disturbance decoupling problem in a robustified manner. They aim at finding a subspace that stays control invariant for a set of parameters, i.e. for a family of plants, and a fixed, non-parametric state feedback. Results for families of plants have also obtained by Otsuka [122] who focuses on generalized and simultaneously control invariant subspaces and extends findings obtained by [19]. Recent results on robust geometric eigenstructure assignment have been obtained by Ntogramatzidis and Schmid [118, 119] who propose new computation

algorithms for output-nulling spaces – i.e. the analogue to control invariant subspaces for plants with feedthrough – and compatible controllers while focusing on optimizing numerical properties of the design steps.

Almost invariance The robustness properties of control invariants w.r.t. parameter uncertainties are often restrictive and many plants simply do not offer suitable robust control invariant subspaces. This means that no controller exists that can robustly keep a state trajectory inside this subspace once it has entered. The concept of *almost* invariance relaxes the requirement of keeping a trajectory *inside* the subspace but allows a controller also to keep a trajectory *arbitrarily close* to it. The idea was introduced by Willems [170] who continued his work in the publications [168, 169]. Results on an approximate version of the disturbance decoupling problem are given by Weiland and Willems [167] and Willems [171]. Trumpf [161] gives a modern and comprehensive overview over the existing material and provides advanced findings on the topology of such spaces.

Reachability Set-based approaches to reachability analysis stand out from the classical geometric ideas to some extent since they are a priori designed for estimating solution sets of linear or nonlinear dynamic systems subject to state and/or input constraints. However, the obtained reachable sets can be fed back to algorithms assessing control invariance or other properties. Much work in this field regarding theory and implementation has been done by Althoff [5], Althoff et al. [6], Althoff and Krogh [7], and Althoff et al. [8] who has as well established a toolbox for the use with Matlab [4]. Theoretical results on convexity of reachable sets have been obtained by Polyak [129], computational aspects of non-convex reachable sets for nonlinear systems are, e.g., covered by Kochdumper and Althoff [83].

Monotone systems, also known as positive systems, depict a system class that is well-suitable for safety verification because they admit simple over-approximations of their reachable sets. They were introduced by Angeli and Sontag [9] and have further been studied by Meyer et al. [105], Rantzer and Bernhardsson [138], Smith [154], and Valcher [162]. Relaxations to so-called mixed-monotone systems are discussed by Chu and Huang [34], Coogan [38], and Yang and Ozay [176].

Set-based system theory is still an ongoing research field as publications like Raković et al. [133] and Raković and Barić [135] show. Applications of set-based approaches in automation include, besides control design, fault detection and diagnosis where Savchenko [145] has delivered interesting results. By computing reachable sets, the nominal state of the plant can be falsified, thereby verifying a fault state of the system.

1.5 Contributions and Structure of this Dissertation

The body of research presented in this thesis is divided into three parts that constitute the content of Chapters 2-4.

In Chapter 2, the focus lies on developing an intuitive approach to the field of redundancy. Common to the priorly presented literature is the interpretation of redundancy as a *system* property. This approach has some downsides concerning intuition and mathematical rigor. A respective discussion is conducted and we propose to assign redundancy to *actuators* instead of the whole dynamic system. A definition of actuator redundancy is developed based on a set of feasible trajectories of the system. It is rigorously shown that, for linear systems without state or input constraints, the proposed definition merges with the definition given by Kreiss and Trégouët [89], which itself merges various results for linear systems. However, the new definition inherently extends to systems with linear dynamics with state and/or input constraints as well as general nonlinear dynamic systems. Therefore, this thesis contributes in closing a theoretical gap between over-actuated systems and systems with redundant actuators which have been treated equally in the literature so far.

After setting up a meaningful redundancy definition, a hands-on approach to the analysis of dynamic systems with respect to their redundancy properties is introduced. It is based on performing a reachability analysis for all fault scenarios that are considered in the analysis. We call their intersections redundantly reachable sets. Using them, worst-case statements can be made about which points in a given output space can be reached under all circumstances. This naturally connects the study of redundancy to the field of fault-tolerance. The computation of redundantly reachable sets can, depending on the underlying fault models, be infeasible because intersections over infinitely many reachable sets may occur. Simplified computation algorithms that do not cause any errors exist for linear systems with constraints and monotone systems with convex reachable sets. The respective proofs are developed and presented in the thesis.

Further contributions in the field of redundancy include the development of a measure indicating *how* redundant an actuator is, or alternatively, how fault-tolerant a set of outputs is w.r.t. a given fault. The measure is based on comparing generalized volume measures on the manifold of the redundantly reachable set. In this way, a meaningful measure attaining values between zero and one is obtained. Chapter 2 is concluded with the derivation of conditions for redundant stabilizability which are stated in terms of control invariance of redundantly reachable sets.

Chapters 3 and 4 cover the synthesis of structured linear control systems. In Chapter 3, we present an approach enabling the separation of the control design step into two parts, i.e. the assignment of the control loop structure, and the design of the dynamics. This is done by using the concepts of control invariant and conditioned invariant subspaces. For the desired control loop structure, i.e. the structure of the closed-loop transfer matrix, we identify subspaces that need to be *made* invariant by the feedback controller. These requirements can be transformed into linear equality constraints in the control parameters by using the idea of the Kalman decomposition [74]. The main advantage of this approach over other existing approaches, like the one chosen in, e.g., [82], is that all other design requirements that can be expressed via linear equations in the controller parameters, like a (partial) decentralization of the control law, can easily be integrated.

Using the above approach enables an investigation on how a system's eigenstructure depends on certain controller and plant parameters. Hence, we can derive statements on which plant param-

eters must be precisely known in order to keep the eigenstructure even despite variation of other parameters, and which ones only play a role for the dynamics but not for the eigenstructure. Furthermore, it can be deduced which controller parameters have to be secured against manipulation because they are crucial for establishing the desired system structure. Together with the results from Chapter 2, we obtain an extensive overview about system components that are essential for the system's desired operation.

Furthermore, we present a method to approximately assign a desired eigenstructure. This is important in cases where the system of equations encoding the desired structure is not solvable, i.e., this system structure cannot be assigned by using state feedback. The approximate method allows for obtaining a closed-loop structure that is somewhat close to the desired eigenstructure. The chapter is concluded by a discussion on the stabilizability of the plant under the constrained controller.

Chapter 4 puts a focus on the question how to populate a controller matrix that fulfills some equality constraints, i.e. an arbitrarily structured controller. Besides pole assignment, the priority of this chapter lies in parameterizing the structured controller by means of linear quadratic regulator design, which is a widely used design scheme for computing linear feedback laws. We interpret this problem as an inverse optimal control problem and identify a set of admissible weighting matrices in the cost functional that shift the optimizer of the optimal control problem to a control law satisfying the desired constraints. Doing this, the existing results on structurally constrained optimal control that are depicted in Section 1.3.1 are extended by a design algorithm that can handle arbitrary controller constraints.

2 Redundancy Analysis

In this chapter, we present an approach to methodologically analyze the presence of redundancy in a dynamic system Σ_0 of the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{2.1a}$$

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{2.1b}$$

$$\boldsymbol{x}(t) \in \mathcal{X}, \boldsymbol{u}(t) \in \mathcal{U}. \tag{2.1c}$$

Here, $\boldsymbol{x}(t) \in \mathcal{X} \subseteq \mathbb{R}^n$ denotes the state vector as an element of some bounded or unbounded subset $\mathcal{X}$ of the state space $\mathbb{R}^n$. The admissible control inputs are given by $\boldsymbol{u}(t) \in \mathcal{U} \subseteq \mathbb{R}^m$, where the set $\mathcal{U}$ may again be a bounded or unbounded subset of $\mathbb{R}^m$. The system output is given by $\boldsymbol{y}(t) \in \mathbb{R}^p$. We assume that $\boldsymbol{f}$ is locally Lipschitz, such that Eq. (2.1a) possesses a unique solution [3, Theorem 4]. Without loss of generality, let the origin be contained in $\mathcal{U}$, i.e. $\mathbf{0}_m \in \mathcal{U}$, where $\mathbf{0}_m = [0, \ldots, 0]^\top \in \mathbb{R}^m$.

As already discussed in Chapter 1, the term redundancy is used in different scientific fields with various interpretations. When speaking of technical applications, it is often associated with the presence of backup systems, that enable further operation of the system at hand after the occurrence of some fault. This is ensured by switching all control activities to a fully decoupled and functionally identical system that only exists for this purpose.

In this thesis, we are concerned with the question which redundancy structures exist naturally in a given dynamic system (2.1). This information can be used to obtain a better understanding whether a given set of requirements can be fulfilled in the case of some predefined fault scenarios. It, hence, plays an important role for making decisions on additional placements of actuators that can be used to ensure some minimally necessary function. However, it may also be used to increase cost efficiency by removing actuators or other system parts that do not contribute to system safety in a desired way. Furthermore, we will gather insights in regions of the state or output space that include only points that can be reached and stabilized in all considered fault scenarios.

2.1 Actuator Redundancy

2.1.1 Present Redundancy Definitions

In the control literature, the term *redundancy* is often used for unconstrained linear time-invariant systems of the form

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{2.2a}$$

$$\boldsymbol{y}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{u}(t) \tag{2.2b}$$

with, again, $\boldsymbol{x}(t)$, $\boldsymbol{u}(t)$, and $\boldsymbol{y}(t)$ as the state, input, and output vectors, respectively, and $\mathcal{X} = \mathbb{R}^n$, $\mathcal{U} = \mathbb{R}^m$. Such systems embody a special case of systems of the form (2.1). Different definitions of redundancy and over-actuation have been collected in a comprehensive paper by Kreiss and Trégouët [89], who analyze their similarities and differences. A brief summary of this discussion shall be given in the sequel.

In order to follow the arguments, let us first introduce the system properties of left- and right-invertibility:

Definition 1 (Left-invertibility [102]). *A system of the form* (2.2) *with* $\boldsymbol{x}(0) = \mathbf{0}_n$ *is called left-invertible if, given any admissible output function* $\boldsymbol{y} : [0;\, t_1] \to \mathbb{R}^p$, $t_1 > 0$, *there exists a unique corresponding input function* $\boldsymbol{u} : [0;\, t_1) \to \mathcal{U}$ *producing that output function* $\boldsymbol{y}$.

Definition 2 (Right-invertibility [102]). *A system of the form* (2.2) *with* $\boldsymbol{x}(0) = \mathbf{0}_n$ *is called right-invertible if there is an integer* $\rho \geq 1$ *such that, given any output function* $\boldsymbol{y} : [0;\, t_1] \to \mathbb{R}^p$, $t_1 > 0$, *with a piecewise continuous* ρ*-th derivative such that* $\boldsymbol{y}(0) = \mathbf{0}_p, \ldots, \boldsymbol{y}^{(\rho-1)}(0) = \mathbf{0}_p$, *there is at least one input function* $\boldsymbol{u} : [0;\, t_1) \to \mathcal{U}$ *producing that output function.*

Left-invertibility is closely related to the injectivity of a map. It asks for every feasible output function $\boldsymbol{y}$ to be produced by some *unique* input $\boldsymbol{u}$. Right-invertibility has similar properties as a surjective map. It essentially characterizes the ability of the system to reach every point $\boldsymbol{y}(t)$ in the output space by applying admissible control inputs $\boldsymbol{u}$.

Outlined already by Johansen and Fossen [70] and, similarly, by Härkegård and Glad [60], *input redundancy* is often characterized by the existence of a kernel of the system's input matrices, i.e.

$$\dim \ker \left(\begin{bmatrix} \mathbf{B} \\ \mathbf{D} \end{bmatrix} \right) > 0. \tag{2.3}$$

This means that there exist infinitely many input signals $\boldsymbol{u}$ that do not affect the state derivative and, therefore, neither its state, nor its output. This is a strong condition for redundancy because it is demanded that different input signals may not be visible in the system behavior at all. Hence, its naming *strong input redundancy* by Zaccarian [179]. A definition that is somewhat relaxed w.r.t. Eq. (2.3) is given, e.g., by Serrani [151], who calls a controllable, observable, right-invertible system (2.2) with $\mathbf{D} = \mathbf{0}$ *weakly input redundant* if

$$m \geq \operatorname{rank}(\mathbf{B}) > p. \tag{2.4}$$

This condition, which resembles the one used by Galeani et al. [50], implicitly demands for the existence of a kernel of the transfer matrix

$$\mathbf{G}(s) = \mathbf{C}\,(s\mathbf{I}_n - \mathbf{A})^{-1}\,\mathbf{B} + \mathbf{D}$$

with $\mathbf{I}_n$ as the n-dimensional identity matrix. The kernel exists if $\operatorname{rank}(\mathbf{B}) > p$. Hence, there exist control directions that do not influence the system output – but they may well act on the

system state. A different and even less demanding definition of weak input redundancy is given by Zaccarian [179] as

$$\dim \ker \left(\lim_{s \to 0} \mathbf{G}(s) \right) > 0. \tag{2.5}$$

Here, the presence of transmission zeros is tacitly demanded for $s = 0$. This means that there exist degrees of freedom in $\boldsymbol{u}$ for controlling the output $\boldsymbol{y}$ in steady state. Eq. (2.5) is weaker than Eq. (2.4) because it may be fulfilled although (2.4) is not. Recall that transmission zeros can occur for square systems with $m = p$, i.e. $m \neq p$. Vice versa, every system (2.2) with $\mathbf{D} = \mathbf{0}$ satisfying (2.4) also fulfills (2.5).

In their publication, Kreiss and Trégouët [89] collect and investigate the different definitions and streamline them into a new definition for input redundancy. Closely related to Eq. (2.4), they develop

Definition 3 (Input redundancy [89])**.** *A system of the form* (2.2) *is called input redundant if there exists an output* $\boldsymbol{y}$ *which can be produced by (at least) two distinct inputs for some* $\boldsymbol{x}_0 \in \mathcal{X}$.

Among others, they obtain the following results which we will make use of later:

Theorem 1 ([89, Th. 3.1])**.** *The following statements are equivalent:*

1. *System* (2.2) *is input redundant.*
2. *The transfer matrix* $\mathbf{G}(s)$ *is not left-invertible, i.e. there exists a non-zero polynomial vector* $\boldsymbol{q}(s)$ *such that* $\mathbf{G}(s)\boldsymbol{q}(s) = \mathbf{0}$ *for all* $s \in \mathbb{C}$.
3. *The Rosenbrock matrix pencil [142]*

$$\mathbf{P}(s) = \begin{bmatrix} s\mathbf{I}_n - \mathbf{A} & -\mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \tag{2.6}$$

is not left-invertible for all $s \in \mathbb{C}$.

The authors point out that Definition 3 is equivalent to the absence of left-invertibility for the system (2.2). This result is useful as it provides a definition which is applicable to a wider system class compared to the ones discussed in [50, 60, 70, 151, 179] which were introduced before. In contrast to the latter approaches that formulate input redundancy based on linear algebra, looking at left-invertibility is not restricted to linear systems. It can well be transferred to nonlinear systems of the form (2.1) by simply exchanging the system description in Definition 1 [102].

2.1.2 A New View on Redundancy

The aim of this thesis is to develop a way to map system theoretic intuition to a mathematical framework for analyzing redundancy properties of dynamic systems of the form (2.1). The definition of redundancy given in Chapter 1 asks a system to "not fail if the main part fails" [24].

It is to be defined by the system designer what needs to be considered as failure. The key point here is that redundancy is interpreted as some functionality of the system that *does not vanish* in the case of some subsystem's failure. Looking at Definition 3, a linear system is regarded input redundant as soon as there exists one output trajectory that is actuated in a non-unique way. These approaches are significantly different, and an input redundant system does not necessarily maintain the original functionality in the case of, e.g., an actuator loss, which we would intuitively assume.

An example that supports this argument is given by a steer-by-wire system as it is used in modern automobile vehicles, see e.g. [47]. Such a system may be composed of two coupled electric drives located at the steering rack and one decoupled motor creating a moment feedback to the steering wheel. The two lower drives are coupled for ensuring a minimal functioning in case of one actuator loss. The car could still be steered to a safe state. During normal operation of the plant, there exist degrees of freedom in distributing the control action between both actuators. This fact renders the nominal system input redundant w.r.t. Definition 3 since it is not left-invertible – there exists at least one output trajectory that can be produced by distinct input signals. However, this gives an incorrect impression of the redundancy properties of the *system*. While the two lower drives might show redundancy properties, the whole *system* clearly does not because the feedback drive cannot fail without a severe loss of functionality.

This motivates a discussion about which entity within a dynamic system should be associated with redundancy properties. Picking up the intuition of functionality of the system that does not vanish in case of a failure, it seems the most meaningful to assign this property to actuators u_i, where we denote u_i as the i-th component of $\boldsymbol{u}$. This way, every single system input is assigned its own redundancy properties, and it should only be called redundant if the system can still fully operate within a predefined operation region whenever this actuator fails. Here, the predefined operation region plays a crucial role, as also pointed out by Michellod [106]. This region could, e.g., be given by a minimally necessary operation range that ensures the system does "not fail if the main part fails".

Transferring this discussion to a mathematical definition, we first need to denote the predefined operation region that plays a fundamental role for our redundancy definition. We state

Definition 4 ([147, Def. 1])**.** *For a constrained dynamic system of the form* (2.1)*, we define the set $\mathcal{S}$ as the set of all tuples $(\boldsymbol{x}_0, \boldsymbol{y})$ that satisfy Eqs.* (2.1a) *and* (2.1b) *as well as the constraints $\boldsymbol{x}(t) \in \mathcal{X}$ and $\boldsymbol{u}(t) \in \mathcal{U}$ for all $t \geq 0$.*

With this, we can define redundancy w.r.t. an actuator failure. Describing the loss of the i-th actuator as $u_i = 0$, we introduce

Definition 5 (Actuator redundancy [147, Def. 2])**.** *Let a system of the form* (2.1) *possess the set of feasible trajectories $\mathcal{S}$ introduced in Definition 4. An actuator u_i is called redundant if $\mathcal{S}$ is not diminished in case of an actuator failure of u_i, i.e.*

$$\mathcal{S} = \mathcal{S}|_{u_i=0}\,.$$

The association of actuators with redundancy is indeed meaningful, but it is still closely related to the classical approaches summarized by Kreiss and Trégouët [89]. This is especially due to the simple fault model $u_i = 0$. To obtain a more general definition, we allow for *arbitrary fault scenarios* that can be included in the redundancy analysis. In this way, we obtain redundancy w.r.t. a fault scenario as opposed to redundancy w.r.t. an actuator. We identify a fault scenario with a dynamic system Σ_z of the form

$$\dot{\boldsymbol{x}}_z(t) = \boldsymbol{f}_z(\boldsymbol{x}_z(t), \boldsymbol{u}_z(t), t), \qquad \boldsymbol{x}_z(0) = \boldsymbol{x}_0 \tag{2.7a}$$

$$\boldsymbol{y}_z(t) = \boldsymbol{g}_z(\boldsymbol{x}_z(t), \boldsymbol{u}_z(t), t) \tag{2.7b}$$

$$\boldsymbol{x}_z(t) \in \mathcal{X}_z(t), \boldsymbol{u}_z(t) \in \mathcal{U}_z(t), \tag{2.7c}$$

where the state $\boldsymbol{x}_z(t)$ and input $\boldsymbol{u}_z(t)$ are defined on some subsets $\mathcal{X}_z(t) \subseteq \mathcal{X}$ and $\mathcal{U}_z(t) \subseteq \mathcal{U}$, respectively. We assume that the input, state, and output dimensions remain unchanged w.r.t. the nominal system Σ_0 defined in (2.1) to allow for a meaningful comparison. The explicit dependence of the quantities $\boldsymbol{f}_z$, $\boldsymbol{g}_z$, $\mathcal{X}_z$, $\mathcal{U}_z$ on the time t allows for, e.g., a switching between the nominal model Σ_0 and the dynamics defining the fault state at some time point of failure $t_\mathrm{f} \geq 0$. We will later make use of the short notation $\Sigma_z = (\boldsymbol{f}_z, \boldsymbol{g}_z, \mathcal{X}_z, \mathcal{U}_z)$. A set $\mathcal{Z}$ of fault scenarios comprises multiple system descriptions Σ_z, i.e., it has the form $\mathcal{Z} = \{\Sigma_{z1}, \Sigma_{z2}, ...\}$. In what follows, we will denote the elements of $\mathcal{Z}$ as z to keep the notation compact.

Example 1. Let us give a small example how such a fault scenario could be defined. Consider a spring damper system with $k_1, k_2, k_3 > 0$ as the nominal mass, damping coefficient, spring rate, respectively, and $F(t) \in \mathbb{R}$ as an externally applied force. The nominal dynamics of such a system, i.e. Eq. (2.1a), is then given by

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} 0 & 1 \\ -\frac{k_3}{k_1} & -\frac{k_2}{k_1} \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 \\ \frac{1}{k_1} \end{bmatrix} F(t).$$

Assuming a fault scenario in which the damping coefficient drops to zero at the time point $t_\mathrm{f} > 0$, e.g., due to a sudden oil leakage in the damper, can be modeled by Eq. (2.7a) as

$$\dot{\boldsymbol{x}}_z(t) = \begin{bmatrix} 0 & 1 \\ -\frac{k_3}{k_1} & -\frac{k_2}{k_1}\gamma(t_\mathrm{f} - t) \end{bmatrix} \boldsymbol{x}_z(t) + \begin{bmatrix} 0 \\ \frac{1}{k_1} \end{bmatrix} F(t)$$

with γ as the Heaviside function. Further examples where a fault scenario is defined by a reduction of the set of admissible controls $\mathcal{U}$ are given later in this chapter by Eqs. (2.12) and (2.14).

As an extension to Definition 5, we arrive at

Definition 6. *Let a system of the form* (2.1) *possess the set of feasible trajectories $\mathcal{S}$ introduced in Definition 4. The system is called redundant w.r.t. a set $\mathcal{Z}$ of fault scenarios z if $\mathcal{S}$ is not diminished in case of their occurrence, i.e.*

$$\mathcal{S} = \mathcal{S}|_z \qquad \forall\, z \in \mathcal{Z},$$

where we denote the respective sets of feasible trajectories of the fault scenarios $z \in \mathcal{Z}$ by $\mathcal{S}|_z$.

2.1.3 Comparison of Actuator Redundancy and Input Redundancy

As a result from the search for an intuitive approach to redundancy, we have stated a new definition for actuator redundancy. A generalized version of this definition is given by Definition 6. It is more abstract but allows for a more flexible inclusion of fault scenarios. As, however, Definition 5 is the natural extension to Definition 3, we will compare the properties of actuator redundancy to input redundancy defined by Kreiss and Trégouët [89]. We show that linear, input redundant systems (2.2) possess at least one redundant actuator. The converse holds as well – the existence of a redundant actuator renders the system input redundant. Conducting the same study for nonlinear systems (2.1) reveals that non-left-invertibility, i.e. input redundancy, *does not* induce the existence of redundant actuators.

2.1.3.1 Linear Systems

The system behavior of a linear system (2.2) is described by the Rosenbrock matrix pencil (2.6) as

$$\underbrace{\begin{bmatrix} s\mathbf{I}_n - \mathbf{A} & -\mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}}_{=\mathbf{P}(s)} \begin{bmatrix} \boldsymbol{x}_\mathrm{s}(s) \\ \boldsymbol{u}_\mathrm{s}(s) \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{y}_\mathrm{s}(s) \end{bmatrix},$$

where we denote by the quantities $\boldsymbol{x}_\mathrm{s}$, $\boldsymbol{u}_\mathrm{s}$, and $\boldsymbol{y}_\mathrm{s}$ the Laplace transformed state, input, and output signals, respectively, whose existence we assume. For every feasible tuple $(\boldsymbol{x}_0,\ \boldsymbol{y}) \in \mathcal{S}$, there exist corresponding state and input trajectories such that the above equation can be inverted. This yields

$$\begin{bmatrix} \boldsymbol{x}_\mathrm{s}(s) \\ \boldsymbol{u}_\mathrm{s}(s) \end{bmatrix} = \mathbf{P}^{+}(s) \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{y}_\mathrm{s}(s) \end{bmatrix} + \mathbf{P}^{\perp}(s)\boldsymbol{q}(s) \tag{2.8a}$$

$$=: \mathbf{P}^{+}(s) \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{y}_\mathrm{s}(s) \end{bmatrix} + \begin{bmatrix} \mathbf{P}_1^{\perp}(s) \\ \mathbf{P}_2^{\perp}(s) \end{bmatrix} \boldsymbol{q}(s), \tag{2.8b}$$

where we denote the Moore-Penrose pseudo inverse of $\mathbf{P}(s)$ by $\mathbf{P}^{+}(s)$, and a basis of $\ker(\mathbf{P}(s))$ by $\mathbf{P}^{\perp}(s)$. The vector $\boldsymbol{q}(s) \in \mathbb{C}^{\overline{m}}$ with $\overline{m} = n + m - \operatorname{rank}(\mathbf{P}(s))$ contains degrees of freedom that exist in the choice of the pair $[\boldsymbol{x}_\mathrm{s}^\top(s) \quad \boldsymbol{u}_\mathrm{s}^\top(s)]^\top$. The decomposition of $\mathbf{P}^{\perp}(s)$ is adapted to the left side of the equation, i.e. $\mathbf{P}_1^{\perp}(s) \in \mathbb{C}^{n\times\overline{m}}$ and $\mathbf{P}_2^{\perp}(s) \in \mathbb{C}^{m\times\overline{m}}$. From Eq. (2.8), we can deduce that the existence of a redundant actuator is equivalent to the existence of some $i \in \mathcal{I}_{1,m}$ with $\mathcal{I}_{1,m} = \{1, \ldots, m\}$ such that $u_i = 0$ is admissible for all $(\boldsymbol{x}_0,\ \boldsymbol{y}) \in \mathcal{S}$.[1] Then, every feasible trajectory can be produced without help of u_i.

Theorem 2 ([147, Th.1]). *System* (2.2) *has at least one redundant actuator if and only if it is input redundant.*

Proof. [147] ($\Rightarrow$) Let u_i be a redundant actuator. Then, by Definition 5, $u_i = 0$ is a possible choice for any tuple $(\boldsymbol{x}_0,\ \boldsymbol{y}) \in \mathcal{S}$. Hence, the i-th column of the transfer matrix $\mathbf{G}(s)$ must necessarily be

[1] We use the abbreviation $\mathcal{I}_{a,b} = \{a, \ldots, b\}$ with $a, b \in \mathbb{Z}$ and $a < b$ throughout the thesis.

linearly dependent on the other columns for all $s \in \mathbb{C}$. This means that $\max_{s\in\mathbb{C}} \operatorname{rank}(\mathbf{G}(s)) < m$ and, according to Theorem 1, the system is input redundant.

($\Leftarrow$) If we show $\mathbf{P}_2^{\perp}(s) \neq \mathbf{0}$ for an input redundant system, there exists at least one $i \in \mathcal{I}_{1,m}$ such that $\boldsymbol{\alpha}_i^{\top}\mathbf{P}_2^{\perp}(s) \neq \mathbf{0}$ with $\boldsymbol{\alpha}_i \in \mathbb{R}^m$ as the canonical unit vector pointing in the i-th coordinate direction. Hence, $\boldsymbol{q}(s)$ can always be chosen such that $u_i = 0$. In view of the decomposition [160, p. 181]

$$\mathbf{P}(s) = \begin{bmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{C}(s\mathbf{I}_n - \mathbf{A})^{-1} & \mathbf{I}_p \end{bmatrix} \underbrace{\begin{bmatrix} s\mathbf{I}_n - \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}(s) \end{bmatrix}}_{=:\,\mathbf{S}_1(s)} \underbrace{\begin{bmatrix} \mathbf{I}_n & -(s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B} \\ \mathbf{0} & \mathbf{I}_m \end{bmatrix}}_{=:\,\mathbf{S}_2(s)},$$

the kernel $\widetilde{\mathbf{P}}^{\perp}(s) := \mathbf{S}_2(s)\mathbf{P}^{\perp}(s)$ fulfills $\mathbf{S}_1(s)\widetilde{\mathbf{P}}^{\perp}(s) = \mathbf{0}$. Since $\operatorname{rank}(s\mathbf{I}_n - \mathbf{A}) = n$ for all but finitely many $s \in \mathbb{C}$, we have

$$\widetilde{\mathbf{P}}^{\perp}(s) = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}^{\perp}(s) \end{bmatrix}$$

for almost all $s \in \mathbb{C}$ with $\mathbf{G}^{\perp}(s)$ as a basis of the kernel of $\mathbf{G}(s)$. Using $\mathbf{P}^{\perp}(s) = \mathbf{S}_2^{-1}(s)\widetilde{\mathbf{P}}^{\perp}(s)$, we obtain

$$\mathbf{P}^{\perp}(s) = \begin{bmatrix} \mathbf{I}_n & (s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B} \\ \mathbf{0} & \mathbf{I}_m \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{G}^{\perp}(s) \end{bmatrix} = \begin{bmatrix} (s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B}\mathbf{G}^{\perp}(s) \\ \mathbf{G}^{\perp}(s) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_1^{\perp}(s) \\ \mathbf{P}_2^{\perp}(s) \end{bmatrix}.$$

From this, $\mathbf{P}_2^{\perp}(s) = \mathbf{G}^{\perp}(s) \neq \mathbf{0}$ follows. □

For unconstrained, linear time-invariant systems and the simple fault model $u_i = 0$, Definitions 3 and 5 coincide in the sense that an input redundant system possesses at least one redundant actuator and vice versa. I.e., there is one actuator that does not contribute to the set $\mathcal{S}$ and can, hence, be substituted by other actuators. This is an important finding because it justifies working with Definition 5 and its natural extension Definition 6. Both are well assimilated to the intuition of redundancy, as discussed in Section 2.1.2, and they admit a seamless use for the analysis of nonlinear systems.

2.1.3.2 Nonlinear Systems

We will now perform the comparison of actuator redundancy and input redundancy for nonlinear systems (2.1). It is important to note that this system class includes all systems with linear dynamics (2.2) for which the sets $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}^m$ are strict subsets of $\mathbb{R}^n$ or $\mathbb{R}^m$. As every real technical system falls into this class of systems due to its physical bounds, the results of this section are of utmost importance.

Theorem 3 ([147, Th. 2])**.** *For any constrained system governed by Eq.* (2.1)*, actuator redundancy implies input redundancy.*

Proof. [147] The set $\mathcal{S}$ of admissible tuples $(\boldsymbol{x}_0,\ \boldsymbol{y})$ is structured as $\mathcal{S} = \mathcal{S}|_{u_i=0} \cup \mathcal{S}_{u_i} \cup \mathcal{S}_{\text{coop}}$, where we denote by $\mathcal{S}_{u_i}$ the output trajectory set that is obtained when exclusively using the i-th

actuator ($u_j = 0 \, \forall \, j \in \mathcal{I}_{1,m} \setminus \{i\}$), and by $\mathcal{S}_{\text{coop}}$ the output trajectory set that is additionally created only when all actuators are working in cooperation, i.e. $\mathcal{S}_{\text{coop}} := \mathcal{S} \setminus \left(\mathcal{S}|_{u_i=0} \cup \mathcal{S}_{u_i}\right)$. If $\exists\, i : \mathcal{S} = \mathcal{S}|_{u_i=0}$, i.e., if there exists a redundant actuator, then it follows that $\mathcal{S}_{\text{coop}} = \emptyset$ and, hence, $\mathcal{S}_{u_i} \subseteq \mathcal{S}|_{u_i=0}$. We can thus choose an arbitrary $(\boldsymbol{x}_0, \boldsymbol{y}) \in \mathcal{S}_{u_i}$ and observe that there are multiple inputs $\boldsymbol{u}$ producing this trajectory because $(\boldsymbol{x}_0, \boldsymbol{y}) \in \mathcal{S}|_{u_i=0}$ holds as well. It follows that the system is non-left-invertible, i.e. input redundant. □

The proof is not conducted in terms of any specific system dynamics. Therefore, it holds for any linear or nonlinear system subject to arbitrary system bounds. The statement of the proof is quite intuitive because it means that, for every system equipped with a redundant actuator, there exists at least one output trajectory that can be realized by choosing at least two distinct input trajectories. This is clear because the redundant actuator does not enlarge the set $\mathcal{S}$. Hence, a trajectory that is produced by u_i could as well be realized using all other actuators u_j, $i \neq j$.

If we look at systems that are constrained, e.g., by a closed and bounded set $\mathcal{U}$, the situation can occur where $\mathcal{S}_{\text{coop}}$ is non-empty. That is, some trajectories from $\mathcal{S}$ can only be realized when all actuators work in cooperation. This can be the case due to non-linearities in the dynamics, or when operating the system close to its physical limits. Then, although there might be non-uniquely actuated tuples $(\boldsymbol{x}_0, \boldsymbol{y})$, there are as well trajectories in $\mathcal{S}$ for which operation of all actuators are necessary. Hence, the system is input redundant but there exists no redundant actuator.

2.1.3.3 Concluding Remarks

Oftentimes, for characterizing whether a dynamic system possesses redundancy properties, Condition (2.4), i.e. $m > p$, is used, see [89, 151]. Indeed, in view of unconstrained linear systems, this is a sufficient condition for input and actuator redundancy. The presence of more inputs than outputs yields $\max_{s\in\mathbb{C}} \operatorname{rank}(\mathbf{G}(s)) < m$. This renders the system input redundant, see Theorem 1. Hence, there exists at least one redundant actuator. The sufficiency of $m > p$ for actuator redundancy does, however, not transfer to nonlinear systems, as already outlined above. This is because the set $\mathcal{S}_{\text{coop}}$ may well be non-empty, although $m > p$. The same arguments can be applied to the condition $m > \operatorname{rank}(\mathbf{B}(\boldsymbol{x}))$ which has been used, e.g., by Härkegård and Glad [60], Khelassi et al. [80], and Tohidi et al. [159].

The results of this section show that the evaluation of fault-tolerance cannot be done on a system level. If an actuator is redundant, then the system can tolerate a fault of this actuator in the sense that the system's operating range is not changed in this case. This does, however, not imply that the system can tolerate the loss of an arbitrary actuator and still operate as in the nominal case.

The system designer has a great influence on the redundancy properties not only by physically constructing the system, choosing input and sensor positions, etc. They can also define the operating range of the system in terms of its system boundaries, i.e., the sets $\mathcal{X}$ and $\mathcal{U}$ can be explicitly specified. Doing this changes the results of a redundancy analysis and may decide on whether redundant actuators are present or not, see also [147, Ex. 1]. This conclusion aligns with the work

by Michellod [106] who states that there is no meaningful discussion about over-actuation and redundancy without defining the operating range of the system at hand.

2.2 Redundant Reachability

Looking closely at the results from Sections 2.1.2 and 2.1.3, the requirements for the presence of redundancy can be strict, depending on the definition of the system bounds. Oftentimes, it is not necessary to keep up the nominal behavior during the presence of faults. It is rather important that the system can be steered to and stabilized at some state that is considered safe. It is, however, often acceptable that the full functionality is not offered anymore.

This section is devoted to the question of *how* redundant an actuator is w.r.t. a given set of outputs or, more generally, how redundant the system is w.r.t. a set $\mathcal{Z}$ of fault scenarios. I.e., we want to find points in the output space $\mathbb{R}^p$ that the system can be steered to in a redundant way. Stabilizability of such points is then discussed in Section 2.3. To approach the problem, we present a method for computing – loosely speaking – an estimate of some trajectory set $\mathcal{S}_\mathcal{Z} \subseteq \mathcal{S}$ which describes all tuples $(\boldsymbol{x}_0, \boldsymbol{y})$ that are feasible independently of the fault scenarios z contained in $\mathcal{Z}$. Computing and storing all such tuples on a computer is an infeasible task as there are usually infinitely many of them. To develop a feasible but meaningful redundancy analysis, we will, therefore, reduce the analysis to one specific initial value $\boldsymbol{x}_0 \in \mathcal{X}$ that is relevant for plant operation. Furthermore, we refrain from computing feasible output *trajectories*, but we will identify *points* $\boldsymbol{y}(t)$ in the output space that are reachable independently of the specific fault scenario $z \in \mathcal{Z}$.

For obtaining the desired redundancy information, we apply reachability techniques. Referring to Lunze [97, p. 65], reachability and controllability are two related concepts. Controllability is concerned with the question whether a system can be driven from a given initial state to the origin by applying a suitable input signal. In turn, reachability asks an arbitrary point in state space to be attainable from the origin. Both properties are equivalent for linear time-invariant systems (2.2), but can deviate for linear time-varying or nonlinear systems (2.1) [97].

For stable linear systems of the form (2.2), strong statements can be made for the set of states that are reachable asymptotically from some initial state $\boldsymbol{x}_0$. This set forms a subspace of $\mathbb{R}^n$ and is given by the controllable subspace, see e.g. Trentelman [160],

$$\mathcal{W} := \operatorname{im}\left(\begin{bmatrix}\mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{bmatrix}\right) \tag{2.9}$$

of the system. This simple characterization makes use of the fact that the time solution of the differential equation (2.2a) is analytically known as

$$\boldsymbol{x}(t) = \mathrm{e}^{\mathbf{A}t}\boldsymbol{x}_0 + \int_0^t \mathrm{e}^{\mathbf{A}(t-\tau)}\mathbf{B}\boldsymbol{u}(\tau)\mathrm{d}\tau. \tag{2.10}$$

For a derivation of Eq. (2.9) from Eq. (2.10), see e.g. [97, Sec. 3.1.2]. The subspace $\mathcal{W}$ describes all directions in which the system can be steered by using the inputs. Hence, at each fixed time

instance $t > 0$, every point in the affine subspace

$$\mathcal{W}_{\mathrm{A}}(\boldsymbol{x}_0, t) = \left\{ \mathrm{e}^{\mathbf{A}t}\boldsymbol{x}_0 + \boldsymbol{w} \,|\, \boldsymbol{w} \in \mathcal{W} \right\}$$

can be reached. This situation is illustrated in Fig. 2.1a.

Such simple statements can unfortunately not be made as soon as we incorporate system bounds or nonlinear dynamics. For this reason, we will make use of a more general formulation of reachability:

Definition 7 (Reachable set, see [6, Sec. 1.1])**.** *Let the solution of Eq.* (2.1a) *at time t be denoted by $\boldsymbol{\Phi}(\boldsymbol{x}_0, \boldsymbol{u}, t)$. The reachable set $\mathcal{R}(\boldsymbol{x}_0, t) \subseteq \mathcal{X}$ containing all attainable states at time $t \geq 0$ is defined by*

$$\mathcal{R}(\boldsymbol{x}_0, t) = \left\{ \boldsymbol{\Phi}(\boldsymbol{x}_0, \boldsymbol{u}, t) \,|\, \boldsymbol{u} \colon [0; t) \to \mathcal{U},\ \boldsymbol{\Phi}(\boldsymbol{x}_0, \boldsymbol{u}, \tau) \in \mathcal{X}\ \forall\ \tau \in [0; t] \right\}.$$

In other words, the reachable set $\mathcal{R}(\boldsymbol{x}_0, t)$ contains all states $\boldsymbol{x}(t)$ for which there exists an input trajectory $\boldsymbol{u}$ driving the system from $\boldsymbol{x}_0$ to $\boldsymbol{x}(t)$ in the time frame $[0; t]$, while obeying the input bounds $\mathcal{U}$ and the state bounds $\mathcal{X}$ for all times along the trajectory.

Along the lines of Definition 6 and our prior discussions, it is necessary to view reachable points in the output space rather than in state space. To this end, we state

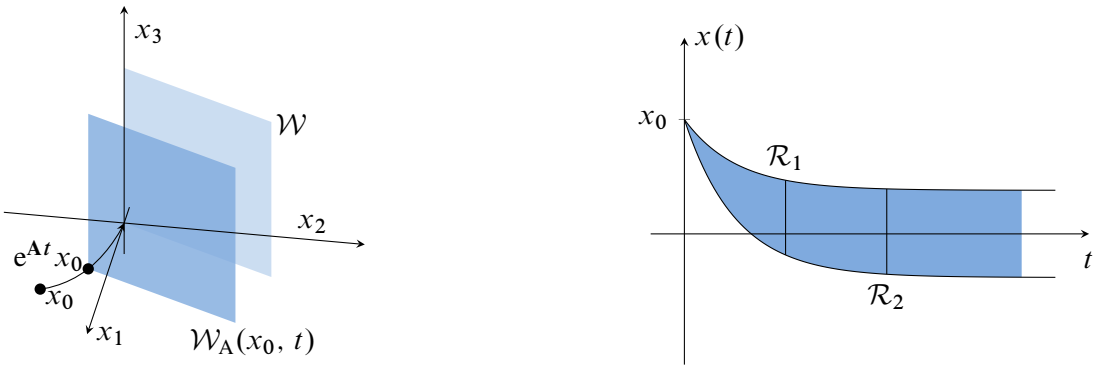
Definition 8 (Output reachable set)**.** *The output reachable set $\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0, t) \subseteq \mathbb{R}^p$ containing all attainable output points at time $t \geq 0$ is defined as the mapping of the reachable states to the output space, that is*

$$\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0, t) = \left\{ \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) \,|\, \boldsymbol{x} \in \mathcal{R}(\boldsymbol{x}_0, t),\ \boldsymbol{u} \in \mathcal{U} \right\}.$$

For ensuring a correct interpretation of reachability analyses, it is important to note what reachable sets $\mathcal{R}$, or $\mathcal{R}_{\boldsymbol{y}}$ depict exactly. Let us discuss this exemplarily for reachable sets $\mathcal{R}$ in state space. For every time instance $t \geq 0$, this set contains all points $\boldsymbol{x}_{\mathrm{R}} \in \mathcal{X}$ that can possibly be reached from a given initial state $\boldsymbol{x}_0$ in the sense that the system is exactly driven to the point $\boldsymbol{x}_{\mathrm{R}}$ *at* this time instance. There is, however, no statement about whether the system can be kept at this point, i.e., if there exists an input trajectory $\boldsymbol{u}$ ensuring $\boldsymbol{f}(\boldsymbol{x}_{\mathrm{R}}, \boldsymbol{u}) = \boldsymbol{0}$.

Furthermore, there is no statement on the state trajectory $\boldsymbol{\Phi}(\boldsymbol{x}_0, \boldsymbol{u}, \cdot)$ driving the system from $\boldsymbol{x}_0$ to $\boldsymbol{x}_{\mathrm{R}}$. The latter is especially important when viewing the propagation of $\mathcal{R}(\boldsymbol{x}_0, t)$ over time. Doing this creates reachable *tubes* in the time domain, see Fig. 2.1b. Looking at two time instances $t_1 < t_2$ and their respective reachable sets $\mathcal{R}_1 := \mathcal{R}(\boldsymbol{x}_0, t_1)$ and $\mathcal{R}_2 := \mathcal{R}(\boldsymbol{x}_0, t_2)$, there is no statement that any point in $\mathcal{R}_2$ is reachable from any point in $\mathcal{R}_1$. For continuous plants (2.1), this is due to the Lipschitz continuity of the differential equation. Assume $t_2 - t_1 \to 0$, i.e., the two lines in Fig. 2.1b approach each other and, in the limit, merge. Being able to choose two arbitrary points $\boldsymbol{x}_1, \boldsymbol{x}_2$ from the sets $\mathcal{R}_1$ and $\mathcal{R}_2$, respectively, with $||\boldsymbol{x}_2 - \boldsymbol{x}_1|| > 0$ would force the state trajectory to be discontinuous, which is ruled out by Lipschitz continuity of the differential equation.

Concluding, the reachable set $\mathcal{R}(\boldsymbol{x}_0, t)$ describes all points in the state space reachable from $\boldsymbol{x}_0$ at time t, while guaranteeing $\boldsymbol{x}(\tau) \in \mathcal{X}$ for all $\tau \in [0; t]$ and $\boldsymbol{u}(\tau) \in \mathcal{U}$ for all $\tau \in [0; t)$.



(a) Evolution of the affine subspace $\mathcal{W}_A$ through state space for a linear unbounded system.

(b) Evolution of a reachable set $\mathcal{R}$ over time for $\mathcal{U}$ bounded.

Figure 2.1: Reachable sets for unbounded and bounded systems.

Remark 1. Later, we will also use the notation $\mathcal{R}(\mathcal{X}_0, t)$ and $\mathcal{R}_{\boldsymbol{y}}(\mathcal{X}_0, t)$, respectively, where $\mathcal{X}_0$ is a set of initial points $\boldsymbol{x}_0$. This is a slight abuse of notation. We denote by it the union of all corresponding reachable sets, i.e.

$$\mathcal{R}(\mathcal{X}_0, t) = \bigcup_{\boldsymbol{x}_0 \in \mathcal{X}_0} \mathcal{R}(\boldsymbol{x}_0, t) \qquad \text{and} \qquad \mathcal{R}_{\boldsymbol{y}}(\mathcal{X}_0, t) = \bigcup_{\boldsymbol{x}_0 \in \mathcal{X}_0} \mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0, t).$$

2.2.1 The Set of Redundantly Reachable Outputs

The discussion conducted in the prior section is important for interpreting the results for redundantly reachable outputs, which we will introduce now. The set of redundantly reachable outputs contains points in the output space that are always reachable from a given initial state $\boldsymbol{x}_0$, independently of which fault occurs. Points that are included in such a set at a given time point t have advanced safety properties. They can be reached independently of the occurring faults because the needed control action is non-uniquely distributed among the system actuators. Therefore, the set of redundantly reachable outputs helps us to reveal redundancy structures that are available due to the construction of the system and its resulting coupled dynamics. The coupling effects make it possible to substitute a nominal control action in a fault scenario while maintaining the same output.

System analysis that is performed in this way enables us to broaden the discussion on the presence of redundancy. Depending on the system's task, there might exist a satisfactory level of redundancy although there are no backup systems. Such an approach can help designing dynamic systems more specifically in terms of safety. It might not always be necessary to provide a whole backup system since the system dynamics provides sufficient degrees of freedom to handle possibly occurring faults. On the one hand, this may save production costs and reduce the overall system complexity as some subsystems may be unnecessary. On the other hand, statements can be obtained on where in the system additional redundancy structures might be needed in order to guarantee a desired safety level. The set of redundantly reachable points can also be used directly in control design. Trajectories can be chosen such that they are feasible in any possible fault sce-

nario. This enables for a fully fault-tolerant, i.e. seamless, system operation without the need of replanning a trajectory in some fault event.

In the following, we provide two suitable definitions of redundantly reachable output sets and examine their properties for specific fault scenarios. Furthermore, we will identify conditions for the considered system class such that a simplified computation without approximation of such sets is admitted.

Definition 9. *Let $\boldsymbol{x}_0 \in \mathcal{X}$ and $\mathcal{Z}$ be a set of fault scenarios z of which every one is governed by a system Σ_z of the form* (2.7)*. Denoting the output reachable set of every system Σ_z from $\boldsymbol{x}_0$ at time t by $\mathcal{R}_{\boldsymbol{y}}^z(\boldsymbol{x}_0, t)$, we define the set $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$ of redundantly reachable outputs from $\boldsymbol{x}_0$ at time t as*

$$\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t) = \bigcap_{z \in \mathcal{Z}} \mathcal{R}_{\boldsymbol{y}}^z(\boldsymbol{x}_0, t). \tag{2.11}$$

An alternative formulation of a redundantly reachable set is given by

Definition 10. *Let $\boldsymbol{x}_0 \in \mathcal{X}$ and $\mathcal{Z}$ be a set of fault scenarios z of which every one is governed by a system Σ_z of the form* (2.7)*. Denoting the solution of Eq.* (2.7a) *at time t by $\boldsymbol{\Phi}_z(\boldsymbol{x}_0, \boldsymbol{u}, t)$ and the output reachable set of every system Σ_z from $\boldsymbol{x}_0$ at time t by $\mathcal{R}_{\boldsymbol{y}}^z(\boldsymbol{x}_0, t)$, we define the set $\hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$ of redundantly reachable outputs from $\boldsymbol{x}_0$ at time t as*

$$\hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, t) = \left\{ \boldsymbol{y} \in \bigcap_{z \in \mathcal{Z}} \mathcal{R}_{\boldsymbol{y}}^z(\boldsymbol{x}_0, t) \;\middle|\; \forall\, z \in \mathcal{Z}\ \exists\, \boldsymbol{u}_z \colon [0; t) \to \mathcal{U}_z(\tau) \text{ s.t.} \right.$$
$$\left. \boldsymbol{y}_z(\tau) = \boldsymbol{g}_z(\boldsymbol{\Phi}_z(\boldsymbol{x}_0, \boldsymbol{u}, \tau), \boldsymbol{u}_z(\tau), \tau) \in \hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, \tau)\ \forall \tau \in [0; t) \right\}.$$

There is a difference in the Definitions 9 and 10. Definition 9 collects all points that are reachable in all fault scenarios at some predefined time point t. However, there is no statement on the reachable points preceding this time point t. This is covered by Definition 10. Here, we implicitly construct a *tube* of output reachable points in the time domain. For each fault scenario $z \in \mathcal{Z}$, it is demanded that there exists an admissible input trajectory $\boldsymbol{u}_z$ keeping the respective output trajectory $\boldsymbol{y}_z$ in this tube at all times up to time t. The set $\hat{\mathcal{Q}}(\boldsymbol{x}_0, t)$ collects all points in the output space that are reachable in this way at time t.

This is a stronger definition compared to Definition 9. In Definition 10, redundantly reachable outputs are forced to be an element of $\hat{\mathcal{Q}}_{\mathcal{Z}}$, also *before* the time point t, not only at this time. This stricter conditions also results in the fact that there may exist a time point $t^* \geq 0$, such that $\hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, t) = \emptyset$ holds for all $t \geq t^*$. Due to the construction of $\hat{\mathcal{Q}}_{\mathcal{Z}}$, it is not possible to recover from this situation. For Definition 9, it is not a priori ruled out to obtain $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t) \neq \emptyset$ for $t > t^*$ although $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t^*) = \emptyset$.

Depending on the purpose, both definitions have their validity. For investigating stabilizability, which we will do in Section 2.3, Definition 9 suffices. However, if we are interested in keeping a whole output trajectory within some tube in all fault scenarios, we need to view the points

collected in $\hat{\mathcal{Q}}_{\mathcal{Z}}$. In the following, we will work with $\mathcal{Q}_{\mathcal{Z}}$ and Definition 9 because it is more compact but still delivers good insights in the challenges that we face during the computation of $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$, respectively. While the theoretical results that we will obtain for $\mathcal{Q}_{\mathcal{Z}}$ in Sections 2.2.1.1 and 2.2.1.2 can be transferred to $\hat{\mathcal{Q}}_{\mathcal{Z}}$, we will need to introduce a separate algorithm for computing $\hat{\mathcal{Q}}_{\mathcal{Z}}$.

Both definitions of redundantly reachable outputs are formulated in a general way as there is no further specification on the set of fault scenarios $\mathcal{Z}$. In view of Definition 5, we now put a focus on scenarios describing the loss of actuators. The properties of $\mathcal{Q}_{\mathcal{Z}}$ under the classical fault model $u_i = 0$, $i \in \mathcal{I}_{1,m}$, are examined as well as its properties under constant actuator faults $u_i = \text{const}$. Both cases depict an interesting and challenging class of faults because they describe an infinite number of faults. For $u_i = 0$, this is because the time of fault occurrence is unknown. For $u_i = \text{const}$, there are naturally infinitely many positions in which the actuator can be locked. In both cases, $\mathcal{Z}$ becomes an infinite set over whose elements we want to perform an intersection operation, see Definition 9.

2.2.1.1 Fault model $u_i = 0$

Modeling an actuator fault as $u_i = 0$ is intuitive. Imagine an electric drive that cannot produce any torque due to the loss of supply voltage. If the drive's torque is used as the input signal of some mechanical dynamic system, this fault situation is equivalent to $u_i = 0$.

Let us assume that a user-defined subset $\mathcal{M}$ of all actuators is included in the redundancy analysis, i.e. $i \in \mathcal{M} \subseteq \mathcal{I}_{1,m}$. Each fault model Σ_z is then defined by the system Σ_0, except for a reduced set of inputs

$$\mathcal{U}_z(t) = \mathcal{U}_i^{t_\mathrm{f}}(t) := \begin{cases} \mathcal{U} & t < t_\mathrm{f} \\ \{\boldsymbol{u} \in \mathcal{U} \mid u_i = 0\} & t \geq t_\mathrm{f} \end{cases}, \tag{2.12}$$

where $t_\mathrm{f} \geq 0$ is the time point at which the fault occurs. Then, the set of fault scenarios reads

$$\mathcal{Z}_0 := \left\{ \left(\boldsymbol{f},\, \boldsymbol{g},\, \mathcal{X},\, \mathcal{U}_i^{t_\mathrm{f}}(t) \right) \mid i \in \mathcal{M},\, t_\mathrm{f} \geq 0 \right\}. \tag{2.13}$$

By the definition of $\mathcal{U}_i^{t_\mathrm{f}}(t)$ according to Eq. (2.12), we implicitly assume that only one fault can happen simultaneously. This assumption is common in the field of fault analysis [67, p. 51]. However, our redundancy analysis is not limited to single fault cases. If multiple faults should be considered, this can be done by integrating all expected fault combinations into the fault dynamics Σ_z. Thereby, additional fault scenarios are produced and included in $\mathcal{Z}$.

Eq. (2.13) defines a set containing infinitely many elements. This is due to the time of failure t_f that can take any real positive value, as it is unknown. In view of Definition 9, this results in an infinite number of reachable sets and intersections that need to be computed. This is not feasible. However, it is possible to identify the fault scenarios that shape the set of redundant outputs $\mathcal{Q}_{\mathcal{Z}}$. They are given by the worst-case scenario, i.e. the actuators failing at the initial time point $t = 0$. We have

Theorem 4. *Let $\boldsymbol{x}_0 \in \mathcal{X}$ and $t \geq 0$. The set $\mathcal{Q}_{\mathcal{Z}_0}(\boldsymbol{x}_0, t)$ of redundantly reachable outputs from x_0 at time t for the set of fault scenarios $\mathcal{Z}_0$ is given by*

$$\mathcal{Q}_{\mathcal{Z}_0}(\boldsymbol{x}_0, t) = \bigcap_{z \in \mathcal{Z}_0} \mathcal{R}_{\boldsymbol{y}}^{z}(\boldsymbol{x}_0, t) = \bigcap_{z \in \mathcal{Z}_0^0} \mathcal{R}_{\boldsymbol{y}}^{z}(\boldsymbol{x}_0, t),$$

with $\mathcal{Z}_0^0 = \left\{\left(\boldsymbol{f}, \boldsymbol{g}, \mathcal{X}, \mathcal{U}_i^{t_\mathrm{f}}(t)\right) \mid i \in \mathcal{M}, t_\mathrm{f} = 0\right\}$.

Proof. Let $0 \leq t_{\mathrm{f}1} \leq t_{\mathrm{f}2}$. Then, we have $\mathcal{U}_i^{t_{\mathrm{f}1}}(t) \subseteq \mathcal{U}_i^{t_{\mathrm{f}2}}(t)$ for each time point $t \geq 0$. Denoting by $\mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}j}}$ the output reachable set of the quadruple $(\boldsymbol{f}, \boldsymbol{g}, \mathcal{X}, \mathcal{U}_i^{t_{\mathrm{f}j}}(t))$, $j \in \mathcal{I}_{1,2}$, the inclusion $\mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}1}}(\boldsymbol{x}_0, t) \subseteq \mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}2}}(\boldsymbol{x}_0, t)$ follows for every $t \geq 0$. This implies

$$\mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}1}}(\boldsymbol{x}_0, t) \cap \mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}2}}(\boldsymbol{x}_0, t) = \mathcal{R}_{\boldsymbol{y}}^{t_{\mathrm{f}1}}(\boldsymbol{x}_0, t)$$

and, therefore, the statement of the theorem

$$\mathcal{Q}_{\mathcal{Z}_0}(\boldsymbol{x}_0, t) = \bigcap_{z \in \mathcal{Z}_0^0} \mathcal{R}_{\boldsymbol{y}}^{z}(\boldsymbol{x}_0, t).$$

□

From the theorem, we have that it is sufficient to look at $t_\mathrm{f} = 0$ as the time of failure. This is because the fault scenario with time of failure $t_\mathrm{f} = 0$ defines the result of the intersection operation in (2.11). Thus, we can simplify the computation of $\mathcal{Q}_{\mathcal{Z}_0}$ originally demanding for the calculation and intersection of infinitely many reachable sets. That is, it suffices to compute and process a finite number of card($\mathcal{M}$) reachable sets, where card($\cdot$) denotes the cardinality of the argument set. No approximation error is introduced by this reduction.

2.2.1.2 Fault model u_i = const

After discussing the simplest possible fault model $u_i = 0$, $i \in \mathcal{M}$, let us extend the class of considered faults by relaxing the fault condition to any input signal $\boldsymbol{u} \in \mathcal{U}$ where the i-th component is constant after the occurrence of the fault at time t_f. This scenario depicts system inputs that are stuck at some value within their constraints. If the actuator signals can be set independently, these constraints are given by an interval. As this is often the case in technical applications we will assume it in the following. I.e., the input bounds $\mathcal{U}$ are given by a cartesian product of intervals as $\mathcal{U} = \bigtimes_{i=1}^{m} [\underline{u}_i; \overline{u}_i]$ with $\underline{u}_i, \overline{u}_i \in \mathbb{R}$, $\underline{u}_i \leq \overline{u}_i$.

Considering u_i = const as a separate fault model can be motivated by chemical plants or examples from the aviation sector. Chemical systems often include valves that can freeze at some fixed position due to mechanical defects, corrosion, or chemical reactions with the process fluid, see [111]. In aviation, freezing of actuators primarily concerns the control surfaces. This depicts a type of fault that influences the aircraft efficiency because it induces additional undesired drag [35]. However, it also reduces its maneuverability causing a considerable safety risk that might – in worst-case scenarios – destabilize the aircraft leading to a crash. Occurrences of stuck control surfaces have caused the crashes of two Boeing 737 aircraft in the 1990s [112, 113]. After a

hydraulic valve had jammed, the rudders of both aircraft moved to their maximally possible deviation. This, in turn, initiated a severe rolling motion that could not be compensated by the pilots. Due to these facts, this type of fault is well researched. Literature focusing on the detection of such faults is exemplarily given by Cieslak et al. [35], Varga [163], and Yang et al. [175].

For the discussed type of faults, the set of fault scenarios $\mathcal{Z}_\mathrm{c}$ is again composed of infinitely many elements. In this setting, we do not only have infinitely many times of failure t_f but there are infinitely many positions at which an actuator may be stuck. For each fault scenario z, we have a set of inputs that is additionally parameterized by a constant $c \in [\underline{u}_i; \overline{u}_i]$ describing the fault position of the actuator. It reads

$$\mathcal{U}_z(t) = \mathcal{U}_i^{c,t_\mathrm{f}}(t) := \begin{cases} \mathcal{U} & t < t_\mathrm{f} \\ \{\boldsymbol{u} \in \mathcal{U} \,|\, u_i = c\} & t \geq t_\mathrm{f} \end{cases}. \tag{2.14}$$

The set $\mathcal{Z}_\mathrm{c}$ is then defined as

$$\mathcal{Z}_\mathrm{c} := \left\{ \left(\boldsymbol{f},\, \boldsymbol{g},\, \mathcal{X},\, \mathcal{U}_i^{c,t_\mathrm{f}}(t) \right) \,|\, i \in \mathcal{M},\, c \in [\underline{u}_i; \overline{u}_i],\, t_\mathrm{f} \geq 0 \right\}.$$

The aim is to determine the set of redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$. In order for the intersection operation to be computationally feasible, $\mathcal{Z}_\mathrm{c}$ must be a finite set, which it is not. Following the same arguments as in the proof of Theorem 4, we can, however, focus on the time of failure $t_\mathrm{f} = 0$ for obtaining the redundantly reachable outputs in $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$. This means that $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}} = \mathcal{Q}_{\mathcal{Z}_\mathrm{c}^0}$, where

$$\mathcal{Z}_\mathrm{c}^0 := \left\{ \left(\boldsymbol{f},\, \boldsymbol{g},\, \mathcal{X},\, \mathcal{U}_i^{c,t_\mathrm{f}}(t) \right) \,|\, i \in \mathcal{M},\, c \in [\underline{u}_i; \overline{u}_i],\, t_\mathrm{f} = 0 \right\}.$$

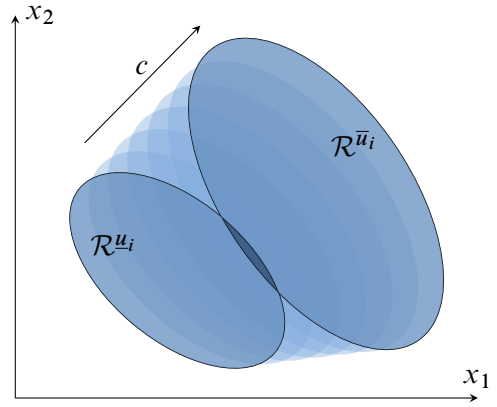
Due to the intervals $[\underline{u}_i; \overline{u}_i]$ containing infinitely many elements, there is still an infinite number of fault scenarios in $\mathcal{Z}_\mathrm{c}^0$. Hence, $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}^0}$ can still not be computed in finite time. In the following, we will identify two properties that are sufficient to base the computation of $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}^0}$, and therefore $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$, on a finite set of fault scenarios. Let us first introduce

Definition 11 (see [147, Def. 8])**.** *Let $\boldsymbol{x}_0 \in \mathcal{X}$ and $\boldsymbol{u}\colon [0;\, t) \to \mathcal{U}_i^{c,0}(t)$, i.e., actuator u_i is equal to a constant c. Further, let the solution of Eq.* (2.1a) *at time t again be denoted by $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t)$. We call the system monotone in the constant input u_i at time t if all elements of $\boldsymbol{\Phi}$, denoted Φ_j, $j \in \mathcal{I}_{1,n}$, fulfill*

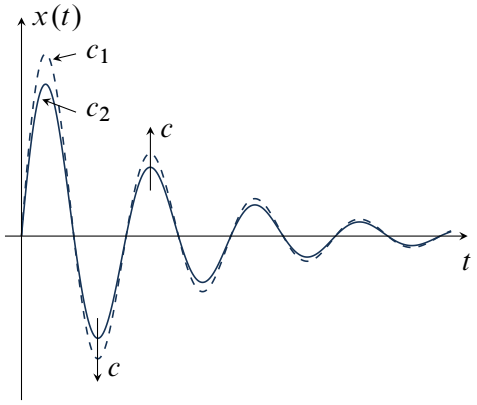
$$\Phi_j(\boldsymbol{x}_0,\, \boldsymbol{u}^1,\, t) \geq (\leq)\, \Phi_j(\boldsymbol{x}_0,\, \boldsymbol{u}^2,\, t) \tag{2.15}$$

for all control signals $\boldsymbol{u}^1\colon [0;\, t) \to \mathcal{U}_i^{c_1,0}(t)$, $\boldsymbol{u}^2\colon [0;\, t) \to \mathcal{U}_i^{c_2,0}(t)$ that are chosen such that $u_k^1(\tau) = u_k^2(\tau)$, $k \neq i$, and $c_1 = u_i^1(\tau) \geq u_i^2(\tau) = c_2$ for all $\tau \in [0;\, t)$.

Eq. (2.15) asks the elements Φ_j of $\boldsymbol{\Phi}$ to be monotone functions of $u_i = c$. Let us look at the illustration in Fig. 2.2a for making this abstract definition more comprehensible. For a fixed time $t \geq 0$, reachable sets of the plant with $\mathcal{U}_z(t) = \mathcal{U}_i^{c,0}(t)$ with different constants $c \in [\underline{u}_i; \overline{u}_i]$ for some actuator u_i are drawn. We denote by $\mathcal{R}^{\underline{u}_i}$ and $\mathcal{R}^{\overline{u}_i}$ the reachable sets of the systems $(\boldsymbol{f},\, \boldsymbol{g},\, \mathcal{X},\, \mathcal{U}_i^{\underline{u}_i,0}(t))$ and $(\boldsymbol{f},\, \boldsymbol{g},\, \mathcal{X},\, \mathcal{U}_i^{\overline{u}_i,0}(t))$, respectively. For a system possessing the monotonicity property from Definition 11, the reachable sets are shifted monotonously through the



(a) Reachable sets being monotonously shifted in state space.



(b) Time solution for two different constant inputs $c_1 > c_2$.

Figure 2.2: Illustrations for Definition 11.

state space when increasing the parameter c because each component of the solution $\boldsymbol{\Phi}$ is a monotone function of c. Hence, intersecting all such reachable sets is equivalent to intersecting the two extremal sets $\mathcal{R}^{\underline{u}_i}$ and $\mathcal{R}^{\overline{u}_i}$, as they define the result of this intersection operation.

Remark 2. We have implicitly used convex sets for our argument in the above paragraph. This is needed for guaranteeing that the intersection of the reachable sets monotonously decreases while c changes. This does not always hold for non-convex sets.

Remark 3. It is important to note the meaning of the formulation *monotone at time t*. The key point is that the monotonicity relation of the solution $\boldsymbol{\Phi}$ w.r.t. the constant control input u_i may have different characteristics for different time points. The shifting direction of $\boldsymbol{\Phi}$, i.e. the relation signs $\geq, \leq$ in Eq. (2.15), may vary over time. Hence, each time point can be viewed separately. See Fig. 2.2b for a system response created with two constant input signals $c_1 > c_2$. The dashed curve that has been created using c_1 has a larger amplitude than the signal created using c_2. Although the shifting direction of the signal w.r.t. a change of c changes over time, the underlying system fulfills Definition 11.

Finally, let us note a property of systems possessing this type of monotonicity. We will make use of it when identifying system classes fulfilling Definition 11.

Lemma 1. *Let $\boldsymbol{x}_0 \in \mathcal{X}$. A system of the form* (2.1) *is monotone in the constant input u_i at time t if*

$$\operatorname{sign}\left(\frac{\mathrm{d}\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t)}{\mathrm{d}u_i}\right)$$

is independent of $\boldsymbol{u}$ for every fixed t. Here, we have used $\operatorname{sign}(\cdot)$ *as the signum function.*

The proof follows directly from the definition of a monotone function in the parameter $u_i = c$ for every element $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t) \in \mathcal{R}(\boldsymbol{x}_0,\, t)$ created by some admissible input function $\boldsymbol{u} \colon [0;\, t) \to \mathcal{U}_i^{c,0}(t)$.

Having discussed Definition 11 in detail, we can now resume the computation of the redundantly reachable output set $\mathcal{Q}_{\mathcal{Z}_c}$. Originally relying on the computation and intersection of infinitely many reachable sets, we can reduce this expense to finitely many operations.

Theorem 5 (see [147, Th. 4]). *Let $\mathcal{U}$ be defined by the cartesian product of intervals as $\mathcal{U} = \bigtimes_{i=1}^{m}[\underline{u}_i; \overline{u}_i]$. For all $t \geq 0$, assume the output equation* (2.1b) *to be a monotone function of $\boldsymbol{x}$ and $\boldsymbol{u}$ for all $\boldsymbol{x}(t) \in \mathcal{R}(\boldsymbol{x}_0, t)$ and let $\mathcal{R}_{\boldsymbol{y}}^{z}(\boldsymbol{x}_0, t)$ with $z \in \mathcal{Z}_c$ be convex. If the system* (2.1a) *is monotone in all constant inputs u_i, $i \in \mathcal{M}$, at time t, then*

$$\mathcal{Q}_{\mathcal{Z}_c}(\boldsymbol{x}_0, t) = \bigcap_{z \in \overline{\mathcal{Z}}_c^0} \mathcal{R}_{\boldsymbol{y}}^{z}(\boldsymbol{x}_0, t)$$

with $\overline{\mathcal{Z}}_c^0 := \left\{ \left(\boldsymbol{f}, \boldsymbol{g}, \mathcal{X}, \mathcal{U}_i^{c,t_f}(t) \right) \mid i \in \mathcal{M},\, c \in \{\underline{u}_i, \overline{u}_i\},\, t_f = 0 \right\}$.

Proof. For notational convenience, let us denote by $\mathcal{R}_{\boldsymbol{y}}^{\lambda}$ the output reachable set of the system $(\boldsymbol{f}, \boldsymbol{g}, \mathcal{X}, \mathcal{U}_i^{\lambda,t_f}(t))$ for some $i \in \mathcal{M}$. We then trivially have $\mathcal{R}_{\boldsymbol{y}}^{\lambda_1} = \mathcal{R}_{\boldsymbol{y}}^{\lambda_2}$ for $\lambda_1 = \lambda_2 = \underline{u}_i$. When increasing λ_2, the reachable set $\mathcal{R}_{\boldsymbol{y}}^{\lambda_2}$ is shifted monotonously in the output space, because each component of the solution $\boldsymbol{\Phi}$ of Eq. (2.1a) is a monotone function of the parameter λ_2 and the output equation is assumed monotone on each corresponding state reachable set and for every admissible $\boldsymbol{u}$ in the case of a feedthrough. Using the assumption of convex sets $\mathcal{R}_{\boldsymbol{y}}^{\lambda}$, the intersection $\mathcal{R}_{\boldsymbol{y}}^{\lambda_1} \cap \mathcal{R}_{\boldsymbol{y}}^{\lambda_2}$ reduces monotonously while λ_2 increases. The limit is reached at $\lambda_2 = \overline{u}_i$, yielding the smallest possible intersection

$$\min_{\lambda_1, \lambda_2 \in [\underline{u}_i; \overline{u}_i]} \mathcal{R}_{\boldsymbol{y}}^{\lambda_1} \cap \mathcal{R}_{\boldsymbol{y}}^{\lambda_2} = \mathcal{R}_{\boldsymbol{y}}^{\underline{u}_i} \cap \mathcal{R}_{\boldsymbol{y}}^{\overline{u}_i}.$$

Applying this procedure for all indices $i \in \mathcal{M}$ yields the statement of the theorem. □

We can reduce the computation of $\mathcal{Q}_{\mathcal{Z}_c}$ to a number of $2 \cdot \mathrm{card}(\mathcal{M})$ reachable set computations and their intersections. For monotone systems w.r.t. Definition 11 possessing convex output reachable sets, this is, again, possible without introducing an error.

Additionally, we can note the inclusion $\mathcal{Q}_{\mathcal{Z}_c} \subseteq \mathcal{Q}_{\mathcal{Z}_0}$ holds. This can easily be verified as the set of fault scenarios $\mathcal{Z}_c$ includes $\mathcal{Z}_0$, i.e. $\mathcal{Z}_0 \subseteq \mathcal{Z}_c$. Due to the intersection over the elements of $\mathcal{Z}_c$, the result will be a smaller set. This shows that a reduction of system performance has to be expected when viewing a larger set of fault scenarios.

2.2.2 Results for Simplifying the Computation of $\mathcal{Q}_{\mathcal{Z}_c}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}_c}$

2.2.2.1 Linear Constrained Systems

We will show now that linear time-invariant systems with convex constraints have the monotonicity property from Definition 11 and maintain convex reachable sets. Hence, the dynamics considered now is given by Eq. (2.2), and the constraint sets $\mathcal{X}$ and $\mathcal{U}$ are assumed convex.

Monotonicity First, we use Lemma 1 to verify monotonicity in every constant input u_i, $i \in \mathcal{M}$, at time t. The solution of Eq. (2.2a) at any time point $t \geq 0$ is given analytically by Eq. (2.10). As we are interested in the sensitivity w.r.t. constant input signals $u_i = c$, we compute the derivative

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}c} &= \frac{\mathrm{d}}{\mathrm{d}c}\mathrm{e}^{\mathbf{A}t}\boldsymbol{x}_0 + \frac{\mathrm{d}}{\mathrm{d}c}\int_0^t \mathrm{e}^{\mathbf{A}(t-\tau)}\mathbf{B}\boldsymbol{u}(\tau)\mathrm{d}\tau \\ &= \frac{\mathrm{d}}{\mathrm{d}c}\int_0^t \mathrm{e}^{\mathbf{A}(t-\tau)}\begin{bmatrix}\boldsymbol{b}_1 & \cdots & \boldsymbol{b}_i & \cdots & \boldsymbol{b}_m\end{bmatrix}\begin{bmatrix}u_1(\tau) & \cdots & c & \cdots & u_m(\tau)\end{bmatrix}^\top \mathrm{d}\tau \\ &= \int_0^t \mathrm{e}^{\mathbf{A}(t-\tau)}\boldsymbol{b}_i\mathrm{d}\tau,\end{aligned}$$

which is a function that does not depend on the input trajectory $\boldsymbol{u}$. This means that, for a given t, the system solution is shifted monotonously through the state space when changing the parameter c. Furthermore, as the derivative $\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}c}$ is identical for each point in the state space, the shape of the reachable sets stays constant over c, i.e. a pure shift without rotation or deformation. By linearity and, therefore, monotonicity of the output equation (2.2b), this behavior transfers to the outputs.

Convexity Second, we show that the reachable sets of systems with linear time-invariant dynamics and convex sets $\mathcal{X}$ and $\mathcal{U}$ remain convex for all $t \geq 0$. For some initial state $\boldsymbol{x}_0$ and time $t \geq 0$, we have

$$\mathcal{R}(\boldsymbol{x}_0, t) = \Big\{\boldsymbol{x}(t) \in \mathbb{R}^n \mid \forall\ t' \in [0; t]\ \exists\, \boldsymbol{u}\colon [0; t') \to \mathcal{U} \text{ s.t. } \boldsymbol{x}(t') = \mathrm{e}^{\mathbf{A}t'}\boldsymbol{x}_0 + \int_0^{t'} \mathrm{e}^{\mathbf{A}(t-\tau)}\mathbf{B}\boldsymbol{u}(\tau)\mathrm{d}\tau \in \mathcal{X}\Big\}.$$

Let us choose two functions $\boldsymbol{u}_1, \boldsymbol{u}_2\colon [0; t') \to \mathcal{U}$. As $\mathcal{U}$ is convex, every control input

$$\boldsymbol{u}_3(\tau) := \boldsymbol{u}_1(\tau)(1-\lambda) + \boldsymbol{u}_2(\tau)\lambda$$

fulfills $\boldsymbol{u}_3(\tau) \in \mathcal{U}$ for all $\lambda \in [0; 1]$ and for all $\tau \in [0; t')$ by definition of a convex set. By linearity of Eq. (2.10), the input signal $\boldsymbol{u}_3$ is mapped to

$$\begin{aligned}\boldsymbol{x}_3(t') &:= \mathrm{e}^{\mathbf{A}t'}\boldsymbol{x}_0 + \int_0^{t'} \mathrm{e}^{\mathbf{A}(t'-\tau)}\mathbf{B}\left(\boldsymbol{u}_1(\tau)(1-\lambda) + \boldsymbol{u}_2(\tau)\lambda\right)\mathrm{d}\tau \\ &= \underbrace{\left(\mathrm{e}^{\mathbf{A}t'}\boldsymbol{x}_0 + \int_0^{t'} \mathrm{e}^{\mathbf{A}(t'-\tau)}\mathbf{B}\boldsymbol{u}_1(\tau)\mathrm{d}\tau\right)}_{=:\,\boldsymbol{x}_1(t')}(1-\lambda) + \underbrace{\left(\mathrm{e}^{\mathbf{A}t'}\boldsymbol{x}_0 + \int_0^{t'} \mathrm{e}^{\mathbf{A}(t'-\tau)}\mathbf{B}\boldsymbol{u}_2(\tau)\mathrm{d}\tau\right)}_{=:\,\boldsymbol{x}_2(t')}\lambda,\end{aligned}$$

showing that the set $\mathcal{R}$ is convex as well since every point between x_1 and x_2 belongs to the reachable set $\mathcal{R}$. Again, monotonicity of the output map (2.2b) transfers the result to the output space.

The above discussion implies that Theorem 5 is applicable to constrained linear systems without introducing any approximation errors in the computation of the redundantly reachable output sets $\mathcal{Q}_{\mathcal{Z}_c}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}_c}$ for constant input faults.

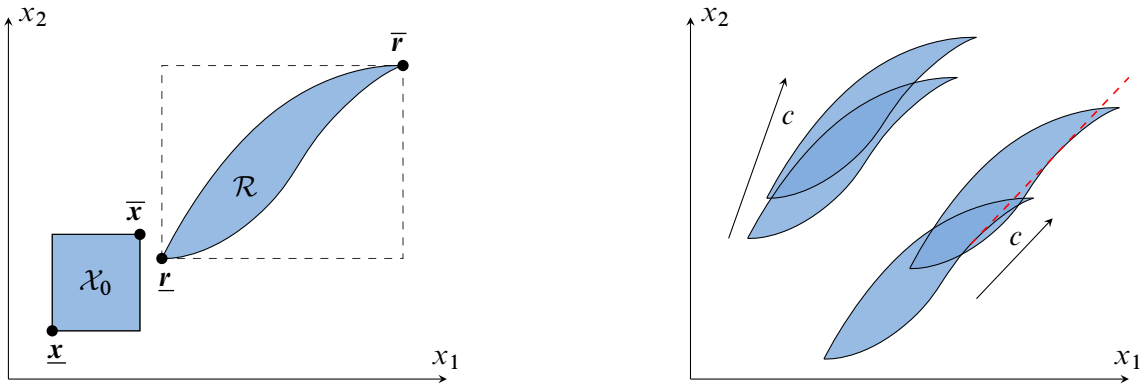
2.2.2.2 Nonlinear Systems

Broadening the class of considered systems to systems (2.1), or (2.7), respectively, has an impact on the system properties that we have made use of in Section 2.2.2.1 to show the satisfaction of the conditions from Theorem 5. In general, convexity of reachable sets is not preserved under nonlinear dynamics. Furthermore, the state trajectory of a general nonlinear system is, without further assumptions, not monotone in a constant input u_i at time t w.r.t. Definition 11. The lack of these properties plays a role if the set of fault scenarios is given by $\mathcal{Z}_\mathrm{c}$. Then, we cannot rely on the reachable sets $\mathcal{R}_{\boldsymbol{y}}^{\underline{u}_i}$ and $\mathcal{R}_{\boldsymbol{y}}^{\overline{u}_i}$, as done in Theorem 5, because these sets do not mark the extremal reachable sets used in the argument of the respective proof. To find the reachable sets shaping $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$ or $\hat{\mathcal{Q}}_{\mathcal{Z}_\mathrm{c}}$ is therefore a more difficult task in the nonlinear case. Furthermore, sticking to the nomenclature of the proof for Theorem 5, it is well possible that the parameter λ defining such an extremal reachable set differs for every time point $t > 0$. This again renders the computation of $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}_\mathrm{c}}$ infeasible since infinitely many reachable sets need to be computed. This situation could only be avoided by switching to discrete-time systems. Finally, note that, if $\mathcal{Z}$ has only finitely many elements, no challenges are induced by the lack of monotonicity or convexity.

We will now introduce monotone systems as a system class for which there is *a good chance*, that the conditions of the Theorem 5 are fulfilled, and, therefore we can reasonably simplify the computation of $\mathcal{Q}_{\mathcal{Z}_\mathrm{c}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}_\mathrm{c}}$. This system class has strong similarities to *positive systems*, as outlined by Benvenuti and Farina [18]. The latter maintain positive values of the system states, given that the respective initial values are positive. As said before, monotone systems are similar, but are not restricted to positive state values. They have the property of preserving a so-called partial order in state space [105]. Hence, they embody a somewhat wider class of systems. Monotone systems have first been introduced by Angeli and Sontag [9] and have been studied by various authors since then, e.g., by De Leenheer et al. [42], Meyer et al. [105], Ramdani et al. [137], Rantzer and Bernhardsson [138], and Smith [154]. Among them, many results have been obtained in the field of reachability. The concepts of monotone systems have also been extended to more general nonlinear systems, e.g., by Chu and Huang [34], Coogan [38], and Yang and Ozay [176]. Examples for monotone dynamic systems can be found in chemical and process engineering [42] as well as in evolutionary dynamics [138].

Before defining monotonicity, partial orders $\preceq_{\boldsymbol{x}}$ and $\preceq_{\boldsymbol{u}}$ need to be introduced. This is to make points in the state and input space comparable. Referring to [105], we define equivalence between the relations $\boldsymbol{v}^1 \preceq_{\boldsymbol{v}} \boldsymbol{v}^2$ and $\boldsymbol{v}^2 - \boldsymbol{v}^1 \in \mathcal{K}_{\boldsymbol{v}}$, where $\boldsymbol{v}^1,\ \boldsymbol{v}^2 \in \mathbb{R}^{n_\mathrm{v}}$, $n_\mathrm{v} \in \mathbb{N}$, and $\mathcal{K}_{\boldsymbol{v}} \subseteq \mathbb{R}^{n_\mathrm{v}}$ denotes a convex and pointed cone. This cone has the properties $\alpha\mathcal{K}_{\boldsymbol{v}} \subseteq \mathcal{K}_{\boldsymbol{v}}$ for any $\alpha \geq 0$, $\mathcal{K}_{\boldsymbol{v}} + \mathcal{K}_{\boldsymbol{v}} \subseteq \mathcal{K}_{\boldsymbol{v}}$, and $\mathcal{K}_{\boldsymbol{v}} \cap (-\mathcal{K}_{\boldsymbol{v}}) = \{\mathbf{0}_{n_\mathrm{v}}\}$. Hence, any vector within the cone scaled by a non-negative factor is an element of the cone, i.e. $\alpha\boldsymbol{v} \in \mathcal{K}_{\boldsymbol{v}}\ \forall\ \boldsymbol{v} \in \mathcal{K}_{\boldsymbol{v}}$. Equally, the sum of two members of the cone is an element of the cone, i.e. $\boldsymbol{v}^1 + \boldsymbol{v}^2 \in \mathcal{K}_{\boldsymbol{v}}\ \forall\ \boldsymbol{v}^1, \boldsymbol{v}^2 \in \mathcal{K}_{\boldsymbol{v}}$. Hence, two convex and pointed cones $\mathcal{K}_{\boldsymbol{x}} \subseteq \mathbb{R}^n$ and $\mathcal{K}_{\boldsymbol{u}} \subseteq \mathbb{R}^m$ define the partial orders $\preceq_{\boldsymbol{x}}$ and $\preceq_{\boldsymbol{u}}$, respectively.

For our purposes, we assume that the plant model is present in a form such that the cones $\mathcal{K}_{\boldsymbol{x}}$ and $\mathcal{K}_{\boldsymbol{u}}$ correspond to orthants of the state and input space, respectively. An orthant $\mathcal{O}$ of $\mathbb{R}^{n_\mathrm{v}}$ is defined as the cartesian product of half-spaces. I.e., $\mathcal{O} = \bigtimes_{\ell=1}^{n_\mathrm{v}} \mathbb{H}_\ell$, where $\mathbb{H}_\ell \in \{\mathbb{R}_{\geq 0}, \mathbb{R}_{\leq 0}\}$



(a) Illustration of a reachable set $\mathcal{R}$ of a monotone system w.r.t. Definition 12.

(b) Shifting of the reachable set $\mathcal{R}$.

Figure 2.3: Illustrations for monotone systems.

and we denote by $\mathbb{R}_{\geq 0}$, or $\mathbb{R}_{\leq 0}$, the set of non-negative or non-positive real numbers, respectively. Hence, for each component v_i^j of some $\boldsymbol{v}^j \in \mathbb{R}^{n_v}$, $i \in \mathcal{I}_{1,n_v}$, $j \in \{1, 2\}$, the partial order $\leq_{\boldsymbol{v}}$ implies $v_i^1 \leq_{\boldsymbol{v}} v_i^2 \Leftrightarrow v_i^1 \leq v_i^2$ for $\mathbb{H}_i = \mathbb{R}_{\geq 0}$, and $v_i^1 \leq_{\boldsymbol{v}} v_i^2 \Leftrightarrow v_i^1 \geq v_i^2$ for $\mathbb{H}_i = \mathbb{R}_{\leq 0}$. Without loss of generality, the plant model can be transformed such that it is given w.r.t. the standard partial order, defined by the cones $\mathcal{K}_{\boldsymbol{x}} = \mathbb{R}^n_{\geq 0}$ and $\mathcal{K}_{\boldsymbol{u}} = \mathbb{R}^m_{\geq 0}$ [38].

Having introduced the concept of partial orders, we can now introduce the definition of a monotone dynamic system.

Definition 12 (see Angeli and Sontag [9, Def. II.1]). *Let $\boldsymbol{x}_0^1, \boldsymbol{x}_0^2 \in \mathcal{X}$ and $\boldsymbol{u}^1, \boldsymbol{u}^2 \colon [0; t) \to \mathcal{U}$. A dynamic system of the form* (2.1) *is monotone w.r.t. the partial order defined by the cones $\mathcal{K}_{\boldsymbol{x}} \subset \mathbb{R}^n$ and $\mathcal{K}_{\boldsymbol{u}} \subset \mathbb{R}^m$ if the implication*

$$\boldsymbol{x}_0^1 \leq_{\boldsymbol{x}} \boldsymbol{x}_0^2, \quad \boldsymbol{u}^1(t) \leq_{\boldsymbol{u}} \boldsymbol{u}^2(t) \quad \Rightarrow \quad \boldsymbol{\Phi}(\boldsymbol{x}_0^1, \boldsymbol{u}^1, t) \leq_{\boldsymbol{x}} \boldsymbol{\Phi}(\boldsymbol{x}_0^2, \boldsymbol{u}^2, t)$$

holds for all times $t \geq 0$.

Theorem 6 (see Angeli and Sontag [9, Proposition III.2]). *A dynamic system of the form* (2.1) *is monotone w.r.t. the standard partial order defined by the cones $\mathcal{K}_{\boldsymbol{x}} = \mathbb{R}^n_{\geq 0}$ and $\mathcal{K}_{\boldsymbol{u}} = \mathbb{R}^m_{\geq 0}$ if and only if*

$$\frac{\partial f_i(\boldsymbol{x}, \boldsymbol{u})}{\partial x_j} \geq 0 \qquad \forall\, i, j \in \mathcal{I}_{1,n}, \qquad i \neq j$$
$$\frac{\partial f_i(\boldsymbol{x}, \boldsymbol{u})}{\partial u_j} \geq 0 \qquad \forall\, i \in \mathcal{I}_{1,n}, \qquad j \in \mathcal{I}_{1,m}$$

for all $\boldsymbol{x} \in \mathcal{X}$, $\boldsymbol{u} \in \mathcal{U}$. Here, we denote the i-th component of $\boldsymbol{f}$ by f_i and the j-th component of $\boldsymbol{x}$ by x_j.

A characteristic property of monotone systems is depicted in Fig. 2.3a. If the initial set $\mathcal{X}_0$ and the input bounds are given by intervals, i.e. $\mathcal{X}_0 = [\underline{\boldsymbol{x}}; \overline{\boldsymbol{x}}]$ and $\mathcal{U} = [\underline{\boldsymbol{u}}; \overline{\boldsymbol{u}}]$, then every reachable

set $\mathcal{R}(\mathcal{X}_0,\, t)$ is bounded by an interval $[\underline{\boldsymbol{r}};\, \overline{\boldsymbol{r}}]$ for any $t \geq 0$, where the lower and upper bounds are given by $\underline{\boldsymbol{r}} = \boldsymbol{\Phi}(\underline{\boldsymbol{x}}, \underline{\boldsymbol{u}}, t)$ and $\overline{\boldsymbol{r}} = \boldsymbol{\Phi}(\overline{\boldsymbol{x}}, \overline{\boldsymbol{u}}, t)$, respectively. Assuming that the depicted reachable set is dependent on a fault parameter c in the actuator u_i, all points from $\mathcal{R}$ are shifted monotonously through the phase plane, see Fig. 2.2a. Further assuming the standard partial order for Fig. 2.3a, this corresponds to a shift of all reachable points towards the upper right.

Monotone systems w.r.t. Definition 12 fulfill a stronger monotonicity property than systems that are only monotone w.r.t. Definition 11. This holds because the partial order that is maintained by the states is constant over time. In Definition 11, the underlying partial order may change over time. Hence, monotone systems sufficiently fulfill the first important requirement of Theorem 5.

Clearly, there is no guarantee that the reachable sets $\mathcal{R}(\boldsymbol{x}_0,\, t)$ of monotone systems are convex, which is the reason why Theorem 5 does not strictly apply for this system class. However, depending on the specific system, the shape of their reachable sets and their shifting in dependence on the parameter $u_i = c$, one might still be able to compute $\mathcal{Q}_{\mathcal{Z}_c}$, $\hat{\mathcal{Q}}_{\mathcal{Z}_c}$ by evaluating only the extremal fault scenarios given by $\mathcal{R}_{\boldsymbol{y}}^{\underline{u}_i}$ and $\mathcal{R}_{\boldsymbol{y}}^{\overline{u}_i}$. In Fig. 2.3b, two possible scenarios are depicted. On the upper left side, the shift happens such that the intersection monotonously decreases in size while the parameter c increases. The procedure of Theorem 5 is applicable. In the lower right scenario, the shifting direction causes errors when only considering the extremal fault scenarios for computing $\mathcal{Q}_{\mathcal{Z}_c}$ or $\hat{\mathcal{Q}}_{\mathcal{Z}_c}$. This is due to the non-convexity of $\mathcal{R}$. As c increases, all points below the red dashed line are cut off which is erroneously omitted by intersection of the extremal reachable sets only. To improve the results, more fault cases, i.e. more instances of the parameter c, could be taken into account.[2]

2.2.3 Computation Algorithms for $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$

We will next focus on how to transfer the theoretical properties discussed so far to the numeric computation of redundantly reachable outputs. Numerical reachability analysis is a broad research field itself. Many contributions have been delivered by Althoff, see e.g. [5, 6, 7, 8, 83], where results for improving computational efficiency are one important pillar of the presented work. Furthermore, strategies for inner and outer approximating reachable sets for nonlinear systems have been proposed in [83] and by Goubault and Putot [56] and Ramdani et al. [137]. Reißig [140] investigates conditions for reachable sets to be convex at a given time point t.

For explicitly computing $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$, we switch to a discrete-time analysis now. Numerically describing reachable sets for a continuous-time system is possible, but needs approximation techniques for condensing the results such that they can be stored using finitely many parameters [5]. To avoid this, the system dynamics of Σ_z is changed to

$$\boldsymbol{x}_{z,k+1} = \bar{\boldsymbol{f}}_{z,k}(\boldsymbol{x}_{z,k},\, \boldsymbol{u}_{z,k}), \qquad \boldsymbol{x}_{z,0} = \boldsymbol{x}_0 \tag{2.16a}$$

$$\boldsymbol{y}_{z,k} = \bar{\boldsymbol{g}}_{z,k}(\boldsymbol{x}_{z,k},\, \boldsymbol{u}_{z,k}), \tag{2.16b}$$

[2] In this paragraph, we have argued in state space and not in the output space. The arguments can be transformed assuming an output function that is monotone for all points in the reachable sets.

for nonlinear systems and

$$\boldsymbol{x}_{z,k+1} = \bar{\mathbf{A}}_{z,k}\boldsymbol{x}_{z,k} + \bar{\mathbf{B}}_{z,k}\boldsymbol{u}_{z,k}, \qquad \boldsymbol{x}_{z,0} = \boldsymbol{x}_0 \tag{2.17a}$$

$$\boldsymbol{y}_{z,k} = \bar{\mathbf{C}}_{z,k}\boldsymbol{x}_{z,k} + \bar{\mathbf{D}}_{z,k}\boldsymbol{u}_{z,k} \tag{2.17b}$$

for systems with linear dynamics. We denote the respective state and input sequences as $\boldsymbol{x}_{z,k} = \boldsymbol{x}_z(kT_\mathrm{s})$ and $\boldsymbol{u}_{z,k} = \boldsymbol{u}_z(kT_\mathrm{s})$ where $T_\mathrm{s} > 0$ is the sampling time and the input signal $\boldsymbol{u}_z$ has been assumed to be constant during one sample interval. The time-varying constraint sets are analogously denoted as $\mathcal{X}_z(kT_\mathrm{s}) = \mathcal{X}_{z,k}$ and $\mathcal{U}_z(kT_\mathrm{s}) = \mathcal{U}_{z,k}$, respectively. For the ease of presentation, we additionally assume $\frac{\partial \bar{\boldsymbol{g}}_{z,k}(\boldsymbol{x}_z,\boldsymbol{u}_z)}{\partial \boldsymbol{u}_z} = \mathbf{0}$ for all $\boldsymbol{x}_z \in \mathcal{X}_{z,k}$, $\boldsymbol{u}_z \in \mathcal{U}_{z,k}$, or $\bar{\mathbf{D}}_{z,k} = \mathbf{0}$, respectively, for all $z \in \mathcal{Z}$. The steps for the computation of the sequence $\mathcal{Q}_{\mathcal{Z},k}$ of length $N \in \mathbb{N}$, i.e. $k \in \mathcal{I}_{0,N}$, are presented in Algorithm 1. For computing the set sequence $\hat{\mathcal{Q}}_{\mathcal{Z},k}$, Algorithm 2 is provided. The components of both algorithms are explained in the sequel.

Algorithm 1 Computation of the sequence $\mathcal{Q}_{\mathcal{Z},k}$.

Input: $\mathcal{Z}$, N, $\boldsymbol{x}_0 \in \bigcap_{z\in\mathcal{Z}} \mathcal{X}_{z,0}$
$k \leftarrow 0$
for all $z \in \mathcal{Z}$ **do**
    $\mathcal{R}_0^z \leftarrow \{\boldsymbol{x}_0\}$
    Compute $\mathcal{R}_{\boldsymbol{y},0}^z$ from $\mathcal{R}_0^z$
end for
while $k < N$ **do**
    for all $z \in \mathcal{Z}$ **do**
        Compute $\mathcal{R}_{k+1}^z$ from $\mathcal{R}_k^z$
        Compute $\mathcal{R}_{\boldsymbol{y},k+1}^z$ from $\mathcal{R}_k^z$
    end for
    $k \leftarrow k + 1$
end while
for all $k \in \mathcal{I}_{0,N}$ **do**
    $\mathcal{Q}_{\mathcal{Z},k} \leftarrow \bigcap_{z\in\mathcal{Z}} \mathcal{R}_{\boldsymbol{y},k}^z$
end for
Output: $\mathcal{Q}_{\mathcal{Z},k}\ \forall\, k \in \mathcal{I}_{0,N}$

First computing the complete sequence of state reachable sets and performing the mapping to the output space and intersection afterwards, as done in Algorithm 1, results in the redundantly reachable outputs according to Definition 9. Here, we only obtain a statement that there exist admissible input sequences $\boldsymbol{u}_{z,k}$ such that for one specific $\tilde{k} \in \mathcal{I}_{0,N}$, the outputs $\boldsymbol{y}_{z,k}$ can be driven to the set $\mathcal{Q}_{\mathcal{Z},\tilde{k}}$. However, no statement can be made about the other time steps $k \neq \tilde{k}$, i.e., whether there exists an input driving $\boldsymbol{y}_{z,k+1}$ into the set $\mathcal{Q}_{\mathcal{Z},k+1}$, given that $\boldsymbol{y}_{z,k} \in \mathcal{Q}_{\mathcal{Z},k}$.

In Algorithm 2, the usage of the preimage set $\mathcal{P}_k^z$ as initial set for the next time step is crucial. It contains all reachable states that are mapped via Eq. (2.16b) or (2.17b) to $\hat{\mathcal{Q}}_{\mathcal{Z},k}$. In this way, we make sure that, for every scenario $z \in \mathcal{Z}$, there is an admissible input sequence $\boldsymbol{u}_{z,k}$ that keeps the system output $\boldsymbol{y}_{z,k}$ in the redundantly output reachable set $\hat{\mathcal{Q}}_{\mathcal{Z},k}$ for all time steps $k \in \mathcal{I}_{0,N}$. I.e., $\forall\, z \in \mathcal{Z}\ \exists\, \boldsymbol{u}_{z,k} \in \mathcal{U}_{z,k}$ such that $\boldsymbol{y}_{z,k} \in \hat{\mathcal{Q}}_{\mathcal{Z},k}\ \forall\, k \in \mathcal{I}_{0,N}$, as it is demanded by Definition 10.

Algorithm 2 Computation of the sequence $\hat{\mathcal{Q}}_{\mathcal{Z},k}$.

Input: $\mathcal{Z}$, N, $\boldsymbol{x}_0 \in \bigcap_{z\in\mathcal{Z}} \mathcal{X}_{z,0}$
$k \leftarrow 0$
for all $z \in \mathcal{Z}$ **do**
    $\mathcal{P}^z_0 \leftarrow \{\boldsymbol{x}_0\}$ ▷ Initialize the preimage sets in state space.
    Compute $\mathcal{R}^z_{\boldsymbol{y},0}$ from $\mathcal{P}^z_0$
end for
$\hat{\mathcal{Q}}_{\mathcal{Z},0} \leftarrow \bigcap_{z\in\mathcal{Z}} \mathcal{R}^z_{\boldsymbol{y},0}$
while $k < N$ **do**
    for all $z \in \mathcal{Z}$ **do**
        Compute $\mathcal{R}^z_{k+1}$ from $\mathcal{P}^z_k$
        Compute $\mathcal{R}^z_{\boldsymbol{y},k+1}$ from $\mathcal{P}^z_k$
    end for
    $\hat{\mathcal{Q}}_{\mathcal{Z},k+1} \leftarrow \bigcap_{z\in\mathcal{Z}} \mathcal{R}^z_{\boldsymbol{y},k+1}$
    if $k+1 = N$ **then**
        break
    end if
    for all $z \in \mathcal{Z}$ **do**
        Compute preimage $\mathcal{P}^z_{k+1} \subseteq \mathcal{R}^z_{k+1}$ of $\hat{\mathcal{Q}}_{\mathcal{Z},k+1}$
    end for
    $k \leftarrow k+1$
end while
Output: $\hat{\mathcal{Q}}_{\mathcal{Z},k}\ \forall\ k \in \mathcal{I}_{0,N}$

2.2.3.1 Linear Systems

We will now show explicitly how the steps from Algorithms 1 and 2 can be put into practice for systems with linear dynamics given by Eq. (2.17). An important aspect when numerically computing reachable sets is the choice of the set *representation*. It has a large impact on the computational efficiency, but also influences the mathematical properties that can be used when working with these sets. For example, *zonotopes* are a special type of polyhedral sets [5]. They can be propagated easily in time for linear systems by a simple matrix multiplication. However, they are not *closed* under intersection operations. The latter is an important property for us as we want to present an algorithm, that computes the set of redundantly reachable outputs without inducing approximation errors. A set representation $\mathcal{C}$ is *closed* under a set operation if the result of this operation applied to two instances of type $\mathcal{C}$ can be depicted by a set of type $\mathcal{C}$. For each fault scenario $z \in \mathcal{Z}$, we therefore rely on choosing $\mathcal{X}_{z,k}$ and $\mathcal{U}_{z,k}$ as polyhedral sets[3]

$$\mathcal{X}_{z,k} = \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{H}^{\boldsymbol{x}}_{z,k}\boldsymbol{x} \leq \boldsymbol{h}^{\boldsymbol{x}}_{z,k}\right\} \tag{2.18a}$$

$$\mathcal{U}_{z,k} = \left\{\boldsymbol{u} \in \mathbb{R}^m \,|\, \mathbf{H}^{\boldsymbol{u}}_{z,k}\boldsymbol{u} \leq \boldsymbol{h}^{\boldsymbol{u}}_{z,k}\right\}, \tag{2.18b}$$

since this representation is closed under intersection, as it is needed for computing $\mathcal{Q}_{\mathcal{Z}}$. This comes at the cost of being numerically more expensive compared to, e.g., using zonotopes.

[3] The representation of polyhedral sets used here is called *half-space representation*, or $\mathcal{H}$-representation.

We now compute the one-step reachable set $\mathcal{R}^z_{k+1}$ from some polyhedral set

$$\mathcal{P}^z_k = \left\{ \boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{H}^{\mathcal{P}}_{z,k} \boldsymbol{x} \leq \boldsymbol{h}^{\mathcal{P}}_{z,k} \right\} \subseteq \mathcal{X}_{z,k}, \tag{2.19}$$

where the initial set is given by $\mathcal{P}^z_0 = \left\{ \boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{H}^{\mathcal{P}}_{z,0} \boldsymbol{x} \leq \boldsymbol{h}^{\mathcal{P}}_{z,0} \right\}$ with $\mathbf{H}^{\mathcal{P}}_{z,0} = \begin{bmatrix} \mathbf{I}_n & -\mathbf{I}_n \end{bmatrix}^\top$ and $\boldsymbol{h}^{\mathcal{P}}_{z,0} = \begin{bmatrix} \boldsymbol{x}_0^\top & -\boldsymbol{x}_0^\top \end{bmatrix}^\top$. This is an intermediate step towards the computation of $\mathcal{R}^z_{\boldsymbol{y},k+1}$. We obtain $\mathcal{R}^z_{k+1}$ by solving the system dynamics (2.17a) for $\boldsymbol{x}_{z,k}$ and inserting it into Eq. (2.19). This yields

$$\mathcal{R}^z_{k+1} = \left\{ \boldsymbol{x}_{z,k+1} \in \mathcal{X}_{z,k+1} \,\middle|\, \exists\, \boldsymbol{u}_{z,k} \in \mathcal{U}_{z,k} \text{ s.t. } \mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} (\boldsymbol{x}_{z,k+1} - \bar{\mathbf{B}}_{z,k} \boldsymbol{u}_{z,k}) \leq \boldsymbol{h}^{\mathcal{P}}_{z,k} \right\},$$

which can be rewritten as

$$\mathcal{R}^z_{k+1} = \left\{ \boldsymbol{x}_{z,k+1} \in \mathbb{R}^n \,\middle|\, \exists\, \boldsymbol{u}_{z,k} \in \mathbb{R}^m \text{ s.t. } \begin{bmatrix} \mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} & -\mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} \bar{\mathbf{B}}_{z,k} \\ \mathbf{H}^{\boldsymbol{x}}_{z,k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}^{\boldsymbol{u}}_{z,k} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{z,k+1} \\ \boldsymbol{u}_{z,k} \end{bmatrix} \leq \begin{bmatrix} \boldsymbol{h}^{\mathcal{P}}_{z,k} \\ \boldsymbol{h}^{\boldsymbol{x}}_{z,k+1} \\ \boldsymbol{h}^{\boldsymbol{u}}_{z,k} \end{bmatrix} \right\} \tag{2.20a}$$

$$=: \left\{ \boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{H}^{\mathcal{R}}_{z,k+1} \boldsymbol{x} \leq \boldsymbol{h}^{\mathcal{R}}_{z,k+1} \right\} \tag{2.20b}$$

by inserting Eq. (2.18). We have implicitly assumed invertibility of the system matrix $\bar{\mathbf{A}}_{z,k}$. This is a valid assumption for discretized continuous plants because no eigenvalues $\lambda \in \mathbb{C}$ of $\bar{\mathbf{A}}_{z,k}$ at $\lambda = 0$ can be obtained by discretization. These would relate to infinitely fast eigenvalues of the continuous plant model, which is not physical.

The transformation of $\mathcal{R}^z_{k+1}$ to $\mathbb{R}^p$ via Eq. (2.17b) can be done in various ways. One way is to compute its vertex-representation as an intermediate step, because linear maps can easily be computed for this representation. However, this comes at the cost of performing the necessary transfers between the different representations. Hence, we choose to incorporate the transformation directly in the $\mathcal{H}$-representation. Making the common assumption for each output matrix $\bar{\mathbf{C}}_{z,k}$ to have rank p and making use of $\bar{\mathbf{D}}_{z,k} = \mathbf{0}$, we can solve Eq. (2.17b) as

$$\boldsymbol{x}_{z,k+1} = \begin{bmatrix} \bar{\mathbf{C}}^+_{z,k+1} & \bar{\mathbf{C}}^\perp_{z,k+1} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_{z,k+1} \\ \boldsymbol{\xi}_{z,k+1} \end{bmatrix}$$

with $\bar{\mathbf{C}}^+_{z,k+1} = \bar{\mathbf{C}}^\top_{z,k+1} (\bar{\mathbf{C}}_{z,k+1} \bar{\mathbf{C}}^\top_{z,k+1})^{-1}$, $\mathrm{im}(\bar{\mathbf{C}}^\perp_{z,k+1}) = \ker(\bar{\mathbf{C}}_{z,k+1})$ and $\boldsymbol{\xi}_{z,k+1} \in \mathbb{R}^{n-p}$. Hence, in Eq. (2.20a), we can express

$$\begin{bmatrix} \boldsymbol{x}_{z,k+1} \\ \boldsymbol{u}_{z,k} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{C}}^+_{z,k+1} & \bar{\mathbf{C}}^\perp_{z,k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_m \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_{z,k+1} \\ \boldsymbol{\xi}_{z,k+1} \\ \boldsymbol{u}_{z,k} \end{bmatrix},$$

leading to the output reachable set

$$\mathcal{R}^z_{\boldsymbol{y},k+1} = \left\{ \boldsymbol{y}_{z,k+1} \in \mathbb{R}^p \,\middle|\, \exists \begin{bmatrix} \boldsymbol{\xi}_{z,k+1} \\ \boldsymbol{u}_{z,k} \end{bmatrix} \in \mathbb{R}^{n-p} \times \mathbb{R}^m \text{ s.t. } \begin{bmatrix} \mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} \bar{\mathbf{C}}^+_{z,k+1} & \mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} \bar{\mathbf{C}}^\perp_{z,k+1} & -\mathbf{H}^{\mathcal{P}}_{z,k} \bar{\mathbf{A}}^{-1}_{z,k} \bar{\mathbf{B}}_{z,k} \\ \mathbf{H}^{\boldsymbol{x}}_{z,k+1} \bar{\mathbf{C}}^+_{z,k+1} & \mathbf{H}^{\boldsymbol{x}}_{z,k+1} \bar{\mathbf{C}}^\perp_{z,k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{H}^{\boldsymbol{u}}_{z,k} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_{z,k+1} \\ \boldsymbol{\xi}_{z,k+1} \\ \boldsymbol{u}_{z,k} \end{bmatrix} \leq \begin{bmatrix} \boldsymbol{h}^{\mathcal{P}}_{z,k} \\ \boldsymbol{h}^{\boldsymbol{x}}_{z,k+1} \\ \boldsymbol{h}^{\boldsymbol{u}}_{z,k} \end{bmatrix} \right\}. \tag{2.21}$$

We denote $\mathcal{R}^z_{\boldsymbol{y},k+1}$ in the compact form $\mathcal{R}^z_{\boldsymbol{y},k+1} =: \left\{\boldsymbol{y} \in \mathbb{R}^p \,|\, \mathbf{H}^{\boldsymbol{y}}_{z,k+1}\boldsymbol{y} \leq \boldsymbol{h}^{\boldsymbol{y}}_{z,k+1}\right\}$. Using this notation, we can compute the redundantly reachable outputs at step $k+1$. It is given by

$$\mathcal{Q}_{\mathcal{Z},k+1} = \bigcap_{z\in\mathcal{Z}} \mathcal{R}^z_{\boldsymbol{y},k+1} = \bigcap_{z\in\mathcal{Z}} \left\{\boldsymbol{y} \in \mathbb{R}^p \,|\, \mathbf{H}^{\boldsymbol{y}}_{z,k+1}\boldsymbol{y} \leq \boldsymbol{h}^{\boldsymbol{y}}_{z,k+1}\right\} \tag{2.22a}$$

$$=: \left\{\boldsymbol{y} \in \mathbb{R}^p \,|\, \mathbf{H}^{\mathcal{Q}}_{k+1}\boldsymbol{y} \leq \boldsymbol{h}^{\mathcal{Q}}_{k+1}\right\}, \tag{2.22b}$$

where $\mathbf{H}^{\mathcal{Q}}_{k+1}$ and $\boldsymbol{h}^{\mathcal{Q}}_{k+1}$ can be obtained by vertically stacking the quantities $\mathbf{H}^{\boldsymbol{y}}_{z,k+1}$ and $\boldsymbol{h}^{\boldsymbol{y}}_{z,k+1}$, respectively, for all $z \in \mathcal{Z}$. Analogously, for $\hat{\mathcal{Q}}_{\mathcal{Z},k+1}$, we obtain $\hat{\mathcal{Q}}_{\mathcal{Z},k+1} =: \{\boldsymbol{y} \in \mathbb{R}^p | \mathbf{H}^{\hat{\mathcal{Q}}}_{k+1}\boldsymbol{y} \leq \boldsymbol{h}^{\hat{\mathcal{Q}}}_{k+1}\}$. This completes steps 9, 10, and 15 of Algorithm 1, as well as 7, 10, 11, and 13 of Algorithm 2.

In step 18 of Algorithm 2, we need to determine the respective preimage sets $\mathcal{P}^z_{k+1}$ of each fault scenario in order to obtain the reachable states corresponding to redundant outputs. These points will be considered as the initial sets for computing the next time step. Due to the assumption $\bar{\mathbf{D}}_{z,k} = \mathbf{0}$, this transformation is simple when using polyhedral sets in $\mathcal{H}$-representation. Substituting $\boldsymbol{y}$ by $\bar{\mathbf{C}}_{z,k+1}\boldsymbol{x}$ in the description of $\hat{\mathcal{Q}}_{\mathcal{Z},k+1}$, we obtain

$$\begin{aligned}
\mathcal{P}^z_{k+1} &= \left\{\boldsymbol{x} \in \mathcal{R}^z_{k+1} \,|\, \mathbf{H}^{\hat{\mathcal{Q}}}_{k+1}\bar{\mathbf{C}}_{z,k+1}\boldsymbol{x} \leq \boldsymbol{h}^{\hat{\mathcal{Q}}}_{k+1}\right\} \\
&= \left\{\boldsymbol{x} \in \mathbb{R}^n \,\middle|\, \begin{bmatrix} \mathbf{H}^{\hat{\mathcal{Q}}}_{k+1}\bar{\mathbf{C}}_{z,k+1} \\ \mathbf{H}^{\mathcal{R}}_{z,k+1} \end{bmatrix} \boldsymbol{x} \leq \begin{bmatrix} \boldsymbol{h}^{\hat{\mathcal{Q}}}_{k+1} \\ \boldsymbol{h}^{\mathcal{R}}_{z,k+1} \end{bmatrix} \right\} \\
&= \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{H}^{\mathcal{P}}_{z,k+1}\boldsymbol{x} \leq \boldsymbol{h}^{\mathcal{P}}_{z,k+1}\right\},
\end{aligned}$$

where the quantities $\mathbf{H}^{\mathcal{R}}_{z,k+1}$ and $\boldsymbol{h}^{\mathcal{R}}_{z,k+1}$ are taken from Eq. (2.20a).

Remark 4. In Eq. (2.20a), we have made use of a projection from the *augmented* state space $\mathbb{R}^n \times \mathbb{R}^m$ onto $\mathbb{R}^n$. This becomes necessary as the reachable set $\mathcal{R}^z_{k+1}$ only contains points in state space, but is dependent on admissible combinations of state and input signals. A similar situation has occurred in Eq. (2.21). Such projections can be performed numerically by, e.g., applying Fourier elimination methods (Keerthi and Sridharan [79]) or methods based on parametric optimization (Jones et al. [72]). Overviews can be found in Borrelli et al. [20] and Jones et al. [71]. An efficient implementation is included in the toolbox [62].

Remark 5. In view of the performed set intersections, redundant inequalities in the $\mathcal{H}$-representations of $\mathcal{R}^z_{k+1}$ and $\mathcal{R}^z_{\boldsymbol{y},k+1}$ may occur. This can cause a rapid growth in the numerical complexity over the time steps k which makes the projection operations from the previous remark numerically expensive. Therefore, it it is preferable to find a minimal representation of each reachable set in state and output space. Such a representation is comprised of a minimal amount of inequalities while maintaining the original shape of the set. The same procedure should be applied when establishing the quantities $\mathbf{H}^{\mathcal{Q}}_{k+1}, \boldsymbol{h}^{\mathcal{Q}}_{k+1}, \mathbf{H}^{\hat{\mathcal{Q}}}_{k+1}, \boldsymbol{h}^{\hat{\mathcal{Q}}}_{k+1}, \mathbf{H}^{\mathcal{P}}_{z,k+1}, \boldsymbol{h}^{\mathcal{P}}_{z,k+1}$ that define $\mathcal{Q}_{\mathcal{Z},k+1}$, $\hat{\mathcal{Q}}_{\mathcal{Z},k+1}$, and $\mathcal{P}^z_{k+1}$, respectively. How to identify redundant inequalities in an $\mathcal{H}$-representation has been studied, e.g., by Cheng [32], Goberna et al. [55], and Telgen [156, 157, 158]. The problem of finding a minimal realization is found to be solvable by means of algorithms based on linear programming. Heuristics can be added to accelerate the computations, as proposed by Paulraj et al. [125]. Again, an implementation for computing a minimal representation is included in [62].

Using the above computation algorithms, we are able to determine the sequences $\mathcal{Q}_{\mathcal{Z},k}$ and $\hat{\mathcal{Q}}_{\mathcal{Z},k}$ of redundantly reachable outputs from a common initial state $\boldsymbol{x}_0$ over the time horizon $k \in \mathcal{I}_{0,N}$. Special for the case of linear dynamics and polyhedral constraint sets is that $\mathcal{Q}_{\mathcal{Z},k}$ or $\hat{\mathcal{Q}}_{\mathcal{Z},k}$, respectively, can be computed without inducing approximation errors. This is because reachable sets can be computed exactly for this system class. The assumptions made are mild and do not severely shrink the system class for which the procedure is applicable. Especially, the assumptions $\text{rank}(\bar{\mathbf{C}}_{z,k}) = p$ and $\bar{\mathbf{D}}_{z,k} = \mathbf{0}$ may be dropped. However, this will render the presentation less compact. Systems with linear dynamics possess the monotonicity and convexity properties demanded by Theorem 5. Hence, $\mathcal{Q}_{\mathcal{Z},k}$ and $\hat{\mathcal{Q}}_{\mathcal{Z},k}$ can be computed without approximation errors even if the set of fault scenarios includes the case of infinitely many unknown but constant actuator faults, i.e. $\mathcal{Z}_\text{c} \subseteq \mathcal{Z}$.

2.2.3.2 Nonlinear Systems

Computing a sequence of redundantly reachable outputs is challenging in view of nonlinear systems because numeric set operations become more involved. Generically, reachable sets for nonlinear systems cannot be determined exactly [6], demanding a switch to set approximations. Note that, in the prior section, we have made use of knowing the analytic solution of the system dynamics. This is usually unknown in the nonlinear case.

Depending on the purpose, outer or inner approximations can be used to describe the system behavior. Outer approximations are not of interest for our system analysis scheme as they provide guarantees for states, or outputs, respectively, that *cannot* be reached when applying admissible inputs. Hence, statements can be made, whether regions in state space that are considered unsafe can be avoided under all circumstances. Our aim is, however, to give guarantees for outputs that *can* always be reached. Hence, inner approximations will be needed for our purpose. Computation techniques for inner approximating reachable sets exist, see e.g. Kochdumper and Althoff [83]. Yet, even if the set of fault scenarios $\mathcal{Z}$ is finite, the sets of redundantly reachable outputs $\mathcal{Q}_\mathcal{Z}$ or $\hat{\mathcal{Q}}_\mathcal{Z}$ can only be approximated, generally.

Algorithms 1 and 2 are formulated without implying the underlying system class. Hence, their structure can be directly transferred to the use for nonlinear systems. If not exactly computable, all set operations have to be substituted by inner approximations.

2.2.4 Redundancy Measure

Reachability analysis and, therefore, numeric redundancy analysis is challenging in various ways. The computation of redundantly reachable outputs is computationally expensive, depending on the underlying system class and the set of fault scenarios $\mathcal{Z}$. Furthermore, for systems with large dimensions, illustration and interpretation of the results is difficult, if not infeasible. Evaluating Algorithm 1 or Algorithm 2 for linear systems, we obtain $N + 1$ polyhedral sets describing the sequence of redundant outputs $\mathcal{Q}_{\mathcal{Z},k}$ or $\hat{\mathcal{Q}}_{\mathcal{Z},k}$, respectively. Every one of these polyhedra is parameterized by some pair $\mathbf{H}_k^\mathcal{Q}$, $\boldsymbol{h}_k^\mathcal{Q}$ or $\mathbf{H}_k^{\hat{\mathcal{Q}}}$, $\boldsymbol{h}_k^{\hat{\mathcal{Q}}}$, whose numbers of rows vary depending on k. In view

of nonlinear models, the number of parameters will be even larger, depending on the chosen set representation.

Hence, it is important to provide methods to make the results of redundancy analysis intuitively comprehensible. To this end, we develop a redundancy measure in this section. It should be designed such that it indicates how strongly the redundant region of operation of the system deviates from the nominal system behavior. This information can be condensed in a time-varying number relating the volumes of the nominal reachable sets $\mathcal{R}_{\boldsymbol{y},k}$ to $\mathcal{Q}_{\boldsymbol{z},k}$ or $\hat{\mathcal{Q}}_{\boldsymbol{z},k}$.

2.2.4.1 Existing Approaches

Research concerning the quantification of system properties has been of interest to many researchers. The probably most spread application is the formulation of controllability measures, as it has, e.g., been covered by Litz [94] and Lückel and Müller [95]. These measures are designed to gain a feeling *how good* a mode of a linear time-invariant system is controllable by the control or disturbance inputs. Hence, they support the structural statements on controllability that are given by the well-known rank criteria by, e.g., Kalman, demanding for $\mathcal{W} = \mathbb{R}^n$ in Eq. (2.9).

A similar approach was introduced into the robotics field by Yoshikawa [177, 178], who proposed so-called manipulability measures for robotic arms. The aim was to rate how good the tool center point can move in which directions of the operation region, depending on the current state of the joints. This has been achieved by deriving the jacobian of the robot's nonlinear kinematics and analyzing its singular values, which quantify the ability to move, and the singular vectors, which specify the corresponding directions. In this way, structural information, e.g., on singular joint positions, is enriched by continuous information on how close the robot is situated to a singular joint position. The latter is provided by the singular values that approach zero in the vicinity of a singular position.

A method using controllability measures for redundancy analysis has been developed by Prochazka et al. [131]. It makes use of controllability gramians for linear time-invariant systems [17, p. 195]. Output controllability gramians are computed for the nominal plant as well as for the fault scenario $u_i = 0, i \in \mathcal{I}_{1,m}$. As done in [177, 178], these matrices are decomposed into their singular values. For each actuator, the singular values obtained for the fault scenario are related to the nominal case, yielding a measure of redundancy for this actuator. If the measure is close to one, the actuator has low impact on the outputs. A measure close or equal to zero indicates a severe loss in controllability by the failure of the actuator. While this method is computationally cheap, it suffers from its limitation to linear time-invariant systems without state or input constraints. There is a theoretic extension of gramians to so-called empirical gramians for nonlinear systems, see Himpe [63]. However, the numeric properties of empirical gramians are critical.

2.2.4.2 A New Redundancy Measure Based on Lebesgue Measures

We propose a new measure for quantifying the degree of redundancy based on volumes of sets. This approach seems intuitive as we have already introduced the sets of redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$.[4] As their computation is not limited to systems with linear dynamics, the new measure is applicable for a wider system class than the measures introduced in the prior discussion. We make use of so-called Lebesgue measures that have originally been published by the French mathematician Lebesgue [91]. The idea of these objects is to generalize the measurements of lengths and volumes in high-dimensional metric spaces. In set-based analysis methods, Lebesgue measures are a commonly used tool for quantifying properties of dynamic systems. Examples in the context of fault-diagnosis and attacks on cyber-physical systems are given by Savchenko [145] and Vlahakis et al. [164], respectively.

Definition 13 (Nelson [114, Def. 1.1.1] and Schuller [150, p. 57])**.** *Let $\mathcal{A} \subseteq \mathbb{R}^d$ with $d \in \mathbb{N}$ be a Lebesgue measurable set. Furthermore, consider a sequence of intervals $\mathcal{B}_j = \bigtimes_{i=1}^{d}[a_{ij};\, b_{ij}] \subseteq \mathbb{R}^d$, where $a_{ij},\, b_{ij} \in \mathbb{R}$, $a_{ij} \leq b_{ij}$, and $j \in \mathcal{I}_{1,\infty}$. The Lebesgue measure μ_d of $\mathcal{A}$ is defined as*

$$\mu_d(\mathcal{A}) = \inf \left\{ \sum_{j=1}^{\infty} \prod_{i=1}^{d} (b_{ij} - a_{ij}) \,\middle|\, \mathcal{A} \subseteq \bigcup_{j=1}^{\infty} \mathcal{B}_j \right\}.$$

Although looking bulky, the above definition is a natural approach to defining volumes of sets in high-dimensional spaces. The union of the sequence of intervals $\mathcal{B}_j \in \mathbb{R}^d$ is required to cover the considered set $\mathcal{A}$. The volume of each interval $\mathcal{B}_j$ is given by the product of its side-lengths, i.e. $\mathrm{vol}(\mathcal{B}_j) = \prod_{i=1}^{d}(b_{ij} - a_{ij})$. Hence, the Lebesgue measure $\mu_d(\mathcal{A})$ is defined by the sequence $\mathcal{B}_j$ whose union covers $\mathcal{A}$ and that produces the smallest possible combined volume. Technically, not every set $\mathcal{A} \in \mathbb{R}^d$ is Lebesgue-measurable. Therefore, Definition 13 does not provide a meaningful definition of volume for all arbitrary sets $\mathcal{A}$. This is, however, not a problem for our purposes, as the standard type sets used in this thesis are Lebesgue-measurable. For a more detailed discussion, the interested reader is referred to Nelson [114] or Royden [144].

Before formulating the redundancy measure, we introduce a formal definition of the dimension of a set:

Definition 14. *Let $\mathcal{A} \subseteq \mathbb{R}^d$. We call $\dim(\mathcal{A}) := d_{\mathcal{A}}$ the dimension of $\mathcal{A}$ if there exists an enclosing manifold $\mathcal{M} \subseteq \mathbb{R}^d$ with $\mathcal{A} \subseteq \mathcal{M}$ and an associated diffeomorphism $\Phi\colon \mathcal{M} \to \mathbb{R}^{d_{\mathcal{A}}}$ such that the interior $\mathrm{int}(\Phi(\mathcal{A})) \neq \emptyset$.*

Definition 15. *Let $\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, t)$ denote the output reachable set of the nominal system (2.1) and assume $0 < \mu_p(\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, t))$ for all $t > 0$. Given a set $\mathcal{Z}$ of fault scenarios, the corresponding set of redundantly reachable outputs is denoted by $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0,\, t)$. The degree of redundancy w.r.t. $\mathcal{Z}$ is defined by a tuple $r(\boldsymbol{x}_0,\, t) = (r_1,\, r_2)$ with*

$$r_1(\boldsymbol{x}_0,\, t) := \frac{\mu_v(\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0,\, t))}{\mu_v\big(\mathrm{proj}(\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, t),\, \mathcal{N}(\boldsymbol{x}_0,\, t))\big)}, \qquad t > 0 \tag{2.23}$$

[4] In the sequel, we formulate all results in terms of $\mathcal{Q}_{\mathcal{Z}}$. It can be substituted by $\hat{\mathcal{Q}}_{\mathcal{Z}}$, depending on the purpose of the analysis.

and

$$r_2(\boldsymbol{x}_0,\, t) := \dim(\mathcal{R}_{\boldsymbol{y}}) - \dim(\mathcal{Q}_{\mathcal{Z}}),$$

where $\mathcal{N}$ *is the lowest-dimensional sub-manifold of* $\mathbb{R}^p$ *containing* $\mathcal{Q}_{\mathcal{Z}}$ *with* $\nu := \dim(\mathcal{N}) \leq p$. *The operator* $\mathrm{proj}(\mathcal{R}_{\boldsymbol{y}},\, \mathcal{N})$ *denotes the projection of* $\mathcal{R}_{\boldsymbol{y}}$ *onto* $\mathcal{N}$.

Due to $\mathcal{Q}_{\mathcal{Z}} \subseteq \mathcal{R}_{\boldsymbol{y}}$, we have $r_1 \in [0;\, 1]$. For the case $r_2 = 0$, $r_1 \approx 1$ relates to a high degree of redundancy, as there is only a small decrease in volume of $\mathcal{Q}_{\mathcal{Z}}$ w.r.t. $\mathcal{R}_{\boldsymbol{y}}$. The lower r_1, the larger the loss in maneuverability of the outputs in the fault cases contained in $\mathcal{Z}$. If $r_2 > 0$, we have a loss in dimension of $\mathcal{Q}_{\mathcal{Z}}$ compared to $\mathcal{R}_{\boldsymbol{y}}$, which depicts a severe loss in maneuverability. The loss is then quantified in more detail by r_1.

The redundancy measure is defined for all $t > 0$ but not for $t = 0$ because, for systems without feedthrough, the initial set $\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, 0)$ contains only one point. Therefore, its Lebesgue measure is $\mu_p(\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, 0)) = 0$ rendering the fraction (2.23) undefined. For systems with feedthrough, it is generally possible to have $\mu_p(\mathcal{R}_{\boldsymbol{y}}(\boldsymbol{x}_0,\, 0)) > 0$. In this case, the evaluation of the redundancy measure could be extended to $t = 0$.

The role of the projection operation of $\mathcal{R}_{\boldsymbol{y}}$ onto the lowest-dimensional sub-manifold $\mathcal{N} \subseteq \mathbb{R}^p$ containing $\mathcal{Q}_{\mathcal{Z}}$ is as follows. In view of Definition 13, for a set $\mathcal{A}$ that is not full-dimensional, i.e. $\dim(\mathcal{A}) < d$, the Lebesgue measure fulfills $\mu_d(\mathcal{A}) = 0$. The natural choice for comparing the volumes of $\mathcal{Q}_{\mathcal{Z}}$ and $\mathcal{R}_{\boldsymbol{y}}$ in Eq. (2.23) is $d = \max\{\dim(\mathcal{Q}_{\mathcal{Z}}),\, \dim(\mathcal{R}_{\boldsymbol{y}})\}$. By definition, we have $\mathcal{Q}_{\mathcal{Z}} \subseteq \mathcal{R}_{\boldsymbol{y}}$. Hence, $\dim(\mathcal{Q}_{\mathcal{Z}}) \leq \dim(\mathcal{R}_{\boldsymbol{y}})$ and $d = \dim(\mathcal{R}_{\boldsymbol{y}})$ follows. If we have strong inequality and do not perform the projection of $\mathcal{R}_{\boldsymbol{y}}$, then the redundancy measure r_1 would take the value zero, as $\mu_d(\mathcal{Q}_{\mathcal{Z}}) = 0$. This is mathematically meaningful because, for example, a plane in a three-dimensional space has volume zero since it covers zero percent of the three-dimensional space. However, from an engineering perspective, we want to have a deeper insight into the system. Therefore we choose to compare $\mathcal{Q}_{\mathcal{Z}}$ with the projection of $\mathcal{R}_{\boldsymbol{y}}$ on the enclosing manifold $\mathcal{N}$ of $\mathcal{Q}_{\mathcal{Z}}$. In this way, we only obtain $r_1 = 0$ if $\mathcal{Q}_{\mathcal{Z}}$ consists of a single point, or is empty. The scalar $r_2 \in \mathcal{I}_{0,\infty}$ indicates the case when the projection becomes active.

2.2.5 Example

Let us use an example to illustrate the results that we have obtained in this section. We use a three-tank system for this purpose. This system has three states while its actuator configuration can be adjusted easily for the purpose of redundancy analysis. It embodies a system that is not trivial for our analyses, but is still complex enough to show some key points that are worth discussing.

The dynamic model of the fluid system that is depicted in Fig. 2.4 is mainly based on modeling the flow between the connected reservoirs depending on the corresponding fluid levels. To this end, we can make use of Torricelli's law [141, p. 75]. This yields a nonlinear state space model of the form

$$\dot{\tilde{x}}_1 = \frac{1}{A_1}\left(\tilde{u}_1 + \mathrm{sign}(\tilde{x}_2 - \tilde{x}_1)\, q_1 \sqrt{2g|\tilde{x}_2 - \tilde{x}_1|}\right) \tag{2.24a}$$

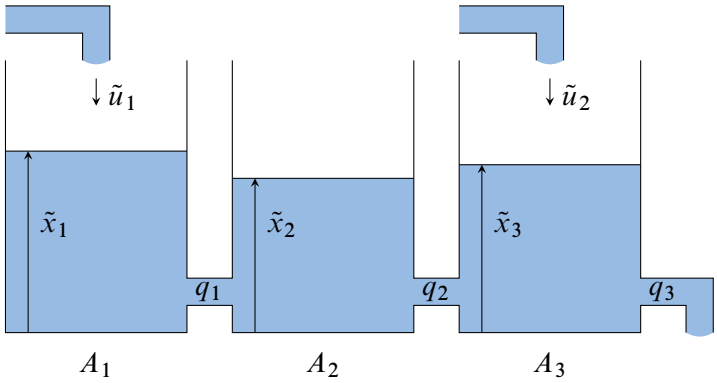


Figure 2.4: Three-tank system.

$$\dot{\tilde{x}}_2 = \frac{1}{A_2}\left(\operatorname{sign}(\tilde{x}_1 - \tilde{x}_2)\, q_1 \sqrt{2g|\tilde{x}_2 - \tilde{x}_1|} + \operatorname{sign}(\tilde{x}_3 - \tilde{x}_2)\, q_2 \sqrt{2g|\tilde{x}_3 - \tilde{x}_2|}\right) \tag{2.24b}$$

$$\dot{\tilde{x}}_3 = \frac{1}{A_3}\left(\tilde{u}_2 + \operatorname{sign}(\tilde{x}_2 - \tilde{x}_3)\, q_2 \sqrt{2g|\tilde{x}_3 - \tilde{x}_2|} - q_3 \sqrt{2g\tilde{x}_3}\right), \tag{2.24c}$$

where $g = 9.81 \frac{\mathrm{m}}{\mathrm{s}^2}$ is the gravitational acceleration. We denote the state vector containing the fluid levels of each tank by $\tilde{\boldsymbol{x}} = [\tilde{x}_1 \quad \tilde{x}_2 \quad \tilde{x}_3]^\top$, while the input vector is comprised of the inflow rates as $\tilde{\boldsymbol{u}} = [\tilde{u}_1 \quad \tilde{u}_2]^\top$. The sign-function in Eq. (2.24) ensures that the flow directions between the tanks is incorporated correctly depending on the respective fluid levels.

For the constellation of fluid levels $\tilde{x}_1 > \tilde{x}_2 > \tilde{x}_3$, there exist equilibrium points which we denote by $\tilde{\boldsymbol{x}}_{\mathrm{OP}} = [\tilde{x}_{1\mathrm{OP}} \quad \tilde{x}_{2\mathrm{OP}} \quad \tilde{x}_{3\mathrm{OP}}]^\top$, $\tilde{\boldsymbol{u}}_{\mathrm{OP}} = [\tilde{u}_{1\mathrm{OP}} \quad \tilde{u}_{2\mathrm{OP}}]^\top$. Linearizing around them, we obtain a linear time-invariant system representation $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$ with the system matrices

$$\mathbf{A} = \begin{bmatrix} -\frac{q_1}{A_1} t_{12} & \frac{q_1}{A_1} t_{12} & 0 \\ \frac{q_1}{A_2} t_{12} & -\frac{1}{A_2}\left(q_1 t_{12} + q_2 t_{23}\right) & \frac{q_2}{A_2} t_{23} \\ 0 & \frac{q_2}{A_3} t_{23} & -\frac{1}{A_3}\left(q_2 t_{23} + q_3 t_3\right) \end{bmatrix} \quad \text{and} \quad \mathbf{B} = \begin{bmatrix} \frac{1}{A_1} & 0 \\ 0 & 0 \\ 0 & \frac{1}{A_3} \end{bmatrix}. \tag{2.25}$$

Here, we have made use of the abbreviations

$$t_{ij} = \sqrt{\frac{g}{2(\tilde{x}_{i\mathrm{OP}} - \tilde{x}_{j\mathrm{OP}})}}, \qquad \tilde{x}_{i\mathrm{OP}} > \tilde{x}_{j\mathrm{OP}}$$

$$t_i = \sqrt{\frac{g}{2\tilde{x}_{i\mathrm{OP}}}}$$

that contain information about the considered operating point as well as the shifted coordinates $\boldsymbol{x} = \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_{\mathrm{OP}}$ and $\boldsymbol{u} = \tilde{\boldsymbol{u}} - \tilde{\boldsymbol{u}}_{\mathrm{OP}}$, respectively.

The redundancy analysis is performed with the parameter set $q_1 = q_2 = q_3 = 1\mathrm{cm}^2$, $A_1 = A_2 = A_3 = 100\mathrm{cm}^2$ at the operating point $\tilde{\boldsymbol{x}} = [39\mathrm{cm} \quad 33\mathrm{cm} \quad 27\mathrm{cm}]^\top$. Using a sampling time of $T_\mathrm{s} = 3\mathrm{s}$ results in the matrices

$$\mathbf{A}_\mathrm{d} = \begin{bmatrix} 0.79 & 0.19 & 0.02 \\ 0.19 & 0.63 & 0.17 \\ 0.02 & 0.17 & 0.69 \end{bmatrix} \quad \text{and} \quad \mathbf{B}_\mathrm{d} = \begin{bmatrix} 0.03 & 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.03 \end{bmatrix}$$

of the discretized system of the form (2.17a).

In order to start the redundancy analysis, we need to choose input and state constraints $\mathcal{X}$ and $\mathcal{U}$ as well as the outputs of interest. A set of state constraints results directly from linearization: The linear system description is only mathematically valid for $\tilde{x}_1 > \tilde{x}_2 > \tilde{x}_3 \geq 0$, leading to the state constraints for the linear plant

$$\mathcal{X} = \left\{ \boldsymbol{x} \in \mathbb{R}^3 \,\middle|\, \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix} \boldsymbol{x} \leq \begin{bmatrix} x_{1\mathrm{OP}} - x_{2\mathrm{OP}} - \varepsilon \\ x_{2\mathrm{OP}} - x_{3\mathrm{OP}} - \varepsilon \\ x_{3\mathrm{OP}} \end{bmatrix} \right\}$$

with some $\varepsilon > 0$, which we choose as $\varepsilon = 0.01$ for this example. Furthermore, we define the input bounds as $u_i \in \left[-30\frac{\mathrm{cm}^3}{s};\ 30\frac{\mathrm{cm}^3}{s}\right]$, $i \in \mathcal{I}_{1,2}$, i.e.

$$\mathcal{U} = \left\{ \boldsymbol{u} \in \mathbb{R}^2 \,\middle|\, \begin{bmatrix} \mathbf{I}_2 \\ -\mathbf{I}_2 \end{bmatrix} \boldsymbol{u} \leq 30 \cdot \mathbf{1}_4 \right\}$$

with $\mathbf{1}_a = [1 \quad \cdots \quad 1]^\top \in \mathbb{R}^a$, $a \in \mathbb{N}$. The outputs for the redundancy analysis are selected as

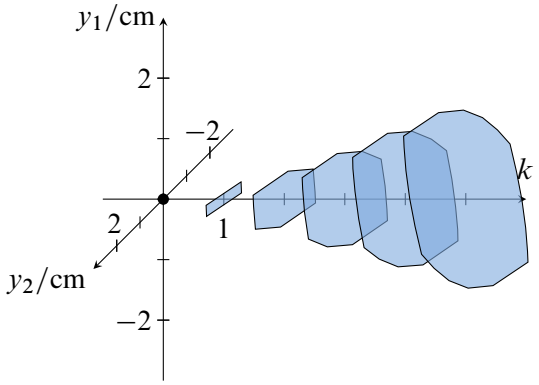
$$\boldsymbol{y} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{x} = \mathbf{C}_\mathrm{d} \boldsymbol{x}.$$

Note that the choice of the outputs only serves the purpose of selecting the quantities that are of special interest for the redundancy analysis. They can be chosen freely which means especially that they do not need to represent the reference outputs of the system. By the choice of the above $\mathbf{C}_\mathrm{d}$, we put a focus on system redundancy in the quantities x_2 and x_3.

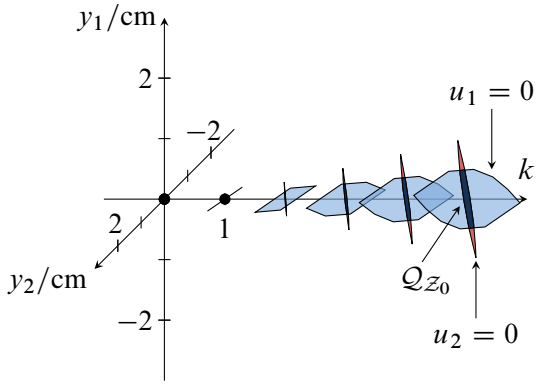
2.2.5.1 Redundancy w.r.t. the Fault Model $u_i = 0$

First, we investigate the set of redundantly reachable outputs in view of the fault model $u_i = 0$, $i \in \mathcal{M} = \mathcal{I}_{1,2}$. That is, the set of fault scenarios is given by $\mathcal{Z}_0$ defined in Eq. (2.13). In line with Theorem 4, we can compute a finite number of reachable sets for obtaining the set sequence of redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}_0,k}$. Results for $\mathcal{Q}_\mathcal{Z}$ are shown in Fig. 2.5. On the left, in Fig. 2.5a, the evolution of the output reachable set $\mathcal{R}_\boldsymbol{y}$ is shown for the nominal plant and an initial state $\boldsymbol{x} = \mathbf{0}_n$. Fig. 2.5b illustrates the corresponding output reachable sets $\mathcal{R}_\boldsymbol{y}^z$, $z \in \mathcal{Z}_0$, which are clearly much smaller in size, compared to the nominal case $\mathcal{R}_\boldsymbol{y}$. As a first observation, we can state that this plant with the given constraint configuration does not possess redundant actuators w.r.t. Definition 5. This can simply be deduced from the strict inclusion $\mathcal{R}_\boldsymbol{y}^z \subset \mathcal{R}_\boldsymbol{y}$ for all $z \in \mathcal{Z}_0$. Further comparing the sets $\mathcal{R}_\boldsymbol{y}^z$ itself, we can see that the sets highlighted in red are much more compressed compared to the ones indicated in light blue. The red sets stem from the fault case $u_2 = 0$. Hence, the actuator u_2 is much more important for the maneuverability in the output space than u_1 as the set of reachable points is strongly diminished due to its failure.

The set $\mathcal{Q}_{\mathcal{Z}_0}$ of outputs reachable independently of the fault scenario is the intersection of the subsets $\mathcal{R}_\boldsymbol{y}^z \subset \mathcal{R}_\boldsymbol{y}$. It is highlighted in dark blue color in Fig. 2.5b. Interestingly, for the time step



(a) Evolution of the output reachable set $\mathcal{R}_{\boldsymbol{y}}$ of the nominal system.



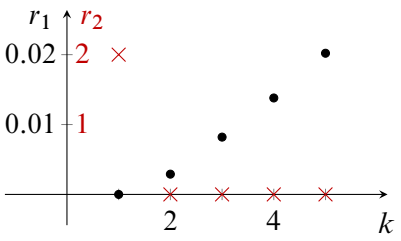
(b) Evolution of the output reachable sets $\mathcal{R}_{\boldsymbol{y}}^{z}$ and $\mathcal{Q}_{\mathcal{Z}_0}$ for the fault scenarios $u_i = 0$, $i \in \mathcal{M} = \mathcal{I}_{1,2}$.

Figure 2.5: Output reachable sets for the three-tank system for the fault scenario $u_i = 0$.

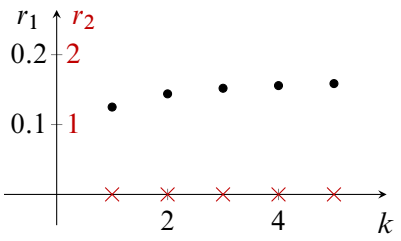
$k = 1$, the intersection $\mathcal{Q}_{\mathcal{Z}_0}$ only contains a single point, whereas, for all steps $k > 1$, $\mathcal{Q}_{\mathcal{Z}_0}$ contains infinitely many points. By the loss of an actuator u_i, the ability to choose an input direction via the choice of $\boldsymbol{u}$ vanishes, due to $\mathcal{U}_z$ embodying a line in $\mathbb{R}^2$ in both cases. As a consequence, the orientation of the output reachable sets $\mathcal{R}_{\boldsymbol{y}}^{z}$ at $k = 1$ only depends on the corresponding input direction $\boldsymbol{b}_{\mathrm{d}i}$, where $\mathbf{B}_{\mathrm{d}} = \begin{bmatrix} \boldsymbol{b}_{\mathrm{d}1} & \boldsymbol{b}_{\mathrm{d}2} \end{bmatrix}$. As these input directions do not coincide, the output reachable sets are oriented differently and their intersection only contains one point. This causes that, for $k = 1$, we can only guarantee $\boldsymbol{y} = \mathbf{0}_p$ to be reachable independently of the occurring fault scenario. No other point in the output space can be reached redundantly at this time.

The graphical redundancy analysis presented above can give a first insight into the redundancy properties of the chosen system setup. Depending on this setup, it may, however, be challenging to perform the analysis. The graphical results are hard to interpret if the dimensions of the system configuration grow. Then, illustrating them inherently demands for projecting them onto some two-dimensional (maximally three-dimensional) subspace of the state space. Although generally feasible, this may cause misleading interpretations of the results. To overcome this issue, we have proposed a measure enabling for a simplified interpretation of the results independently of any system dimensions. For the setup above, the redundancy measure introduced in Section 2.2.4 is shown in Fig. 2.6a.

In the graph, we can see that, at step $k = 1$, there is a loss in dimension of $\mathcal{Q}_{\mathcal{Z}_0}$ w.r.t. $\mathcal{R}_{\boldsymbol{y}}$, indicated by $r_2 = 2$. This is caused by the already mentioned fact that $\mathcal{Q}_{\mathcal{Z}_0,1}$ only contains one point whereas $\mathcal{R}_{\boldsymbol{y}1}$ is a two-dimensional set. The evolution of r_1 shows that the set of redundantly reachable outputs grows over time compared to $\mathcal{R}_{\boldsymbol{y}}$, indicated by increasing values of r_1. However, this happens at a low level, implying that a lot of guaranteed maneuverability is lost due to the fault scenarios.

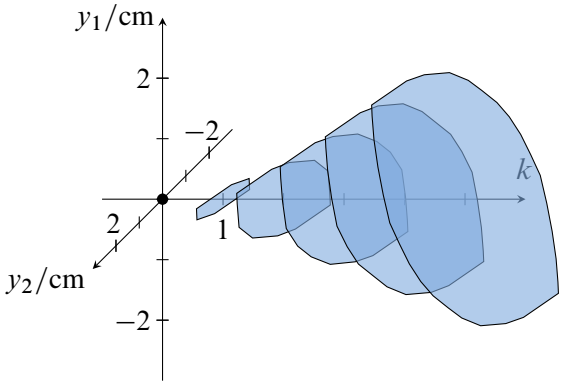


(a) Redundancy measures r_1 (•) and r_2 (×) for the fault scenarios $u_i = 0$, $i \in \mathcal{M} = \mathcal{I}_{1,2}$.

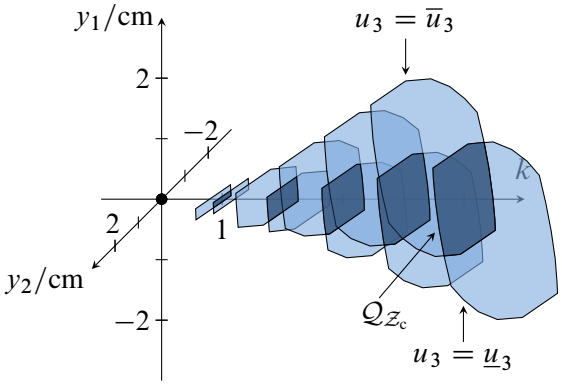


(b) Redundancy measures r_1 (•) and r_2 (×) for the fault scenarios $u_i = \text{const}$, $i \in \mathcal{M} = \{3\}$.

Figure 2.6: Redundancy measures for the three-tank system.



(a) Evolution of the output reachable set $\mathcal{R}_y$ of the nominal system.



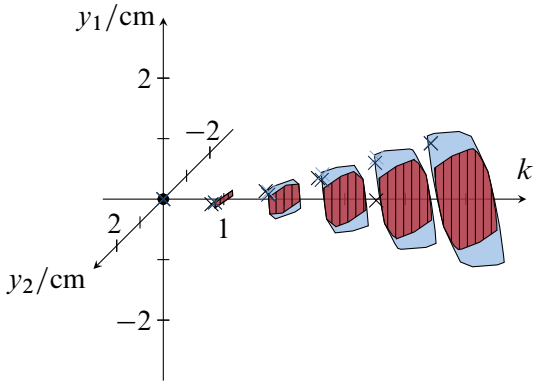
(b) Evolution of the output reachable sets $\mathcal{R}_y^z$ and $\mathcal{Q}_{\mathcal{Z}_c}$ for the fault scenarios $u_i = \text{const}$, $i \in \mathcal{M} = \{3\}$.

Figure 2.7: Output reachable sets for the three-tank system for the fault scenario $u_i = \text{const}$.

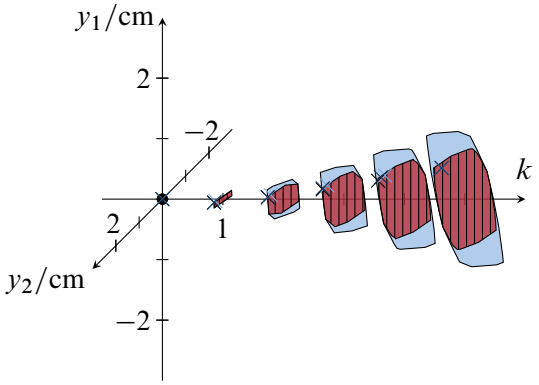
2.2.5.2 Redundancy w.r.t. the Fault Model $u_i = \text{const}$

As a second example, we examine the case of stuck actuators. For the system under consideration, such faults could result from defective valves. These scenarios are collected in the infinite set $\mathcal{Z}_c$. Making use of Theorem 5, the set of redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}_c}$ can be computed by using the finite set $\overline{\mathcal{Z}}_c^0$ instead. Hence, only the scenarios, where the i-th actuator is stuck at its maximal or minimal value, are explicitly computed. However, it turns out that, using the model from the prior section and $\mathcal{M} = \mathcal{I}_{1,2}$, there are no output points which can redundantly be reached because $\mathcal{Q}_{\mathcal{Z}_c,k} = \emptyset$ for all $k > 0$. This means we do not have any redundancy properties w.r.t. this class of faults.

We introduce one more input to overcome this situation. It has the same input bounds like the existing actuators, but has a shared influence on the first and the third actuator. We choose $\boldsymbol{b}_{d3} = \frac{1}{2}(\boldsymbol{b}_{d1} + \boldsymbol{b}_{d2})$ and use the augmented input matrix $\widetilde{\mathbf{B}}_d = \begin{bmatrix}\mathbf{B}_d & \boldsymbol{b}_{d3}\end{bmatrix}$ for redundancy analysis. Choosing $\mathcal{M} = \{3\}$, we obtain the results depicted in Fig. 2.7. In Fig. 2.7a, we can observe that the capabilities of the nominal plant are now extended w.r.t. the original plant (Fig. 2.5a), which is clear in view of the additional input power provided by the third input. We can also see



(a) Evolution of the redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}_0}$ (blue) and $\hat{\mathcal{Q}}_{\mathcal{Z}_0}$ (red, shaded) for the fault scenarios $u_i = 0$, $i \in \mathcal{M} = \mathcal{I}_{1,3}$. Output trajectories for the three scenarios and the constraints $\boldsymbol{y}_{z,k} \in \mathcal{Q}_{\mathcal{Z}_0,k}\ \forall\ k \in \mathcal{I}_{0,N}, \forall\ z \in \mathcal{Z}_0$ and the reference point $\boldsymbol{y}_{\text{ref}} = [1.5 \quad 1.5]^\top$.



(b) Evolution of the redundantly reachable outputs $\mathcal{Q}_{\mathcal{Z}_0}$ (blue) and $\hat{\mathcal{Q}}_{\mathcal{Z}_0}$ (red, shaded) for the fault scenarios $u_i = 0$, $i \in \mathcal{M} = \mathcal{I}_{1,3}$. Output trajectories for the three scenarios and the constraints $\boldsymbol{y}_{z,k} \in \hat{\mathcal{Q}}_{\mathcal{Z}_0,k}\ \forall\ k \in \mathcal{I}_{0,N}, \forall\ z \in \mathcal{Z}_0$ and the reference point $\hat{\boldsymbol{y}}_{\text{ref}} = [0.9 \quad 1]^\top$.

Figure 2.8: Comparison of $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$. For the output trajectories, the different scenarios are depicted in black ($u_1 = 0$), light blue ($u_2 = 0$), dark blue ($u_3 = 0$).

that, assuming no faults in the actuators u_1 and u_2, there remain points in the output space that can guaranteed to be reachable even if actuator u_3 gets stuck at an arbitrary position within its input bounds.

The redundancy measure for this analysis is shown in Fig. 2.6b. There are no losses in dimensionality as in the first show case, indicated by $r_2 = 0$ for all $k > 0$. The output reachable set is always two-, i.e. full-dimensional. However, the area that can be reached independently of the fault case is still significantly reduced compared to the nominal case. The redundancy measure r_1 at step $k = 5$ indicates a loss of reachable area of approximately 85%.

2.2.5.3 Comparison of $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$

In this part, we want to examine the difference between Definitions 9 and 10, i.e. $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$. For best visualizing them, we choose the setup with three actuators, i.e., the input matrix is again $\widetilde{\mathbf{B}}_{\text{d}}$. However, we return to the fault scenario $u_i = 0$ and include all actuators into the analysis. Hence, $\mathcal{M} = \mathcal{I}_{1,3}$. The results are depicted in Fig. 2.8, where both types of sets are drawn.

We can see that the relation $\hat{\mathcal{Q}}_{\mathcal{Z}_0,k} \subseteq \mathcal{Q}_{\mathcal{Z}_0,k}$ holds for all depicted time steps k. This is not surprising, as we posed stronger requirements for points $\boldsymbol{y}$ to be a member of $\hat{\mathcal{Q}}_{\mathcal{Z}_0,k}$ compared to $\mathcal{Q}_{\mathcal{Z}_0,k}$. Let us briefly discuss the difference between the two sets for the time steps $k = 2$ and $k = 3$. Starting from the initial point $\boldsymbol{x}_0 = \mathbf{0}_n$, it is possible to reach every point in the union of the blue and the shaded red area, i.e. $\mathcal{Q}_{\mathcal{Z}_0,k}$, at $k = 2$, independently of the specific fault. Hence, also the points in the shaded red area, $\hat{\mathcal{Q}}_{\mathcal{Z}_0,k}$, are attainable at $k = 2$. Furthermore, by construction of the set $\hat{\mathcal{Q}}_{\mathcal{Z}_0}$, we are able to find an input for every fault scenario driving each output to the shaded

red area at $k = 3$, given that the output is located in the shaded red area at $k = 2$. This is not guaranteed for the points in the blue area. Hence, given that the common system output is located in the blue area at $k = 2$, there is no guarantee that there exists an input for each fault scenario driving this output to $\mathcal{Q}_{\mathcal{Z}_0,k}$ at $k = 3$.

We use an optimal control scheme with quadratic cost function and the prediction horizon $N = 5$ to validate the results. In case of $\hat{\mathcal{Q}}_{\mathcal{Z}_0}$, we expect that it is feasible to find a trajectory satisfying the time-varying constraints $\boldsymbol{y}_{z,k} \in \hat{\mathcal{Q}}_{\mathcal{Z}_0,k}$ for all $k \in \mathcal{I}_{0,N}$ and all $z \in \mathcal{Z}_0$ while driving all systems from $\boldsymbol{x}_0$ to $\hat{\boldsymbol{y}}_{\text{ref}} = [0.9 \quad 1]^\top \in \hat{\mathcal{Q}}_{\mathcal{Z}_0,N}$. This reference point is chosen randomly from $\hat{\mathcal{Q}}_{\mathcal{Z}_0,N}$. In turn, we expect for at least one $z \in \mathcal{Z}_0$ that there is no feasible solution for driving all systems to $\boldsymbol{y}_{\text{ref}} = [1.5 \quad 1.5]^\top \in \mathcal{Q}_{\mathcal{Z}_0,N} \setminus \hat{\mathcal{Q}}_{\mathcal{Z}_0,N}$, while satisfying $\boldsymbol{y}_{z,k} \in \mathcal{Q}_{\mathcal{Z}_0,k}$ for all $k \in \mathcal{I}_{0,N}$. Again, the point $\boldsymbol{y}_{\text{ref}}$ is chosen randomly such that it is an element of $\mathcal{Q}_{\mathcal{Z}_0,N}$, but not of $\hat{\mathcal{Q}}_{\mathcal{Z}_0,N}$, i.e. the blue area in Fig. 2.8a. In both settings, we enforce reaching the reference point by the terminal constraints $\boldsymbol{y}_{z,N} = \hat{\boldsymbol{y}}_{\text{ref}}$ and $\boldsymbol{y}_{z,N} = \boldsymbol{y}_{\text{ref}}$, respectively.

The expressed expectations are validated as can be seen in Fig. 2.8. For all scenarios, an output trajectory can be found staying in the shaded red area while driving to $\hat{\boldsymbol{y}}_{\text{ref}}$, see Fig. 2.8b. This is not the case for the blue area and the reference point $\boldsymbol{y}_{\text{ref}}$, as shown in Fig. 2.8a. However, the reference points are reachable at $k = N$ in any case. To enable the computation of the results shown in Fig. 2.8a, we have made use of slack variables in the time-varying constraints. For more information on the optimal control scheme used to produce the results, see Section A.1.

2.3 Redundant Stabilizability

We have put a lot of effort into developing a meaningful definition of redundancy and transferring its idea into two algorithms to analyze a dynamic system w.r.t. its redundancy properties. To this end, we have made use of reachability techniques and derived the set $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$ of redundantly reachable points in the output space. From a control perspective, we are not only interested in reachability. A natural requirement of a controlled system is to be stabilizable at some predefined reference point. In our setting, this means that we are interested in a statement about whether it is possible to reach some reference point $\boldsymbol{y}_{\text{ref}} \in \mathbb{R}^p$ and hold the system there independently of any of the considered fault scenarios in $\mathcal{Z}$.

Knowing the set evolution of $\hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$, we know that there exists an input trajectory keeping the system output within $\hat{\mathcal{Q}}_{\mathcal{Z}}$ for all fault scenarios. However, this does not suffice to guarantee that there exists an input trajectory being capable of keeping the output at a reference point $\boldsymbol{y}_{\text{ref}} \in \hat{\mathcal{Q}}_{\mathcal{Z}}$ for all times. The formulation of conditions ensuring this is subject of this section.

2.3.1 Conditions for Redundant Stabilizability

Intuitively, for every fault scenario $z \in \mathcal{Z}$, the reference point $\boldsymbol{y}_{\text{ref}}$ must be reachable from $\boldsymbol{x}_0$, i.e. there must exist some time point $t_{\text{r}}^z \in [0; \infty)$ such that $\boldsymbol{y}_{\text{ref}} \in \mathcal{R}_{\boldsymbol{y}}^z(\boldsymbol{x}_0, t_{\text{r}})$. Associated with $\boldsymbol{y}_{\text{ref}}$ is

a set $\mathcal{P}^z$ of reachable states $\boldsymbol{x}_z \in \mathcal{R}^z(\boldsymbol{x}_0,\, t_\mathrm{r}^z)$ that are mapped to $\boldsymbol{y}_\mathrm{ref}$ via the output equation. If the state $\boldsymbol{x}_z$ can be kept within this preimage set

$$\mathcal{P}^z(\mathcal{R}^z(\boldsymbol{x}_0,\, t_\mathrm{r}^z),\, \boldsymbol{y}_\mathrm{ref}) = \left\{\boldsymbol{x}_z \in \mathcal{R}^z(\boldsymbol{x}_0,\, t_\mathrm{r}^z) \,|\, \exists\, \boldsymbol{u}_z(t_\mathrm{r}^z) \in \mathcal{U}_z(t_\mathrm{r}^z) \text{ s.t. } \boldsymbol{y}_\mathrm{ref} = \boldsymbol{g}_z(\boldsymbol{x}_z,\, \boldsymbol{u}_z(t_\mathrm{r}^z))\right\}$$

of $\boldsymbol{y}_\mathrm{ref}$ by some control function $\boldsymbol{u}_z$, the output remains at $\boldsymbol{y}_\mathrm{ref}$. Note that, having a reachable equilibrium point in the state space that is mapped to $\boldsymbol{y}_\mathrm{ref}$ is not necessary in order to keep the system at $\boldsymbol{y}_\mathrm{ref}$ because we want to allow for any system movement which is not visible at the output. This approach is aligned with computing worst-case capabilities of the system at hand.

The above arguments ask for the existence of some individual time points $t_\mathrm{r}^z \in [0;\, \infty)$ at which every system Σ_z can reach $\boldsymbol{y}_\mathrm{ref}$. Stabilizing each system at $\boldsymbol{y}_\mathrm{ref}$ implies that there must also be a common point t_r, for which $\boldsymbol{y}_\mathrm{ref}$ is reachable by all systems, i.e. $\boldsymbol{y}_\mathrm{ref} \in \mathcal{Q}_\mathcal{Z}(\boldsymbol{x}_0,\, t_\mathrm{r})$. Furthermore, demanding existence of an input signal $\boldsymbol{u}_z$, such that the system states $\boldsymbol{x}_z$ remains in $\mathcal{P}^z$ once $\boldsymbol{x}_z \in \mathcal{P}^z$ is equivalent to demanding for the existence of a *control invariant set* $\mathcal{P}_\mathrm{c}^z \subseteq \mathcal{P}^z$ for every $z \in \mathcal{Z}$.

Definition 16 ([136, Sec. 3.1.2])**.** *Given a system of the form* (2.1)*, let its solution at time t again be denoted by $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t)$. Then, a control invariant set $\mathcal{V}$ is defined as*

$$\mathcal{V} = \{\boldsymbol{x} \in \mathcal{X} \,|\, \exists\, \boldsymbol{u} \colon [0;\, t) \to \mathcal{U} \text{ s.t. } \boldsymbol{\Phi}(\boldsymbol{x},\, \boldsymbol{u},\, t) \in \mathcal{V}\; \forall\, t \geq 0\}\,.$$

Let us collect the above results in

Theorem 7. *Let $\mathcal{Z}$ denote a set of fault scenarios. A reference point $\boldsymbol{y}_\mathrm{ref} \in \mathbb{R}^p$ is redundantly asymptotically stabilizable from any initial state $\boldsymbol{x}_0 \in \mathcal{X}_z(0)\; \forall z \in \mathcal{Z}$ if and only if*

1. *there exists a time point $t_\mathrm{r} \in [0;\, \infty)$ such that $\boldsymbol{y}_\mathrm{ref} \in \mathcal{Q}_\mathcal{Z}(\boldsymbol{x}_0,\, t_\mathrm{r})$,*
2. *there exist control invariant subsets $\mathcal{P}_\mathrm{c}^z \subseteq \mathcal{P}^z(\mathcal{R}^z(\boldsymbol{x}_0,\, t_\mathrm{r}^z),\, \boldsymbol{y}_\mathrm{ref})$ for all $z \in \mathcal{Z}$.*

Proof. ($\Leftarrow$) Assume both conditions are satisfied. Then, for each fault scenario $z \in \mathcal{Z}$, there exists an input trajectory $\boldsymbol{u}_z$ driving the system to $\boldsymbol{y}_\mathrm{ref}$ at time t_r. Furthermore, the state can be driven to some $\boldsymbol{x}_z \in \mathcal{P}_\mathrm{c}^z$. Since all sets $\mathcal{P}_\mathrm{c}^z$ are control invariant for the respective fault dynamics, there exist input trajectories maintaining $\boldsymbol{y}_z = \boldsymbol{y}_\mathrm{ref}$.

($\Rightarrow$) Assume that Condition 1 is violated and Condition 2 holds. Then, $\boldsymbol{y}_\mathrm{ref}$ can be maintained in all cases after being reached. However, the absence of a common time point t_r with $\boldsymbol{y}_\mathrm{ref} \in \mathcal{Q}_\mathcal{Z}(\boldsymbol{x}_0,\, t_\mathrm{r})$ implies, that $\boldsymbol{y}_\mathrm{ref}$ cannot be reached at least asymptotically in at least one fault scenario.

Now, assume that Condition 1 is fulfilled but Condition 2 is violated. Then, $\boldsymbol{y}_\mathrm{ref}$ can be reached at least asymptotically in every scenario. However, there exists at least one fault scenario z, in which the there does not exist a control invariant set $\mathcal{P}_\mathrm{c}^z \subseteq \mathcal{P}^z(\mathcal{R}^z(\boldsymbol{x}_0,\, t_\mathrm{r}^z),\, \boldsymbol{y}_\mathrm{ref})$ that the state $\boldsymbol{x}_z$ can be driven to. This means that $\boldsymbol{y}_\mathrm{ref}$ cannot be maintained in this case. □

Remark 6. If the common time point t_r is at infinity, then the reference point is only asymptotically stabilizable. If it is finite, the point $\boldsymbol{y}_\mathrm{ref}$ is exactly reachable and stabilizable.

We can formulate Algorithm 3 that is used to check a point $\boldsymbol{y}_{\text{ref}}$ for redundant stabilizability. We will make use of computing the *maximal* control invariant set $\mathcal{V}^*$ which is defined for system (2.1) as the control invariant set within some superset $\overline{\mathcal{V}} \subseteq \mathcal{X}$ containing all other control invariant sets $\mathcal{V} \subseteq \overline{\mathcal{V}}$ [20]. Algorithms for computing maximal control invariant sets are, among others, explained by Borrelli et al. [20].

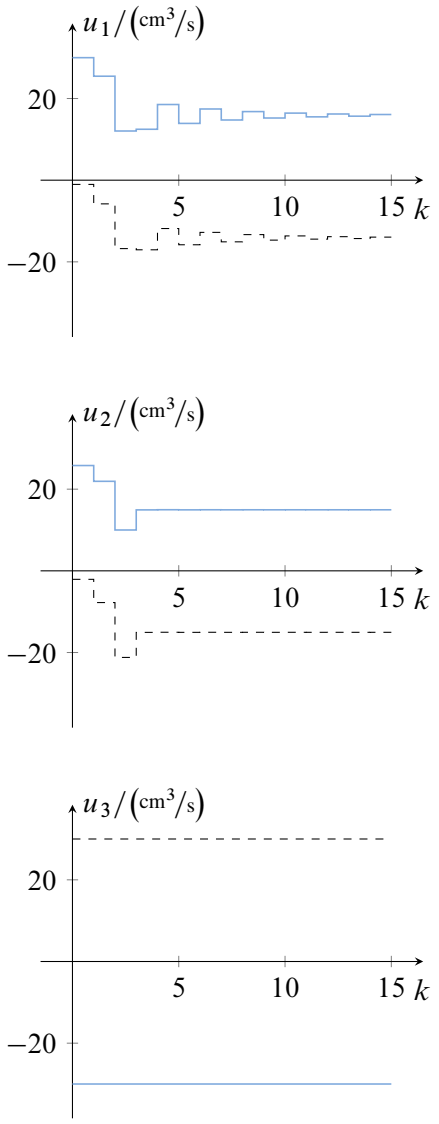
Algorithm 3 Check for redundant stabilizability of $\boldsymbol{y}_{\text{ref}}$.

Input: $\mathcal{Z}$, N, $\boldsymbol{y}_{\text{ref}}$, $\mathcal{Q}_{\mathcal{Z},k}$, $\mathcal{R}_k^z$ for all $k \in \mathcal{I}_{0,N}$
$k_{\text{s}} \leftarrow -1$
for all $k \in \mathcal{I}_{0,N}$ **do**
  if $\boldsymbol{y}_{\text{ref}} \in \mathcal{Q}_{\mathcal{Z},k}$ **then**
    $v \leftarrow 1$
    for all $z \in \mathcal{Z}$ **do**
      Compute $\mathcal{P}_k^z(\mathcal{R}_k^z,\ \boldsymbol{y}_{\text{ref}})$
      Compute $\mathcal{V}_z^* \subseteq \mathcal{P}_k^z(\mathcal{R}_k^z,\ \boldsymbol{y}_{\text{ref}})$
      if $\mathcal{V}_z^* = \emptyset$ **then** ▷ There exists no control invariant set within $\mathcal{P}_k^z$.
        $v \leftarrow 0$
      end if
    end for
    if $v = 1$ **then**
      $k_{\text{s}} \leftarrow k$
      return ▷ $\boldsymbol{y}_{\text{ref}}$ is redundantly stabilizable.
    end if
  end if
end for
Output: k_{s}

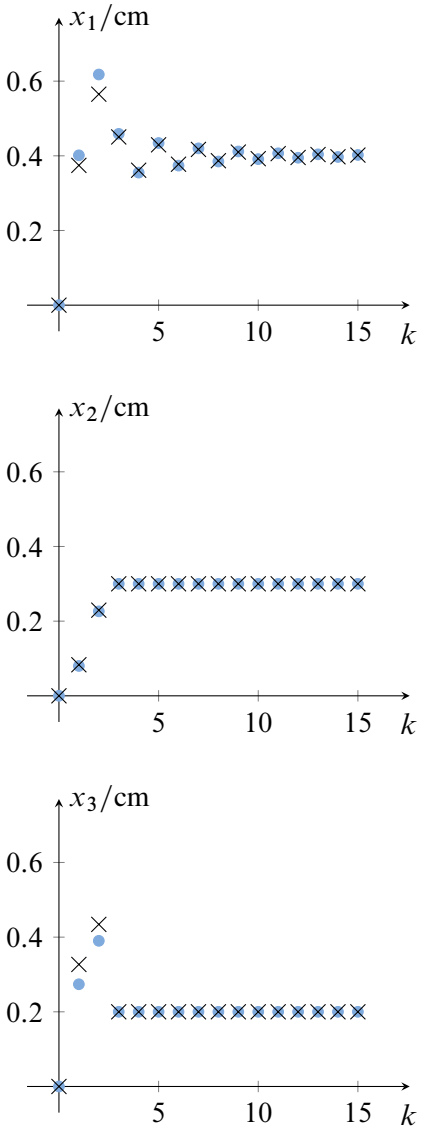
Algorithm 3 outputs the integer k_{s} that takes a non-negative value if a step $k \leq N$ has been identified for which the conditions of Theorem 7 are fulfilled. In every fault scenario, the reference point $\boldsymbol{y}_{\text{ref}}$ can be reached and maintained for all $k \geq k_{\text{s}}$. If no $k_{\text{s}} \geq 0$ is returned, i.e. if $k_{\text{s}} = -1$, the algorithm failed. This does, however, not imply that $\boldsymbol{y}_{\text{ref}}$ is not redundantly stabilizable. This is due to N being a finite number. This renders the output of the algorithm only sufficient, but not necessary.

2.3.2 Example

Let us continue the example from Section 2.2.5 and analyze the three-tank system for redundant stabilizability. We do this for the setup used in Section 2.2.5.2 and the set of fault scenarios defined by $\overline{\mathcal{Z}}_{\text{c}}^0$. Looking at the redundantly reachable sets in Fig. 2.7b, we can choose a point from $\mathcal{Q}_{\mathcal{Z}_{\text{c}}} = \mathcal{Q}_{\overline{\mathcal{Z}}_{\text{c}}^0}$ and test whether it is redundantly stabilizable using Algorithm 3. As an example, the point $\boldsymbol{y}_{\text{ref}} = \begin{bmatrix} 0.3 & 0.2 \end{bmatrix}^\top$ is chosen. It is an element of $\mathcal{Q}_{\overline{\mathcal{Z}}_{\text{c}}^0}$ from step $k = 3$ onward. Hence, we know that Condition 1 from Theorem 7 is satisfied. It remains to prove if Condition 2 is fulfilled



(a) Input evolution for both fault scenarios. The scenario $u_3 = \underline{u}_3$ is depicted in blue (—), the scenario $u_3 = \overline{u}_3$ in black (- - -).



(b) State evolution for both fault scenarios. The scenario $u_3 = \underline{u}_3$ is depicted in blue (•), the scenario $u_3 = \overline{u}_3$ in black (×).

Figure 2.9: State and input signals for driving the system to $\boldsymbol{y}_{\mathrm{ref}}$ in each fault scenario.

as well. I.e., we need to check whether there exist control invariant sets $\mathcal{V}_z^*$ in the preimage sets $\mathcal{P}^z$ of $\boldsymbol{y}_{\mathrm{ref}}$ for each $z \in \overline{\mathcal{Z}}_{\mathrm{c}}^0$.

For each of the fault scenarios in $\overline{\mathcal{Z}}_{\mathrm{c}}^0$, Algorithm 3 returns nonempty control invariant sets contained in the respective preimage sets from time step $k = 3$ onward, i.e. $k_{\mathrm{s}} = 3$ in the algorithm. As $\boldsymbol{y}_{\mathrm{ref}} \in \mathbb{R}^2$, there is only one degree of freedom in states $\boldsymbol{x}_z$ that are mapped to $\boldsymbol{y}_{\mathrm{ref}}$. Hence, both control invariant sets $\mathcal{V}_z^*$ are line segments in $\mathbb{R}^3$. Again using an optimal control scheme with quadratic cost function and $\mathcal{V}_z^*$ as terminal region, it is possible to compute admissible input trajectories that drive the system to $\boldsymbol{y}_{\mathrm{ref}}$ and stabilize it there. The results are depicted in Fig. 2.9. We can see that, in both scenarios, we can reach $\boldsymbol{y}_{\mathrm{ref}}$ within three time steps and hold the output

there. Driving the system to an equilibrium point in state space is also achieved in both cases, but this process takes longer. For more information on the underlying optimal control scheme and the control invariant sets $\mathcal{V}_z^*$, see Section A.1.

Remark 7. Note that we have restricted the statement on redundant stabilizability to the set of fault scenarios $\overline{\mathcal{Z}}_{\mathrm{c}}^{0}$ as opposed to $\mathcal{Z}_{\mathrm{c}}$. This is done because it is involved, if not infeasible, to prove the existence of admissible control invariant sets for the infinite set of fault scenarios $\mathcal{Z}_{\mathrm{c}}$.

2.4 Use of Redundancy Information in System and Control Design

Having discussed the computation of redundantly reachable sets and redundant stabilizability of a reference point $\boldsymbol{y}_{\mathrm{ref}}$, let us shortly outline two applications, where the possession of such information can be useful in system or control design.

Choice and Design of Actuators Optimal placement of actuators in a dynamic system such that properties like controllability are optimized has been of vast interest for many decades. To the best of the author's knowledge, directly evaluating the impact of actuator placement on redundancy properties has not been investigated. The procedure introduced in this chapter can help in an early stage of development to predict the corresponding system behavior. In real applications, it is not only important to have knowledge about the structural redundancy induced by the placement of actuators. For linear systems, this can be verified using criteria as given in Eqs. (2.3), (2.4), (2.5), or Theorem 1. However, in view of limited actuation ranges due to the input bounds $\mathcal{U}$, structural knowledge only creates a coarse and potentially misleading picture of the system's capabilities in various fault scenarios. This picture can be drawn more precisely and more explicitly with help of the developed algorithms.

Trajectory Planning In many technical applications, certain system states have a larger importance for the output quality than others. An example could be given by a chemical plant, where we are flexible in a certain process temperature as long as the concentration of some product is always kept within predefined limits. In this scenario, it is advisable to specify the product concentration as output variable of the dynamic system and compute its redundancy properties applying the introduced algorithms. Answers can be given, whether a desired concentration can be reached from a certain initial nominal state of the system. Additionally, we have knowledge about its redundant stabilizability.

Based on this knowledge, it is possible to choose a trajectory for the process quantities that are of high importance such that they always lie in a band with enhanced safety features. That is, they can be kept in the band even under an unexpected occurrence of a fault state. This can be done with the help of the sequence $\hat{\mathcal{Q}}_{\mathcal{Z},k}$ computed in Algorithm 2. It returns points for which there exists at least one input sequence keeping the system's output within $\hat{\mathcal{Q}}_{\mathcal{Z},k}$ for all considered times $k \in \mathcal{I}_{0,N}$.

2.5 Summary

In this chapter, we establish a framework to methodologically analyze systems for their redundancy properties. Besides identifying explicit redundant system parts such as double-placed system components, we especially enable finding redundancy structures that are hidden in the system dynamics. With *redundancy*, we associate the system's persistent capability of executing a task in case of a subsystem's failure. This view is established by [24].

Various existing definitions of redundancy for unconstrained linear time-invariant systems are discussed. They are based on geometric properties such as matrix rank or null spaces, which result in conditions that are hard to transfer to a nonlinear setting. A definition for input redundancy based on the absence of left-invertibility is due to Kreiss and Trégouët [89]. Following a discussion about entities that are meaningful to be called *redundant*, we use it as baseline for introducing a new definition for actuator redundancy which is designed for extending to linear systems with state and input constraints as well as purely nonlinear systems. In contrast to the present definitions, our new definition matches the intuitions that are established by [24].

In Theorem 2, we prove equivalence of Definition 3 that is proposed in [89] and Definition 5 which is introduced in this thesis for the case of unconstrained linear time-invariant systems. This means that at least one redundant actuator exists if and only if the system is input redundant. It is further due to Theorem 3 that this equivalence is only partly maintained in view of constrained linear or nonlinear systems. An obvious extension to a set of arbitrary fault scenarios $\mathcal{Z}$ in contrast to only incorporating actuator faults is presented in Definition 6.

In the sequel, a novel practical approach for analyzing redundancy of dynamic systems w.r.t. predefined fault scenarios is introduced and discussed in detail. We use reachability analysis for this and introduce two different types of redundantly reachable sets in the output space, $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$, see Definitions 9 and 10. The set $\hat{\mathcal{Q}}_{\mathcal{Z}}$ has enhanced invariance properties. It collects points for which there exists at least one input trajectory keeping the system within $\hat{\mathcal{Q}}_{\mathcal{Z}}$ over time and for every fault scenario contained in $\mathcal{Z}$. The other one, i.e. $\mathcal{Q}_{\mathcal{Z}}(\boldsymbol{x}_0, t)$, collects all points in the output space that are commonly reachable in all fault scenarios at a specified time point. No invariance properties are included here. This set is the basis for analyzing redundant stabilizability, as discussed later in the chapter.

For the sets $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$, we prove important properties in Theorems 4 and 5 for two important classes of faults, i.e. $u_i = 0$ and $u_i = \text{const}$, which depict powerless actuators and actuators that are stuck at some unknown values. Doing this, we establish conditions on the system class that enable a computation of these sets in finite time without introducing approximation errors, even if considering an infinite amount of faults, as done by considering the scenario $u_i = \text{const}$. It is proven in Section 2.2.2.1 that linear constrained systems fulfill these conditions. Explicit computation of $\mathcal{Q}_{\mathcal{Z}}$ and $\hat{\mathcal{Q}}_{\mathcal{Z}}$ is shown in the Algorithms 1 and 2.

To simplify redundancy analysis, we further introduce a measure in Definition 15 providing a quick first impression on the system's redundancy properties w.r.t. a set $\mathcal{Z}$ of fault scenarios. It is based on Lebesgue measures and relates the volumes of the nominal output reachable sets of

the plant to the volume of the redundantly reachable set. Thereby, we condense the information obtained through the study of possibly high-dimensional reachable sets to a scalar that can be easily displayed and interpreted.

After establishing redundant reachability, we discuss redundant stabilizability and set up corresponding necessary and sufficient conditions in Theorem 7. A reference point $\boldsymbol{y}_{\text{ref}}$ in the output space must be redundantly reachable for some time point $t \geq 0$. Furthermore, the reachable preimage sets of $\boldsymbol{y}_{\text{ref}}$ in the state space must contain control invariant sets. The conditions ensure, that a reference point can be reached under all considered circumstances and can be kept there. In line with the aim of identifying worst-case properties of the systems at hand, redundant stabilizability is not formulated in state space, but in output space. This leaves the system freedom to move in state space while stabilizing only the outputs. Algorithm 3 is presented for verifying the mentioned conditions.

The redundancy analysis developed in this chapter can be used in system and control design. It can be used to assess the necessity of further actuators for an existing plant or can justify the omission of actuators due to their superfluity for redundancy. In control design, the analysis can be used to choose trajectories and reference points that are redundantly reachable or stabilizable, i.e. independently of any unforeseen fault event.

3 Structured Control Design

Having analyzed dynamic systems for their redundancy properties w.r.t. various given fault scenarios, we have gained knowledge about operation regions with enhanced safety properties. For example, we can determine regions that a system can be kept in after having it steered to this region, independently of any specific fault.

For the second part of this thesis, we will shift towards the second research question that is formulated in Chapter 1. That is, we will observe robustness properties of *control structures* w.r.t. parameter uncertainties in the plant model, or even fault events. Control structures are hereby primarily associated with the structure of the closed-loop transfer matrix $\mathbf{G}_{\mathrm{w}}(s)$ of a linear time-invariant system. We denote such structures by a *structure matrix*, for example by

$$\mathbf{G}_{\mathrm{w}}^{*} = \begin{bmatrix} * & 0 & 0 \\ 0 & * & 0 \\ * & * & * \end{bmatrix} \tag{3.1}$$

for a closed-loop system with three inputs and three outputs. Here, we indicate elements $g_{\mathrm{w}ij}$ of $\mathbf{G}_{\mathrm{w}}$ that may take some transfer function $g_{\mathrm{w}ij}(s) \neq 0$ by an element $g_{\mathrm{w}ij}^{*} = *$ of $\mathbf{G}_{\mathrm{w}}^{*}$. In contrast, elements for which we want to obtain $g_{\mathrm{w}ij}(s) \equiv 0$ are denoted with $g_{\mathrm{w}ij}^{*} = 0$. Allowing the structure matrix $\mathbf{G}_{\mathrm{w}}^{*}$ to be arbitrarily chosen as in Eq. (3.1) has not been covered in the literature. Many contributions in this field focus on standard structures like diagonal decoupling control or decentralized feedback separately, but not in a combined manner. Establishing the latter is the aim of this chapter.

Designing not only the closed-loop *dynamics* of some control system can occur for various reasons. For example, a controller could be designed such that given subsystems are fully synchronized or, in turn, decoupled from one another. As we will see later, this topic is also closely related to designing structured controllers, i.e. state or output feedback matrices with a prescribed structure. This connects our work to, e.g., PI-control, or decentralized feedback.

We will be able to make statements on the importance of, e.g., actuators, measurements, or, more general, signal paths, plant parameters, or control parameters for robustly realizing a control structure. Furthermore, we can differentiate whether these entities are important for the closed-loop control *structure* or rather its *dynamics*, i.e. for placing eigenvalues of the closed-loop system. As an example, it can be stated whether a control structure can be achieved by using, e.g., output feedback or decentralized control. Conditions for the stabilizability of control structures will be discussed as well.

Throughout this chapter, we will work with linear time-invariant systems of the form

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.2a}$$

$$\boldsymbol{y}(t) = \mathbf{C}\boldsymbol{x}(t) + \mathbf{D}\boldsymbol{u}(t) \tag{3.2b}$$

with $\boldsymbol{x}(t) \in \mathcal{X} = \mathbb{R}^n$, $\boldsymbol{u}(t) \in \mathbb{R}^m$, and $\boldsymbol{y}(t) \in \mathbb{R}^p$ as the state, input, and output vectors. In contrast to Chapter 2, the state and output space are not constrained in the following. We will denote the elements of the output equation as

$$\boldsymbol{y}(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_p(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_1^\top \\ \vdots \\ \boldsymbol{c}_p^\top \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} \boldsymbol{d}_1^\top \\ \vdots \\ \boldsymbol{d}_p^\top \end{bmatrix} \boldsymbol{u}(t). \tag{3.3}$$

The methodology that is developed in this chapter for conducting the above mentioned studies turns out to be constructive. This means that we will formulate a control problem for imposing a structure $\mathbf{G}_{\mathrm{w}}^*$ on the closed-loop transfer matrix. It is solved by means of the control law

$$\begin{aligned} \boldsymbol{u} &= \mathbf{F}_{\boldsymbol{y}} \boldsymbol{y}_{\mathrm{m}} + \mathbf{L}\boldsymbol{w}, \\ &= \mathbf{F}_{\boldsymbol{x}} \boldsymbol{x} + \mathbf{L}\boldsymbol{w}, \end{aligned} \tag{3.4}$$

where we denote by $\mathbf{F}_{\boldsymbol{y}} \in \mathbb{R}^{m \times p_{\mathrm{m}}}$ an output feedback w.r.t. the measurement output

$$\boldsymbol{y}_{\mathrm{m}} = \mathbf{C}_{\mathrm{m}} \boldsymbol{x}$$

with $\boldsymbol{y}_{\mathrm{m}} \in \mathbb{R}^{p_{\mathrm{m}}}$. The corresponding state feedback is $\mathbf{F}_{\boldsymbol{x}} = \mathbf{F}_{\boldsymbol{y}} \mathbf{C}_{\mathrm{m}}$. The prefilter matrix is $\mathbf{L} \in \mathbb{R}^{m \times p}$ and the reference input is denoted by $\boldsymbol{w} \in \mathbb{R}^p$. The statements of interest will then be obtained when constructing the matrices $\mathbf{F}_{\boldsymbol{y}}$ and $\mathbf{L}$. Hereby, the main idea is to translate the requirements for the closed-loop structure $\mathbf{G}_{\mathrm{w}}^*$ into requirements for the structure of the feedback controller $\mathbf{F}_{\boldsymbol{y}}$ and, if necessary, the prefilter $\mathbf{L}$. We will express these as linear equality constraints in the respective parameters as

$$\mathbf{Z}_{\mathrm{F}} \boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{z}_{\mathrm{F}} \tag{3.5a}$$

$$\mathbf{Z}_{\mathrm{L}} \boldsymbol{\ell} = \boldsymbol{z}_{\mathrm{L}} \tag{3.5b}$$

denoting by $\boldsymbol{f}_{\boldsymbol{y}}$ and $\boldsymbol{\ell}$ the vectorizations of $\mathbf{F}_{\boldsymbol{y}}$ and $\mathbf{L}$, i.e. $\boldsymbol{f}_{\boldsymbol{y}} = \mathrm{vec}(\mathbf{F}_{\boldsymbol{y}})$ and $\boldsymbol{\ell} = \mathrm{vec}(\mathbf{L})$, respectively.[1]

The approach is first demonstrated in a purely algebraic setting for an input-output decoupling structure, before presenting a procedure for incorporating arbitrary structures. As realizing an arbitrary control structure may be infeasible, depending on the plant, we will provide solutions for obtaining an approximation of the required structure in the closed loop.

3.1 Direct Approach for Diagonal Decoupling

Diagonal input-output decoupling is an extensively researched control design technique. In this section, we review the method published by Falb and Wolovich [49] for obtaining the constraints

[1] The vectorization of some matrix $\mathbf{M} = [\boldsymbol{m}_1 \ \cdots \ \boldsymbol{m}_b] \in \mathbb{R}^{a \times b}$ with $a, b \in \mathbb{N}$ is denoted by $\mathrm{vec}(\mathbf{M}) = [\boldsymbol{m}_1^\top \ \cdots \ \boldsymbol{m}_b^\top]^\top \in \mathbb{R}^{ab}$.

of the form (3.5) for a system (3.2) without feedthrough, i.e., we have $\mathbf{D} = \mathbf{0}$. Hence, our aim is to encode the diagonal structure

$$\mathbf{G}_{\mathrm{w}}^{*} = \begin{bmatrix} * & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & * \end{bmatrix} \tag{3.6}$$

into structures for the matrices $\mathbf{F}_{\boldsymbol{x}}$ and $\mathbf{L}$.[2] By applying the control law

$$\boldsymbol{u} = -\mathbf{D}^{*-1} \underbrace{\begin{bmatrix} \boldsymbol{c}_1^{\top}\mathbf{A}^{\delta_1} \\ \vdots \\ \boldsymbol{c}_p^{\top}\mathbf{A}^{\delta_p} \end{bmatrix}}_{=:\,\mathbf{E}_1} \boldsymbol{x} - \mathbf{D}^{*-1} \underbrace{\begin{bmatrix} \sum_{\nu=0}^{\delta_1-1} r_{1\nu}\boldsymbol{c}_1^{\top}\mathbf{A}^{\nu} \\ \vdots \\ \sum_{\nu=0}^{\delta_p-1} r_{p\nu}\boldsymbol{c}_p^{\top}\mathbf{A}^{\nu} \end{bmatrix}}_{=:\,\mathbf{E}_2} \boldsymbol{x} + \underbrace{\mathbf{D}^{*-1} \begin{bmatrix} r_{10} & & \\ & \ddots & \\ & & r_{p0} \end{bmatrix}}_{=\mathbf{L}} \boldsymbol{w} \tag{3.7}$$

designed in [49], it is possible to create the elements

$$g_{ii}(s) = \frac{r_{i0}}{s^{\delta_i} + r_{i,\delta_i-1}s^{\delta_i-1} + \cdots + r_{i0}}$$

of the transfer matrix $\mathbf{G}_{\mathrm{w}}(s)$ with arbitrarily chosen coefficients $r_{i\nu} \in \mathbb{R}$, while ensuring $g_{ij}(s) \equiv 0$ for all $i \neq j$. We have used the notation from Eq. (3.3), and the integers δ_i with $i \in \mathcal{I}_{1,p}$ represent the relative degree of the outputs y_i. We have assumed that the decoupling matrix

$$\mathbf{D}^{*} = \begin{bmatrix} \boldsymbol{c}_1^{\top}\mathbf{A}^{\delta_1-1}\mathbf{B} \\ \vdots \\ \boldsymbol{c}_p^{\top}\mathbf{A}^{\delta_p-1}\mathbf{B} \end{bmatrix}$$

is invertible such that the control law (3.7) is well defined. Hereby, we have implicitly assumed a square system (3.2), i.e. $m = p$. All developed results are, however, extendable to non-square systems with $m \neq p$, see Wahrburg and Adamy [166].

The parameters $r_{i\nu}$ can be used for placing the $n - \delta$ closed-loop poles of the system, where $\delta := \sum_{i=1}^{p} \delta_i$. Their values, however, do not influence the structure of $\mathbf{G}_{\mathrm{w}}(s)$. Therefore, it is our aim to obtain descriptions for $\mathbf{F}_{\boldsymbol{x}}$ and $\mathbf{L}$ that are independent of $r_{i\nu}$. Closely following the steps that have previously been published in [146], we start with rewriting the feedback controller $\mathbf{F}_{\boldsymbol{x}} = -\mathbf{D}^{*-1}(\mathbf{E}_1 + \mathbf{E}_2)$ from Eq. (3.7) as

$$\mathbf{D}^{*}\mathbf{F}_{\boldsymbol{x}} = -\mathbf{E}_1 - \mathbf{E}_2.$$

First transposing and then vectorizing this relation leads to [153, p. 162]

$$\left(\mathbf{D}^{*} \otimes \mathbf{I}_n\right) \mathrm{vec}(\mathbf{F}_{\boldsymbol{x}}^{\top}) = -\begin{bmatrix} \left(\mathbf{A}^{\top}\right)^{\delta_1} \boldsymbol{c}_1 \\ \vdots \\ \left(\mathbf{A}^{\top}\right)^{\delta_p} \boldsymbol{c}_p \end{bmatrix} - \begin{bmatrix} \sum_{\nu=0}^{\delta_1-1} r_{1\nu} \left(\mathbf{A}^{\top}\right)^{\nu} \boldsymbol{c}_1 \\ \vdots \\ \sum_{\nu=0}^{\delta_p-1} r_{p\nu} \left(\mathbf{A}^{\top}\right)^{\nu} \boldsymbol{c}_p \end{bmatrix} \tag{3.8}$$

[2] Since the method in [49] is designed for $\mathbf{C}_{\mathrm{m}} = \mathbf{I}_n$, we have $\mathbf{F}_{\boldsymbol{y}} = \mathbf{F}_{\boldsymbol{x}}$ and, therefore, $\boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{f}_{\boldsymbol{x}}$.

where the symbol $\otimes$ represents the Kronecker product. This form enables to split off the parameters $r_{i\nu}$ in a separate vector $\boldsymbol{r} = [r_{10} \quad r_{11} \quad \cdots \quad r_{20} \cdots \quad r_{p,\delta_p-1}]^\top$ by substituting the last term of (3.8) by

$$\begin{bmatrix} \sum_{\nu=0}^{\delta_1-1} r_{1\nu} \left(\mathbf{A}^\top\right)^\nu \boldsymbol{c}_1 \\ \vdots \\ \sum_{\nu=0}^{\delta_p-1} r_{p\nu} \left(\mathbf{A}^\top\right)^\nu \boldsymbol{c}_p \end{bmatrix} =: \begin{bmatrix} \mathbf{H}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{H}_p \end{bmatrix} \boldsymbol{r} := \mathbf{H}\boldsymbol{r}, \tag{3.9}$$

where $\mathbf{H} \in \mathbb{R}^{np\times\delta}$ is composed of blocks

$$\mathbf{H}_i = \begin{bmatrix} \boldsymbol{c}_i & \mathbf{A}^\top\boldsymbol{c}_i & \left(\mathbf{A}^\top\right)^{\delta_i-1}\boldsymbol{c}_i \end{bmatrix} \in \mathbb{R}^{n\times\delta_i}. \tag{3.10}$$

For later use, we record that $\mathbf{H}_i$ has full column rank:

Lemma 2 (see [146, Lem. 1]). *Given a system of the form* (3.2) *with* $\mathbf{D} = \mathbf{0}$*, let the relative degree* δ_i *of the output* y_i *be well-defined for each* $i \in \mathcal{I}_{1,p}$*. Then, the rank of* $\mathbf{H}_i$ *defined in Eq.* (3.10) *equals* δ_i*.*

Proof. [146] According to the definition of the relative degree δ_i, the sequence $\boldsymbol{c}_i^\top\mathbf{A}^\nu\mathbf{B} = \mathbf{0}$, $\nu \in \mathcal{I}_{0,\delta_i-2}$, holds. However, for $\nu = \delta_i - 1$, we have $\boldsymbol{c}_i^\top\mathbf{A}^{\delta_i-1}\mathbf{B} \neq \mathbf{0}$. Defining the subspaces

$$\mathcal{H}_i^\nu = \operatorname{im}\left(\begin{bmatrix} \boldsymbol{c}_i & \mathbf{A}^\top\boldsymbol{c}_i & \cdots & \left(\mathbf{A}^\top\right)^\nu\boldsymbol{c}_i \end{bmatrix}\right) \subseteq \mathbb{R}^n,$$

we can reformulate the above statement as $\mathcal{H}_i^\nu \subseteq \ker(\mathbf{B}^\top)$ for $\nu \in \mathcal{I}_{0,\delta_i-2}$. As in [160], we define $\mathcal{V}_1 + \mathcal{V}_2 := \{\boldsymbol{v}_1 + \boldsymbol{v}_2 \mid \boldsymbol{v}_1 \in \mathcal{V}_1,\ \boldsymbol{v}_2 \in \mathcal{V}_2\}$ for two vector spaces $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathbb{R}^N$ for some $N \in \mathbb{N}$. Clearly, $\mathcal{H}_i^{\nu+1} \supseteq \mathcal{H}_i^\nu$ holds, because $\mathcal{H}_i^{\nu+1} = \mathcal{H}_i^\nu + \operatorname{im}\left(\left(\mathbf{A}^\top\right)^{\nu+1}\boldsymbol{c}_i\right)$. If, for any $\nu^* < \delta_i - 1$ in the sequence, $\mathcal{H}_i^{\nu^*+1} = \mathcal{H}_i^{\nu^*}$, then $\left(\mathbf{A}^\top\right)^{\nu^*+1}\boldsymbol{c}_i \in \mathcal{H}_i^{\nu^*}$ is linearly dependent on the vectors spanning $\mathcal{H}_i^{\nu^*}$. It can thus be written as a sum of these vectors as $\left(\mathbf{A}^\top\right)^{\nu^*+1}\boldsymbol{c}_i = \sum_{j=0}^{\nu^*}\alpha_j\left(\mathbf{A}^\top\right)^j\boldsymbol{c}_i$ for some $\alpha_j \in \mathbb{R}$. Due to

$$\left(\mathbf{A}^\top\right)^{\nu^*+2}\boldsymbol{c}_i = \mathbf{A}^\top\left(\mathbf{A}^\top\right)^{\nu^*+1}\boldsymbol{c}_i = \sum_{j=0}^{\nu^*}\alpha_j\left(\mathbf{A}^\top\right)^{j+1}\boldsymbol{c}_i = \sum_{j=0}^{\nu^*}\alpha'_j\left(\mathbf{A}^\top\right)^j\boldsymbol{c}_i,$$

with $\alpha'_j \in \mathbb{R}$, we can deduce $\left(\mathbf{A}^\top\right)^{\nu^*+2}\boldsymbol{c}_i \in \mathcal{H}_i^{\nu^*}$ as well. Continuing this argumentation, $\mathcal{H}_i^{\nu^*+k} = \mathcal{H}_i^{\nu^*}$ holds for any $k \geq 0$. This is a contradiction to the relative degree δ_i being well-defined because $\left(\mathbf{A}^\top\right)^{\delta_i-1}\boldsymbol{c}_i \notin \ker(\mathbf{B}^\top)$ and, thus, $\mathcal{H}_i^{\delta_i-1}$ must strictly include $\mathcal{H}_i^{\delta_i-2}$. Therefore, the dimension of the subspaces $\mathcal{H}_i^\nu$ must be strictly increasing with ν for $\nu \in \mathcal{I}_{1,\delta_i-1}$. This results in $\dim\left(\mathcal{H}_i^{\delta_i-1}\right) = \delta_i$, which is equivalent to $\operatorname{rank}(\mathbf{H}_i) = \delta_i$. □

Theorem 8 (see [146, Th. 1]). *Given a system of the form* (3.2) *with* $\mathbf{D} = \mathbf{0}$*, let the relative degree* δ_i *of the output* y_i *be well-defined for each* $i \in \mathcal{I}_{1,p}$*. Then, the rank of* $\mathbf{H}$ *defined in Eq.* (3.9) *equals* δ*.*

Proof. [146] Since the rank of a block diagonal matrix equals the sum of the ranks of its block elements, the proof results from Lemma 2 and $\delta = \sum_{i=1}^{p} \delta_i$. □

The key idea for extracting the structure of $\mathbf{F}_{\boldsymbol{x}}$ for establishing the closed-loop structure (3.6) is decomposing the matrix $\mathbf{H}$ into its singular components. Using its singular value decomposition $\mathbf{H} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{np \times np}$ and $\mathbf{V} \in \mathbb{R}^{\delta \times \delta}$ are orthogonal matrices, we obtain from Eqs. (3.8) and (3.9)

$$\begin{aligned} & \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \operatorname{vec}(\mathbf{F}_{\boldsymbol{x}}^\top) = -\operatorname{vec}(\mathbf{E}_1) - \mathbf{U}\mathbf{S}\mathbf{V}^\top \boldsymbol{r} \\ \Leftrightarrow \qquad & \mathbf{U}^\top \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}} = -\mathbf{U}^\top \operatorname{vec}(\mathbf{E}_1) - \mathbf{S}\mathbf{V}^\top \boldsymbol{r}. \end{aligned}$$

Here, we have made use of $\mathbf{U}^{-1} = \mathbf{U}^\top$ which holds for orthogonal matrices, as well as the *commutation matrix* $\mathbf{K}^{(m,n)} \in \mathbb{R}^{nm \times nm}$ from [100] transforming $\operatorname{vec}(\mathbf{F}_{\boldsymbol{x}}^\top) = \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}}$. Due to Theorem 8, we have $\mathbf{S} = \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{0} \end{bmatrix}^\top$ with the singular values of $\mathbf{H}$ contained in the diagonal matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{\delta \times \delta}$. Partitioning $\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 \end{bmatrix}$ with $\mathbf{U}_1 \in \mathbb{R}^{np \times \delta}$ and $\mathbf{U}_2 \in \mathbb{R}^{np \times np-\delta}$, respectively, we arrive at

$$\begin{bmatrix} \mathbf{U}_1^\top \\ \mathbf{U}_2^\top \end{bmatrix} \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}} = -\begin{bmatrix} \mathbf{U}_1^\top \\ \mathbf{U}_2^\top \end{bmatrix} \operatorname{vec}(\mathbf{E}_1) - \begin{bmatrix} \boldsymbol{\Sigma} \\ \mathbf{0} \end{bmatrix} \mathbf{V}^\top \boldsymbol{r}.$$

This delivers the desired separation of the control structure and the dynamics as

$$\mathbf{U}_1^\top \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}} = -\mathbf{U}_1^\top \operatorname{vec}(\mathbf{E}_1) - \boldsymbol{\Sigma}\mathbf{V}^\top \boldsymbol{r} \tag{3.11a}$$

$$\mathbf{U}_2^\top \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}} = -\mathbf{U}_2^\top \operatorname{vec}(\mathbf{E}_1). \tag{3.11b}$$

Finding a controller $\mathbf{F}_{\boldsymbol{x}}$ satisfying Eq. (3.11b) clearly is a necessary condition for establishing the structure (3.6). If, however, we have determined controller parameters $\boldsymbol{f}_{\boldsymbol{x}}$ solving (3.11b), we can also solve Eq. (3.11a) by choosing

$$\boldsymbol{r} = -\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}_1^\top \left(\operatorname{vec}(\mathbf{E}_1) + \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \boldsymbol{f}_{\boldsymbol{x}}\right).$$

Hence, solving Eq. (3.11b) by some admissible $\boldsymbol{f}_{\boldsymbol{x}}$ is also sufficient for establishing (3.6). We collect this result in

Proposition 1 ([146, Proposition 3])**.** *Given a system of the form* (3.2) *without feedthrough, i.e.* $\mathbf{D} = \mathbf{0}$*, and* $\det(\mathbf{D}^*) \neq 0$*, all decoupling controllers* (3.7) *are defined by the structure established by* $\mathbf{Z}_{\mathrm{F}} \boldsymbol{f}_{\boldsymbol{x}} = \boldsymbol{z}_{\mathrm{F}}$ *with*

$$\mathbf{Z}_{\mathrm{F}} = \mathbf{U}_2^\top \left(\mathbf{D}^* \otimes \mathbf{I}_n\right) \mathbf{K}^{(m,n)} \qquad \text{and} \qquad \boldsymbol{z}_{\mathrm{F}} = -\mathbf{U}_2^\top \operatorname{vec}(\mathbf{E}_1).$$

Remark 8. For the control scheme published in [49], the prefilter matrix $\mathbf{L}$ is designed to obtain a steady-state gain of one, i.e. $g_{\mathrm{w}ii}(0) = 1$ for all $i \in \mathcal{I}_{1,p}$. It can be computed as

$$\mathbf{L} = \left(-\mathbf{C}(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}})^{-1}\mathbf{B}\right)^{-1}$$

after choosing $\boldsymbol{f}_{\boldsymbol{x}}$ and the dynamics parameters $\boldsymbol{r}$. Hence, we do not need to state extra constraints for the structure of $\mathbf{L}$ since it is automatically obtained by the design procedure.

3.2 Geometric Interpretation of the Design Task

In Section 3.1, we have used a purely algebraic approach to formulate the controller constraints. It is simple to apply and delivers a necessary and sufficient condition for establishing diagonal decoupling. However, it is only designed for complete state feedback and not applicable to more general control structures as the one exemplified in Eq. (3.1). To this end, we will switch to a fully geometric interpretation of the design of structured control systems. Substantial parts of the work presented in the sequel have previously been published in [148].

3.2.1 Invariance Concepts

Geometrically approaching the design task of finding a control law (3.4) that establishes a desired structure $\mathbf{G}_{\mathrm{w}}^{*}$ results in speaking in terms of control invariant or conditioned invariant subspaces, respectively. We have already made use of control invariant sets in Section 2.3, which is a closely related concept. Similar to Definition 16, we state

Definition 17 (Control invariant subspace, see [160, Def. 4.1]). *Given a system of the form* (3.2) *and an input function* $\boldsymbol{u}\colon [0;\, t) \to \mathbb{R}^m$*, let its solution at time* t *be denoted by* $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t)$*. A subspace* $\mathcal{V} \subseteq \mathcal{X}$ *is called control invariant if, for any* $\boldsymbol{x}_0 \in \mathcal{V}$*, there exists* $\boldsymbol{u}$ *such that* $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t) \in \mathcal{V}$ *for all* $t \geq 0$*.*

A fundamental property that we will make use of in control design is given by

Theorem 9 ([17, Th. 4.1.2]). *A subspace* $\mathcal{V} \subseteq \mathcal{X}$ *is a control invariant subspace if and only if there exists at least one matrix* $\mathbf{F}_{\boldsymbol{x}}$ *such that* $(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}})\boldsymbol{v} \in \mathcal{V}$ *for all* $\boldsymbol{v} \in \mathcal{V}$*, i.e.*

$$(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}})\mathcal{V} \subseteq \mathcal{V}. \tag{3.12}$$

Assume for now that $\mathbf{L} = \mathbf{0}$, i.e. $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$. Then, the theorem states that, for a control invariant subspace $\mathcal{V}$, we are able to compute at least one feedback controller $\mathbf{F}_{\boldsymbol{x}}$ that maps a state $\boldsymbol{x}(t)$ located in $\mathcal{V}$ to a state derivative $\dot{\boldsymbol{x}}(t)$ that is, again, located in $\mathcal{V}$. Applying such a controller causes a restriction of any control action to the control invariant subspace, because no trajectory outside this subspace will be excited, as soon as the state trajectory has entered it. For an illustration of this situation, see Fig. 3.1.

Let us develop an intuition why the concept of control invariance is important for our aim of realizing control structures $\mathbf{G}_{\mathrm{w}}^{*}$. These structures are associated with entries $g_{\mathrm{w}ij}(s)$ of the closed-loop transfer matrix $\mathbf{G}_{\mathrm{w}}(s)$, where $i,\, j \in \mathcal{I}_{1,p}$, that either are identically zero for all $s \in \mathbb{C}$, i.e. $g_{\mathrm{w}ij}^{*} = 0$, or attain some non-zero transfer function, i.e. $g_{\mathrm{w}ij}^{*} = *$. Assuming $\mathbf{D} = \mathbf{0}$ for the ease of presentation, the requirement $g_{\mathrm{w}ij}^{*} = 0$ implies that no state trajectories outside the kernel of $\boldsymbol{c}_i^{\top}$ describing the corresponding output $y_i = \boldsymbol{c}_i^{\top}\boldsymbol{x}$ must be excited by neither the controller nor the reference signal w_j, where we denote $\boldsymbol{w} = [w_1 \;\; \cdots \;\; w_p]^{\top}$. That is, it is our aim to choose

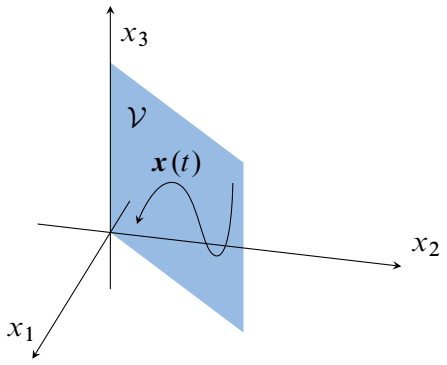


Figure 3.1: Illustration of a control invariant subspace $\mathcal{V}$. The state $\boldsymbol{x} \in \mathbb{R}^3$ only moves inside $\mathcal{V}$.

$\mathbf{F}_{\boldsymbol{x}}$ such that it is a *friend*[3] of some control invariant subspace $\mathcal{V} \subseteq \ker(\boldsymbol{c}_i^\top)$. More specifically, we will choose $\mathcal{V}$ as the maximal control invariant subspace $\mathcal{V}^* \subseteq \ker(\boldsymbol{c}_i^\top)$, which is defined analogously to the maximal control invariant set used in Section 2.3.1.

This choice has two reasons. First, there exist algorithms for uniquely computing $\mathcal{V}^*$. Second, all state components in the orthogonal complement $\mathcal{V}^\perp$ of $\mathcal{V}$ must be uncontrollable via w_j which needs to be accomplished using the prefilter matrix $\mathbf{L}$. These modes will not be controllable, also for the other signal paths $w_j \to y_k$ with $k \neq i$. This can impair their respective output controllability properties and may, therefore, be undesired. The latter can be mitigated by choosing $\mathcal{V}$ as large as possible.

Before we rigorously describe the geometric requirements for establishing $\mathbf{G}_{\mathrm{w}}^*$, we need another concept, i.e. *conditioned* invariance. For us, it is sufficient to interpret this property as the dual w.r.t. control invariance:

Definition 18 (Conditioned invariant subspace, see [17, Th. 4.1.3])**.** *Given a system of the form* (3.2) *with* $\mathbf{D} = \mathbf{0}$*, a subspace* $\mathcal{S} \subseteq \mathcal{X}$ *is called conditioned invariant if there exists at least one matrix* $\mathbf{F}$ *such that*

$$(\mathbf{A} + \mathbf{FC})\mathcal{S} \subseteq \mathcal{S}.$$

It is of interest for us because it is connected with the existence of an output feedback establishing control invariance of a subspace $\mathcal{V} \subseteq \mathcal{X}$. Referring to [17, p. 250], a subspace $\mathcal{V}$ needs to be control invariant and, simultaneously, conditioned invariant for guaranteeing the existence of a feedback law $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{y}}\boldsymbol{y}_{\mathrm{m}}$ with $\boldsymbol{y}_{\mathrm{m}} = \mathbf{C}_{\mathrm{m}}\boldsymbol{x}$ establishing $(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}})\mathcal{V} \subseteq \mathcal{V}$. For control invariance, we outlined before that we are interested in the maximal control invariant $\mathcal{V}^*$ contained in the kernel of the output map. This subspace can be numerically computed. Analogously, there exist computation algorithms for computing the *smallest* conditioned invariant $\mathcal{S}^*$ containing some other subspace $\mathcal{S} \subseteq \mathcal{X}$. However, to the best of the author's knowledge, there exists no algorithm for computing a maximal control and conditioned invariant contained in some other subspace. Therefore, in view of the design of the output feedback $\mathbf{F}_{\boldsymbol{y}}$, we will compute $\mathcal{V}^*$ and check this well-defined space for conditioned invariance w.r.t. the measurement $\boldsymbol{y}_{\mathrm{m}}$. Thereby, we obtain a

[3] A friend $\mathbf{F}_{\boldsymbol{x}}$ of $\mathcal{V}$ establishes (3.12).

feasible procedure, but lose necessity of the constraints (3.5). Note that, if $\operatorname{rank}(\mathbf{C}_\mathrm{m}) = n$, every control invariant $\mathcal{V}^*$ is simultaneously conditioned invariant. Thus, in such cases, necessity is preserved.

The concepts of control invariant and conditioned invariant subspaces have been set up for purely dynamic systems, i.e. systems with $\mathbf{D} = \mathbf{0}$. They can, however, be extended to systems with feedthrough by the addition of integrators in front of the inputs, or behind the outputs, respectively. Thereby, an auxiliary system without feedthrough is created that has $\dot{\boldsymbol{u}}$ as input, or $\int_0^t \boldsymbol{y}(\tau)\mathrm{d}\tau$ as output, respectively. Control invariant or conditioned invariant subspaces can be computed in the extended state space and then be projected back to the original state space. The resulting subspaces are called output nulling control invariant and input containing conditioned invariant, respectively. For a deeper insight, the interested reader is referred to Basile and Marro [17, p. 239 ff.]

3.2.2 Geometric Conditions Establishing a Structure $\mathbf{G}_\mathrm{w}^*$

Having outlined the connection between invariance and structural control design, we will now turn to developing the geometric conditions establishing a given structure $\mathbf{G}_\mathrm{w}^*$. Let us, therefore, denote the columns of $\mathbf{G}_\mathrm{w}^*$ by $\boldsymbol{g}_{\mathrm{w}j}^*$, i.e. $\mathbf{G}_\mathrm{w}^* = \begin{bmatrix}\boldsymbol{g}_{\mathrm{w}1}^* & \cdots & \boldsymbol{g}_{\mathrm{w}p}^*\end{bmatrix}$. The overall procedure is to iterate through the columns $\boldsymbol{g}_{\mathrm{w}j}^*$ of $\mathbf{G}_\mathrm{w}^*$, $j \in \mathcal{I}_{1,p}$, and to set up the geometric constraints for producing the elements $g_{\mathrm{w}ij} = 0$ that are demanded by the structure. For establishing a concise notation, let us introduce

Definition 19. *With every column $\boldsymbol{g}_{\mathrm{w}j}^*$ of the transfer structure $\mathbf{G}_\mathrm{w}^*$, we associate outputs of the form*

$$\boldsymbol{y}^j(t) = \mathbf{C}^j\boldsymbol{x}(t) + \mathbf{D}^j\boldsymbol{u}(t). \tag{3.13}$$

The matrices $\mathbf{C}^j$ and $\mathbf{D}^j$ are adapted to the structure of the column $\boldsymbol{g}_{\mathrm{w}j}^* = [g_{\mathrm{w}1j}^* \quad \cdots \quad g_{\mathrm{w}pj}^*]^\top$. Using the index set $\mathcal{M}_j^* := \{i \in \mathbb{N} \mid g_{\mathrm{w}ij}^* = 0\}$, they contain all row vectors $\boldsymbol{c}_i^\top$, or $\boldsymbol{d}_i^\top$, respectively, for which $i \in \mathcal{M}_j^*$. The remaining outputs are analogously defined as $\overline{\boldsymbol{y}}^j(t) = \overline{\mathbf{C}}^j\boldsymbol{x}(t) + \overline{\mathbf{D}}^j\boldsymbol{u}(t)$.

In view of the control law (3.4), the transfer behavior of each column of the closed loop is defined by

$$\boldsymbol{g}_{\mathrm{w}j}(s) = \left((\mathbf{C} + \mathbf{D}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})(s\mathbf{I}_n - \mathbf{A} - \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})^{-1} + \mathbf{D}\right)\mathbf{L}\boldsymbol{\alpha}_j,$$

where we have used the canonical unit vector $\boldsymbol{\alpha}_j \in \mathbb{R}^p$ in the j-th coordinate direction. Our aim is to choose the control law such that $\boldsymbol{y}^j(t) = \mathbf{0}$ holds, independently of $w_j(t)$, while retaining controllability of the other outputs $\overline{\boldsymbol{y}}^j(t)$ via $w_j(t)$. Inserting the control law (3.4) into Eq. (3.13), we obtain

$$\boldsymbol{y}^j(t) = (\mathbf{C}^j + \mathbf{D}^j\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})\boldsymbol{x}(t) + \mathbf{D}^j\mathbf{L}\boldsymbol{\alpha}_j w_j(t) = \mathbf{0},$$

which should hold for arbitrary quantities $\boldsymbol{x}(t) \in \mathcal{X}$ and $w_j(t) \in \mathbb{R}$. Because of this arbitrarity, this induces two conditions, one for each of the terms. They read

$$\boldsymbol{x}(t) \in \ker(\mathbf{C}^j + \mathbf{D}^j\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m}) \tag{3.14a}$$

$$\mathbf{D}^j\mathbf{L}\boldsymbol{\alpha}_j w_j(t) = \mathbf{0} \qquad \forall w_j(t) \in \mathbb{R}. \tag{3.14b}$$

First focusing on Eq. (3.14a), we require the trajectory of the state $\boldsymbol{x}$ to remain in some subspace $\mathcal{V}_j \subseteq \mathcal{X}$ satisfying

$$\mathcal{V}_j \subseteq \ker(\mathbf{C}^j + \mathbf{D}^j \mathbf{F}_{\boldsymbol{y}} \mathbf{C}_\mathrm{m}).$$

By Definition 17 and Theorem 9, this is directly connected to the condition

$$(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})\mathcal{V}_j \subseteq \mathcal{V}_j \tag{3.15}$$

and, therefore, Eq. (3.14a) requires $\mathbf{F}_{\boldsymbol{x}} = \mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m}$ to be chosen such that it is a friend of an output nulling subspace for the quadruple $(\mathbf{A}, \mathbf{B}, \mathbf{C}^j, \mathbf{D}^j)$. If $\mathbf{D}^j = \mathbf{0}$, the feedback $\mathbf{F}_{\boldsymbol{x}}$ needs to be a friend of a control invariant subspace contained in $\ker(\mathbf{C}^j)$, respectively.

Let us now focus on conditions for the prefilter matrix $\mathbf{L}$. One is given by Eq. (3.14b). A second one appears when looking at the closed-loop dynamics

$$\dot{\boldsymbol{x}}(t) = (\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})\boldsymbol{x}(t) + \mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j w_j(t).$$

No movement must be triggered outside $\mathcal{V}_j$, i.e., $\dot{\boldsymbol{x}}(t) \in \mathcal{V}_j$ must hold, independently of $\boldsymbol{x}(t)$ and $w_j(t)$. By establishing Eq. (3.15), this is guaranteed for the first term. The second term must then be chosen such that

$$\mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j w_j(t) \in \mathcal{V}_j \qquad \forall w_j(t) \in \mathbb{R}.$$

Rewriting this condition as well as Eq. (3.14b) in geometric terminology, we obtain

$$\begin{aligned} \mathrm{im}(\mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j) &\subseteq \mathcal{V}_j \\ \mathrm{im}(\mathbf{L}\boldsymbol{\alpha}_j) &\subseteq \ker(\mathbf{D}^j). \end{aligned}$$

Remark 9. By choosing the controller matrices $\mathbf{F}_{\boldsymbol{y}}$, and therefore $\mathbf{F}_{\boldsymbol{x}}$, and $\mathbf{L}$ adapted to the subspaces $\mathcal{V}_j$ for all $j \in \mathcal{I}_{1,p}$, the structure $\mathbf{G}_\mathrm{w}^*$ is established. However, note that there is no guarantee for the existence of such matrices. This is due to the fact that we try to find a *common* friend $\mathbf{F}_{\boldsymbol{x}}$ to a *set* of subspaces $\mathcal{V}_j$. Control invariance of a subspace $\mathcal{V}_j$, however, only guarantees the existence of some $\mathbf{F}_{\boldsymbol{x}}$ establishing Eq. (3.12) for this specific subspace $\mathcal{V}_j$. A similar argumentation holds for the prefilter matrix $\mathbf{L}$. As an example for the situation, consider diagonal decoupling as described in Section 3.1. If $\det(\mathbf{D}^*) = 0$, no controller $\mathbf{F}_{\boldsymbol{x}}$ establishing diagonal decoupling exists. Analogously, there exists no set of suitable control invariant subspaces $\mathcal{V}_j$ that possess a common friend.

Remark 10. We stated above that it is our aim to retain controllability of the outputs $\overline{\boldsymbol{y}}^j$. Defining $r_* \in \mathbb{N}$ as the structural rank of $\mathbf{G}_\mathrm{w}^*$ [103] and denoting the controllable subspace $\mathcal{W}$, see Eq. (2.9), of a pair $(\mathbf{A}, \mathbf{B})$ as $\mathcal{W} = \langle \mathbf{A} \mid \mathrm{im}(\mathbf{B}) \rangle$, this could be included as

$$\sum_{j=1}^{p} \overline{\mathbf{C}}^j \left\langle \mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m} \mid \mathrm{im}\big(\mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j\big) \right\rangle + \mathrm{im}\Big(\overline{\mathbf{D}}^j \mathbf{L}\boldsymbol{\alpha}_j\Big) = \mathbb{R}^{r_*}. \tag{3.16}$$

The above condition is difficult to be directly included in control design. Doing this would restrict the feasible set of subspaces $\mathcal{V}_j$ compatible with the design objectives defined by $\mathbf{G}_\mathrm{w}^*$. No technique is known to the author to find the corresponding feasible set of subspaces. Therefore, we will omit Condition (3.16) while computing the control law. However, it should be checked after control design whether the outputs $\overline{\boldsymbol{y}}^j$ are controllable in an acceptable manner.

Let us summarize the findings of this section by condensing them into a problem description and the respective solvability conditions:

Definition 20 (Structural control design)**.** *Given a plant of the form* (3.2) *and a structure* $\mathbf{G}_{\mathrm{w}}^{*}$ *of the closed-loop transfer matrix with elements* $g_{\mathrm{w}ij}^{*} \in \{0; *\}$ *with* $i, j \in \mathcal{I}_{1,p}$*, find a control law of the form* (3.4) *with a non-trivial prefilter* $\mathbf{L} \neq \mathbf{0}$ *establishing* $g_{\mathrm{w}ij}(s) = 0$ *for all elements with* $g_{\mathrm{w}ij}^{*} = 0$*.*

Proposition 2. *Structural control design as defined in Definition 20 is solvable if and only if there exist matrices* $\mathbf{F}_{\boldsymbol{y}}$ *and* $\mathbf{L}$ *and a set of compatible subspaces* $\mathcal{V}_j \subseteq \mathcal{X},\ j \in \mathcal{I}_{1,p}$ *satisfying*

1. $\mathcal{V}_j \subseteq \ker(\mathbf{C}^j + \mathbf{D}^j \mathbf{F}_{\boldsymbol{y}} \mathbf{C}_{\mathrm{m}})$
2. $(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}})\mathcal{V}_j \subseteq \mathcal{V}_j$
3. $\mathrm{im}(\mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j) \subseteq \mathcal{V}_j$
4. $\mathrm{im}(\mathbf{L}\boldsymbol{\alpha}_j) \subseteq \ker(\mathbf{D}^j)$

with the matrices $\mathbf{C}^j$ *and* $\mathbf{D}^j$ *defined in Definition 19.*

The proof can be obtained by applying Trentelman [160, Th. 7.11] and [17, p. 250]. There, the arguments are made for a single subspace $\mathcal{V}$ as opposed to a set of subspaces $\mathcal{V}_j$. This does, however, not change the structure of the problem.

Systems without feedthrough If we have a purely dynamic system with $\mathbf{D} = \mathbf{0}$ which implies $\mathbf{D}^j = \mathbf{0}$, Condition 4 of Proposition 2 becomes trivial since $\ker(\mathbf{D}^j) = \mathbb{R}^m$. It can therefore be omitted in this case.

3.3 Algebraic Conditions

In this section, we will transfer the geometric conditions summarized in Proposition 2 into a set of algebraic constraints of the form (3.5) formulated in the parameters of the controller matrices. First discussing the necessary steps for designing the control law for one plant model, we will later focus on an extension to the multiple-plant case that can be used for robustifying the control design in case of parameter uncertainties.

3.3.1 Single-Plant Case

As can be deduced from Proposition 2, formulating the constraints (3.5) relies heavily on the knowledge of the set of some subspaces $\mathcal{V}_j$ that simultaneously fulfill all conditions from the

proposition. Unfortunately, no algorithms are known for determining such a set. For this reason, we will use the maximal control invariant, or output nulling control invariant subspace, respectively, denoted by $\mathcal{V}_j^*$, instead. This has an influence on the results. Then, the conditions (3.5) only serve as sufficient solvability conditions in case that there exist $j_1, j_2 \in \mathcal{I}_{1,p}$ with $j_1 \neq j_2$ such that $\mathcal{V}_{j_1}^* \neq \mathcal{V}_{j_2}^*$. In turn, they retain their necessity in case $\mathcal{V}_{j_1}^* = \mathcal{V}_{j_2}^*$ for all $j_1, j_2 \in \mathcal{I}_{1,p}$. This is because, in the latter case, Proposition 2 merges exactly with Trentelman [160, Th. 7.11] establishing necessity and sufficiency of the conditions.

For every triple $(\mathbf{A}, \mathbf{B}, \mathbf{C}^j)$, or quadruple $(\mathbf{A}, \mathbf{B}, \mathbf{C}^j, \mathbf{D}^j)$, respectively, computing $\mathcal{V}_j^*$ can be achieved by evaluating the well-known algorithms that are presented and explained by Basile and Marro [17], Trentelman [160], and Wonham [173]. A numeric implementation is included in the toolbox [101].

In the following, let us refer to an orthonormal basis of $\mathcal{V}_j^*$ by the matrix $\mathbf{V}_j \in \mathbb{R}^{n \times v_j}$, i.e. $\mathrm{im}(\mathbf{V}_j) = \mathcal{V}_j^*$. Analogously defining an orthonormal basis $\mathbf{V}_j^\perp$ of its orthogonal complement, i.e. $\mathrm{im}(\mathbf{V}_j^\perp) = (\mathcal{V}_j^*)^\perp$, we can establish an orthonormal transformation matrix

$$\mathbf{T}_j := \begin{bmatrix} \mathbf{V}_j & \mathbf{V}_j^\perp \end{bmatrix}$$

naturally satisfying $\mathbf{T}_j^{-1} = \mathbf{T}_j^\top$. We can use the coordinate transformation $\boldsymbol{x} := \mathbf{T}_j \tilde{\boldsymbol{x}}$ for *adapting* the state space coordinates to the subspace $\mathcal{V}_j^*$. I.e., the first v_j coordinate axes are aligned with $\mathcal{V}_j^*$. Doing this, the coordinates of any vector $\tilde{\boldsymbol{x}} = [\tilde{x}_1 \quad \cdots \quad \tilde{x}_n]^\top \in \mathcal{V}_j^*$ satisfy $\tilde{x}_i = 0$ for all $i \in \mathcal{I}_{v_j+1,n}$. This is helpful as it induces a structure for the closed-loop system matrix if $\mathcal{V}_j^*$ is an invariant subspace in the closed loop. Then, we have

$$\tilde{\mathbf{A}}_{\mathrm{F}}^j(\mathbf{F}_{\boldsymbol{y}}) := \mathbf{T}_j^{-1}(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}})\mathbf{T}_j =: \begin{bmatrix} \tilde{\mathbf{A}}_{\mathrm{F11}}^j & \tilde{\mathbf{A}}_{\mathrm{F12}}^j \\ \tilde{\mathbf{A}}_{\mathrm{F21}}^j & \tilde{\mathbf{A}}_{\mathrm{F22}}^j \end{bmatrix},$$

for which

$$\tilde{\mathbf{A}}_{\mathrm{F21}}^j = (\mathbf{V}_j^\perp)^\top(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}})\mathbf{V}_j = \mathbf{0} \tag{3.17}$$

holds. This can be verified in view of the special structure of vectors $\tilde{\boldsymbol{x}} \in \mathcal{V}_j^*$, as discussed above. Choosing $\mathbf{F}_{\boldsymbol{y}}$ solving Eq. (3.17) satisfies Condition 2 of Proposition 2. Satisfaction of Condition 1 is equivalent to solving

$$\left(\mathbf{C}^j + \mathbf{D}^j\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\right)\mathbf{V}_j = \mathbf{0}.$$

The requirements can be rewritten as

$$\begin{aligned}(\mathbf{V}_j^\perp)^\top\mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\mathbf{V}_j &= -(\mathbf{V}_j^\perp)^\top\mathbf{A}\mathbf{V}_j \\ \mathbf{D}^j\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\mathbf{V}_j &= -\mathbf{C}^j\mathbf{V}_j,\end{aligned}$$

which are transformed to the form (3.5a) with

$$\begin{aligned}\mathbf{Z}_{\mathrm{F}j} &= \begin{bmatrix} (\mathbf{C}_{\mathrm{m}}\mathbf{V}_j)^\top \otimes (\mathbf{V}_j^\perp)^\top\mathbf{B} \\ (\mathbf{C}_{\mathrm{m}}\mathbf{V}_j)^\top \otimes \mathbf{D}^j \end{bmatrix} \\ \boldsymbol{z}_{\mathrm{F}j} &= \begin{bmatrix} \mathrm{vec}\left(-(\mathbf{V}_j^\perp)^\top\mathbf{A}\mathbf{V}_j\right) \\ \mathrm{vec}\left(-\mathbf{C}^j\mathbf{V}_j\right) \end{bmatrix}\end{aligned}$$

and

$$\mathbf{Z}_\mathrm{F} = \begin{bmatrix} \mathbf{Z}_{\mathrm{F}1}^\top & \cdots & \mathbf{Z}_{\mathrm{F}p}^\top \end{bmatrix}^\top, \qquad z_\mathrm{F} = \begin{bmatrix} z_{\mathrm{F}1}^\top & \cdots & z_{\mathrm{F}p}^\top \end{bmatrix}^\top. \tag{3.18}$$

The remaining Conditions 3 and 4 of Proposition 2 deliver constraints for the prefilter $\mathbf{L}$. Here, Condition 3 can easiest be transferred to an algebraic equation by expressing the subspace $\mathcal{V}_j$ in terms of its orthogonal complement as $\mathcal{V}_j = \ker(\mathcal{V}_j^\perp)$. Then, it reads $\mathrm{im}(\mathbf{B}\mathbf{L}\boldsymbol{\alpha}_j) \subseteq \ker(\mathcal{V}_j^\perp)$ leading to

$$\underbrace{\begin{bmatrix} (\mathbf{V}_j^\perp)^\top \mathbf{B} \\ \mathbf{D}^j \end{bmatrix}}_{=:\,\mathbf{Z}_{\mathrm{L}j}} \mathbf{L}\boldsymbol{\alpha}_j = \underbrace{\mathbf{0}}_{=:\,z_{\mathrm{L}j}}, \tag{3.19}$$

together with Condition 4. Stacking all quantities $\mathbf{Z}_{\mathrm{L}j}$ and $z_{\mathrm{L}j}$ for all $j \in \mathcal{I}_{1,p}$, we arrive at the prefilter constraints (3.5b) with

$$\mathbf{Z}_\mathrm{L} = \begin{bmatrix} \mathbf{Z}_{\mathrm{L}1} & & \\ & \ddots & \\ & & \mathbf{Z}_{\mathrm{L}p} \end{bmatrix}, \qquad z_\mathrm{L} = \begin{bmatrix} z_{\mathrm{L}1} \\ \vdots \\ z_{\mathrm{L}p} \end{bmatrix}. \tag{3.20}$$

Additional Controller Constraints

A strength of the proposed approach to designing a control law establishing a predefined closed-loop structure is that it is easily extendable by additional linear constraints that are directly imposed on the controller parameters. They embody additional rows in the system of equations (3.5a).

An example for the occurrence of such restrictions is decentralized control. The feedback matrix $\mathbf{F}_{\boldsymbol{y}}$ has an overall block structure, i.e.

$$\mathbf{F}_{\boldsymbol{y}} = \begin{bmatrix} \mathbf{F}_{\boldsymbol{y},11} & \cdots & \mathbf{F}_{\boldsymbol{y},1n_{\mathrm{d},2}} \\ \vdots & \ddots & \vdots \\ \mathbf{F}_{\boldsymbol{y},n_{\mathrm{d},1}1} & \cdots & \mathbf{F}_{\boldsymbol{y},n_{\mathrm{d},1}n_{\mathrm{d},2}} \end{bmatrix},$$

of which predefined blocks $\mathbf{F}_{\boldsymbol{y},ij}$ with $i \in \mathcal{I}_{1,n_{\mathrm{d},1}}$, $j \in \mathcal{I}_{1,n_{\mathrm{d},2}}$ are forced to zero.

A further example is given by the design of dynamic controllers such as PI-controllers. Enforcing the respective dynamics leads to a control problem with a structured feedback as well. Details can, e.g., be reviewed in [173, Ch. 8].

3.3.2 Multiple-Plant Case

Besides incorporating additional controller constraints, as previously discussed, the geometric approach to structural controller design enables a straight-forward extension for robustifying control design w.r.t. parameter uncertainties of the plant. This extension can as well be used to investigate the robustness properties of the control structure, as we have discussed it in Section 1.3.

Let us now consider a parameter-dependent version of the plant model in Eq. (3.2), i.e.

$$\dot{\boldsymbol{x}}_{\boldsymbol{\theta}}(t) = \mathbf{A}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{\theta}}(t) + \mathbf{B}(\boldsymbol{\theta})\boldsymbol{u}_{\boldsymbol{\theta}}(t) \tag{3.21a}$$

$$\boldsymbol{y}_{\boldsymbol{\theta}}(t) = \mathbf{C}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{\theta}}(t) + \mathbf{D}(\boldsymbol{\theta})\boldsymbol{u}_{\boldsymbol{\theta}}(t), \tag{3.21b}$$

where the parameter vector $\boldsymbol{\theta} \in \mathbb{R}^{\theta}$ can take values from the some set $\Theta = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{n_\theta}\}$, $n_\theta \in \mathbb{N}$. The control problem changes from finding a control law (3.4) establishing a structure $\mathbf{G}_{\mathrm{w}}^{*}$ for the *single* plant defined in Eq. (3.2) to finding the control law such that it establishes $\mathbf{G}_{\mathrm{w}}^{*}$ for the *family* of plants defined in Eq. (3.21).

Having discussed the procedure for determining the controller and prefilter structure for a single-plant control design in Section 3.3.1, we can easily proceed to the multiple-plant case. The respective constraints (3.5) can be obtained by setting up the single plant constraints defined by the Eqs. (3.18) and (3.20) for all parameters $\boldsymbol{\theta} \in \Theta$ and combining them in a common system of equations. Denoting the respective quantities by $\mathbf{Z}_{\mathrm{F}}^{\boldsymbol{\theta}}$, $\mathbf{Z}_{\mathrm{L}}^{\boldsymbol{\theta}}$, $\boldsymbol{z}_{\mathrm{F}}^{\boldsymbol{\theta}}$, and $\boldsymbol{z}_{\mathrm{L}}^{\boldsymbol{\theta}}$, we have

$$\mathbf{Z}_{\mathrm{F}} = \begin{bmatrix} \mathbf{Z}_{\mathrm{F}}^{\boldsymbol{\theta}_1} \\ \vdots \\ \mathbf{Z}_{\mathrm{F}}^{\boldsymbol{\theta}_{n_\theta}} \end{bmatrix}, \qquad \mathbf{Z}_{\mathrm{L}} = \begin{bmatrix} \mathbf{Z}_{\mathrm{L}}^{\boldsymbol{\theta}_1} \\ \vdots \\ \mathbf{Z}_{\mathrm{L}}^{\boldsymbol{\theta}_{n_\theta}} \end{bmatrix}, \qquad \boldsymbol{z}_{\mathrm{F}} = \begin{bmatrix} \boldsymbol{z}_{\mathrm{F}}^{\boldsymbol{\theta}_1} \\ \vdots \\ \boldsymbol{z}_{\mathrm{F}}^{\boldsymbol{\theta}_{n_\theta}} \end{bmatrix}, \qquad \boldsymbol{z}_{\mathrm{L}} = \begin{bmatrix} \boldsymbol{z}_{\mathrm{L}}^{\boldsymbol{\theta}_1} \\ \vdots \\ \boldsymbol{z}_{\mathrm{L}}^{\boldsymbol{\theta}_{n_\theta}} \end{bmatrix}.$$

By this approach, we seek a controller that is a common friend of a generally much larger set of control invariant subspaces $\mathcal{V}_{\boldsymbol{\theta}j}^{*}$ with $\boldsymbol{\theta} \in \Theta$ and $j \in \mathcal{I}_{1,p}$. Hence, also in this case, solvability of Eq. (3.5) is sufficient for the existence of a suitable controller, but not necessary.

Still, the procedure has some important advantages over existing techniques such as, e.g., modal control techniques which are presented in Magni [99]. Separating structural design from the dynamics design as shown in this thesis allows for a fully automatic control design where no manual tuning is needed for fulfilling the structural requirements imposed by $\mathbf{G}_{\mathrm{w}}^{*}$. Such tuning is needed for the procedure introduced in [99]. Depending on the number of parameter vectors $\boldsymbol{\theta}$, i.e. the value of n_θ, manually intervening in the control design may be practically infeasible.

Furthermore, with the method proposed here, we can gain insight regarding the system parameters, or signals, respectively, that are crucial for establishing $\mathbf{G}_{\mathrm{w}}^{*}$. Investigating robustness of the control structure w.r.t. plant parameters can be done in two ways. The first and obvious option is to try a set of parameters Θ and check solvability of Eq. (3.5). If solvability is preserved, this is a hint that the control structure can be realized by a constant control law even if the parameters that have been varied when creating Θ change during operation, or are only known with some uncertainty.

A second option that grants a deeper insight into the system structure is to analytically solve the invariant subspace algorithm [17, 160, 173] that is used to determine the subspaces $\mathcal{V}_{\boldsymbol{\theta}j}^{*}$. In this way, we can verify whether their orientation, or even their dimensions, depend on the chosen parameters $\boldsymbol{\theta}$. Parameters changing any of these quantities are likely to render Eq. (3.5) unsolvable and should, therefore, be free of uncertainty.

Solvability of Eq. (3.5) may in specific cases also depend on n_θ since for small families of plants, there may be enough degrees of freedom in the control matrices for compensating the parameter

changes. Due to an increase of n_θ, this might no longer be given and no control law can be found simultaneously establishing $\mathbf{G}_{\mathrm{w}}^*$ for the family of plants.

The dispensability of signal paths for the establishment of $\mathbf{G}_{\mathrm{w}}^*$ can be detected by the situation that all parameters of a row or a column in the control matrices can be chosen freely while fulfilling (3.5). Then, the measurement or the control signal paths may, e.g., be distorted without being critical for the structure $\mathbf{G}_{\mathrm{w}}^*$.

3.4 Approximating the Control Loop Structure

We have mentioned several times that, depending on the structure $\mathbf{G}_{\mathrm{w}}^*$ and the number n_θ of plant models, the constraints (3.5) might not possess a solution. Oftentimes, it is, however, acceptable to approximate the structure $\mathbf{G}_{\mathrm{w}}^*$ in the closed loop. This is because, generally, every plant model (3.2) or (3.21) is flawed and does not exactly mimic the true plant behavior. Hence, even if the control law exactly establishes $\mathbf{G}_{\mathrm{w}}^*$ for the model, the real world behavior will usually deviate due to the model-plant mismatch. Another reason for an approximate solution being acceptable is given if it is already a large achievement for the system if the controller can suppress present coupling effects significantly.

Besides, we will now argue why it is considerably easier to find a well-functioning approximation of (3.5) compared to explicitly searching for a set of subspaces $\mathcal{V}_{\boldsymbol{\theta} j} \subseteq \mathcal{V}_{\boldsymbol{\theta} j}^*$ that have a common friend $\mathbf{F}_{\boldsymbol{x}} = \mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}$, and also admit existence of an appropriately structured prefilter $\mathbf{L}$. To this end, let us state the problem that we would have to solve, formulated as a feasibility problem. It reads

$$
\begin{aligned}
\text{for all} \quad & \boldsymbol{\theta} \in \Theta,\ j \in \mathcal{I}_{1,p} && \text{(3.22a)}\\
\text{find} \quad & \mathbf{F}_{\boldsymbol{y}} \in \mathbb{R}^{m\times n},\ \mathbf{L} \in \mathbb{R}^{m\times p},\ v_{\boldsymbol{\theta} j} \in \mathbb{N},\ \mathbf{V}_{\boldsymbol{\theta} j} \in \mathbb{R}^{n\times v_{\boldsymbol{\theta} j}} && \text{(3.22b)}\\
\text{s.t.} \quad & \mathbf{V}_{\boldsymbol{\theta} j}^{\top}\mathbf{V}_{\boldsymbol{\theta} j}^{\perp} = \mathbf{0} && \text{(3.22c)}\\
& \mathbf{V}_{\boldsymbol{\theta} j}^{\top}\mathbf{V}_{\boldsymbol{\theta} j} = \mathbf{I}_{v_{\boldsymbol{\theta} j}} && \text{(3.22d)}\\
& (\mathbf{V}_{\boldsymbol{\theta} j}^{\perp})^{\top}\mathbf{V}_{\boldsymbol{\theta} j}^{\perp} = \mathbf{I}_{n-v_{\boldsymbol{\theta} j}} && \text{(3.22e)}\\
& (\mathbf{V}_{\boldsymbol{\theta} j}^{\perp})^{\top}\mathbf{B}(\boldsymbol{\theta})\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\mathbf{V}_{\boldsymbol{\theta} j} = -(\mathbf{V}_{\boldsymbol{\theta} j}^{\perp})^{\top}\mathbf{A}(\boldsymbol{\theta})\mathbf{V}_{\boldsymbol{\theta} j} && \text{(3.22f)}\\
& \mathbf{D}^j(\boldsymbol{\theta})\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\mathbf{V}_{\boldsymbol{\theta} j} = -\mathbf{C}^j(\boldsymbol{\theta})\mathbf{V}_{\boldsymbol{\theta} j} && \text{(3.22g)}\\
& (\mathbf{V}_{\boldsymbol{\theta} j}^{\perp})^{\top}\mathbf{B}(\boldsymbol{\theta})\mathbf{L}\boldsymbol{\alpha}_j = \mathbf{0} && \text{(3.22h)}\\
& \mathbf{D}^j(\boldsymbol{\theta})\mathbf{L}\boldsymbol{\alpha}_j = \mathbf{0}, && \text{(3.22i)}
\end{aligned}
$$

where Eqs. (3.22f) – (3.22i) represent the conditions from Proposition 2, and Eqs. (3.22c) – (3.22e) force $\begin{bmatrix}\mathbf{V}_{\boldsymbol{\theta} j} & \mathbf{V}_{\boldsymbol{\theta} j}^{\perp}\end{bmatrix}$ to be orthonormal bases of $\mathbb{R}^n$. The main challenge in solving Problem (3.22) is that the subspaces $\mathcal{V}_{\boldsymbol{\theta} j}$ represented by the respective orthonormal bases $\mathbf{V}_{\boldsymbol{\theta} j}$ are decision variables. There exist techniques to optimize over the manifold of orthonormal bases of predefined dimensions. The interested reader is referred to Absil et al. [1], Boumal [21], and Edelman et al. [46] for a detailed introduction to the topic. However, solving problem (3.22) remains challenging because the numbers $v_{\boldsymbol{\theta} j}$ defining the dimensions of the subspaces $\mathcal{V}_{\boldsymbol{\theta} j}$ are unknown as well.

In view of this situation, we choose to minimize the *leakage* of signals between the subspaces $\mathcal{V}^*_{\boldsymbol{\theta}j}$ and $(\mathcal{V}^*_{\boldsymbol{\theta}j})^\perp$, respectively. That is, we want to minimize the impact of the reference signals $w_j(t)$ on movements in $(\mathcal{V}^*_{\boldsymbol{\theta}j})^\perp$ as well as in $\mathrm{im}(\mathbf{D}^j(\boldsymbol{\theta}))$. Choosing the prefilter $\mathbf{L}$ accordingly, we can reduce the resulting impact of $w_j(t)$ on the outputs $\boldsymbol{y}^j(t)$, which should be zero, ideally. With the same argument, we seek $\mathbf{F}_{\boldsymbol{y}}$ to minimize the movement in $(\mathcal{V}^*_{\boldsymbol{\theta}j})^\perp$ triggered by a state vector $\boldsymbol{x}(t) \in \mathcal{V}^*_{\boldsymbol{\theta}j}$, and the transfer from states $\boldsymbol{x}(t) \in (\mathcal{V}^*_{\boldsymbol{\theta}j})^\perp$ to $\boldsymbol{y}^j(t)$.

Expressing this in terms of matrix norms, we can set up two objective functions J_F and J_L, respectively, reading

$$J_\mathrm{F} := \sum_{j=1}^{p} \sum_{\boldsymbol{\theta}\in\Theta} \left\| \left(\mathbf{C}^j(\boldsymbol{\theta}) + \mathbf{D}^j(\boldsymbol{\theta})\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m}\right)\mathbf{V}_{\boldsymbol{\theta}j} \right\|_\mathrm{F}^2 + \left\| (\mathbf{V}^\perp_{\boldsymbol{\theta}j})^\top (\mathbf{A}(\boldsymbol{\theta}) + \mathbf{B}(\boldsymbol{\theta})\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})\mathbf{V}_{\boldsymbol{\theta}j} \right\|_\mathrm{F}^2 \tag{3.23a}$$

$$J_\mathrm{L} := \sum_{j=1}^{p} \sum_{\boldsymbol{\theta}\in\Theta} \left\| \begin{bmatrix} (\mathbf{V}^\perp_{\boldsymbol{\theta}j})^\top \mathbf{B}(\boldsymbol{\theta}) \\ \mathbf{D}^j(\boldsymbol{\theta}) \end{bmatrix} \mathbf{L}\boldsymbol{\alpha}_j \right\|_\mathrm{F}^2, \tag{3.23b}$$

where $\|\cdot\|_\mathrm{F}^2$ denotes the squared Frobenius norm, and we use the maximal output nulling control invariant, i.e. $\mathrm{im}(\mathbf{V}_{\boldsymbol{\theta}j}) = \mathcal{V}^*_{\boldsymbol{\theta}j}$ and $\mathrm{im}((\mathbf{V}_{\boldsymbol{\theta}j})^\perp) = (\mathcal{V}^*_{\boldsymbol{\theta}j})^\perp$, respectively. Minimizing J_F and J_L is equivalent to minimizing the squared Frobenius norms of the error functions

$$\begin{aligned} \boldsymbol{e}_\mathrm{F}(\boldsymbol{f}_{\boldsymbol{y}}) &:= \mathbf{Z}_\mathrm{F}\boldsymbol{f}_{\boldsymbol{y}} - \boldsymbol{z}_\mathrm{F}, \\ \boldsymbol{e}_\mathrm{L}(\boldsymbol{\ell}) &:= \mathbf{Z}_\mathrm{L}\boldsymbol{\ell} - \boldsymbol{z}_\mathrm{L}. \end{aligned}$$

Hence, we have $J_\mathrm{F}(\boldsymbol{f}_{\boldsymbol{y}}) = \boldsymbol{e}_\mathrm{F}^\top(\boldsymbol{f}_{\boldsymbol{y}})\boldsymbol{e}_\mathrm{F}(\boldsymbol{f}_{\boldsymbol{y}})$ and $J_\mathrm{L}(\boldsymbol{\ell}) = \boldsymbol{e}_\mathrm{L}^\top(\boldsymbol{\ell})\boldsymbol{e}_\mathrm{L}(\boldsymbol{\ell})$. Minimizing them leads to solving the *normal equations*

$$\mathbf{Z}_\mathrm{F}^\top \mathbf{Z}_\mathrm{F} \boldsymbol{f}_{\boldsymbol{y}} = \mathbf{Z}_\mathrm{F}^\top \boldsymbol{z}_\mathrm{F} \tag{3.24a}$$

$$\mathbf{Z}_\mathrm{L}^\top \mathbf{Z}_\mathrm{L} \boldsymbol{\ell} = \mathbf{Z}_\mathrm{L}^\top \boldsymbol{z}_\mathrm{L}, \tag{3.24b}$$

which always possess at least one feasible solution [153, p. 296].

There are two situations, that need attention. First, the right side of Eq. (3.24b) is zero because $\boldsymbol{z}_\mathrm{L} = \mathbf{0}$ by Eqs. (3.19) and (3.20). Hence, $\boldsymbol{\ell} = \mathbf{0}$, i.e. $\mathbf{L} = \mathbf{0}$, is always a solution. In case of $\det(\mathbf{Z}_\mathrm{L}^\top \mathbf{Z}_\mathrm{L}) \neq 0$ it is the unique solution of (3.24b). This is, however, a highly undesired solution since the system is not controllable anymore by the reference input $\boldsymbol{w}$, see Remark 10.

A second issue arises if Eq. (3.5) does not only contain constraints establishing a closed-loop structure $\mathbf{G}^*_\mathrm{w}$, but also direct constraints for the controller matrices $\mathbf{F}_{\boldsymbol{y}}$ and $\mathbf{L}$. Then, also this control law structure is approximated, which might be undesired.

Let us provide solutions for handling the challenges when approximating the control law structure. A zero prefilter can be avoided by imposing an additional constraint

$$\mathbf{1}_{mp}^\top \boldsymbol{\ell} = c_{\boldsymbol{\ell}}, \qquad c_{\boldsymbol{\ell}} \in \mathbb{R} \setminus \{0\}. \tag{3.25}$$

Doing this, we demand the prefilter coefficients to sum up to some number $c_{\boldsymbol{\ell}} \neq 0$ forcing at least one prefilter parameter to be non-zero. Including this in the structure (3.5b), we first obtain

extended quantities

$$\tilde{\mathbf{Z}}_{\mathrm{L}} = \begin{bmatrix} \mathbf{Z}_{\mathrm{L}} \\ \mathbf{1}_{mp}^{\top} \end{bmatrix} \qquad \text{and} \qquad \tilde{z}_{\mathrm{L}} = \begin{bmatrix} z_{\mathrm{L}} \\ c_{\boldsymbol{\ell}} \end{bmatrix}$$

and the new approximating system of equations

$$\tilde{\mathbf{Z}}_{\mathrm{L}}^{\top} \tilde{\mathbf{Z}}_{\mathrm{L}} \boldsymbol{\ell} = \tilde{\mathbf{Z}}_{\mathrm{L}}^{\top} \tilde{z}_{\mathrm{L}}.$$

This equation has, again, at least one solution, which is known to be non-zero. This is due to the non-zero right side $\tilde{\mathbf{Z}}_{\mathrm{L}}^{\top} \tilde{z}_{\mathrm{L}} = \mathbf{1}_{mp} c_{\boldsymbol{\ell}}$ which can only be met by choosing $\boldsymbol{\ell} \neq \mathbf{0}$.

The proposed procedure guarantees finding some non-zero prefilter $\mathbf{L}$ while approximating the closed-loop structure $\mathbf{G}_{\mathrm{w}}^{*}$. This is a necessary condition for satisfying the output controllability condition (3.16). It is not sufficient. However, having the situation of Eq. (3.5b) not being solvable by a non-zero prefilter is already an indicator that the control design, as desired, might fail using the control law (3.4). Therefore, in this situation, it should be thoroughly analyzed which structural requirements lead to the difficulties in the design step. Further, it should be considered to relax some of them.

Remark 11. The parameter $c_{\boldsymbol{\ell}}$ in (3.25) is no degree of freedom for influencing the prefilter structure. Its specific value is not of interest since it only scales the steady-state gain of the closed loop. A suitable approach for achieving a prescribed steady-state reference tracking behavior while maintaining the required prefilter structure will be presented in Section 4.3. The method automatically compensates for the choice of $c_{\boldsymbol{\ell}}$.

Let us now focus the second challenge mentioned previously. It can be solved relatively easy by splitting the system of equations (3.5a) into two parts by defining

$$\mathbf{Z}_{\mathrm{F}} = \begin{bmatrix} \mathbf{Z}_{\mathrm{F}}^{\approx} \\ \mathbf{Z}_{\mathrm{F}}^{=} \end{bmatrix} \qquad \text{and} \qquad z_{\mathrm{F}} = \begin{bmatrix} z_{\mathrm{F}}^{\approx} \\ z_{\mathrm{F}}^{=} \end{bmatrix}.$$

Hereby, we separate constraints that must exactly be fulfilled from constraints that are acceptable to be approximated. They are characterized by the equations

$$\mathbf{Z}_{\mathrm{F}}^{\approx} \boldsymbol{f}_{\boldsymbol{y}} = z_{\mathrm{F}}^{\approx} \qquad \text{and} \tag{3.26a}$$

$$\mathbf{Z}_{\mathrm{F}}^{=} \boldsymbol{f}_{\boldsymbol{y}} = z_{\mathrm{F}}^{=}, \tag{3.26b}$$

respectively. As an example, constraints establishing a closed-loop structure $\mathbf{G}_{\mathrm{w}}^{*}$ could be gathered in Eq. (3.26a) while direct constraints on the controller structure, e.g. decentralized control, are collected in (3.26b). Such a split also creates new opportunities in weighting parts of the closed-loop structure according to their importance. Exemplarily, if it is of greater importance of realizing $\boldsymbol{g}_{\mathrm{w}1}^{*}$ compared to $\boldsymbol{g}_{\mathrm{w}2}^{*}$ of some structure $\mathbf{G}_{\mathrm{w}}^{*}$, the respective controller constraints for $\boldsymbol{g}_{\mathrm{w}2}^{*}$ could be allocated to Eq. (3.26a), and the constraints for $\boldsymbol{g}_{\mathrm{w}1}^{*}$ to Eq. (3.26b). In this light, it is also possible to compute the constraints for each element $g_{\mathrm{w}ij}^{*}$ of $\mathbf{G}_{\mathrm{w}}^{*}$ separately and allocate them in (3.26). An obvious assumption we have to make in the sequel is solvability of Eq. (3.26b).

If this assumption is met, we solve (3.26b) as

$$\boldsymbol{f}_{\boldsymbol{y}} = \left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{+} \boldsymbol{z}_{\mathrm{F}}^{=} + \left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{\perp} \boldsymbol{q}_{\mathrm{F}} \tag{3.27}$$

with $\left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{+}$ as the pseudo inverse and $\left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{\perp}$ as a kernel representation of $\mathbf{Z}_{\mathrm{F}}^{=}$. This can then be inserted into (3.26a) yielding

$$\underbrace{\mathbf{Z}_{\mathrm{F}}^{\approx} \left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{\perp}}_{=:\,\tilde{\mathbf{Z}}_{\mathrm{F}}} \boldsymbol{q}_{\mathrm{F}} = \underbrace{\boldsymbol{z}_{\mathrm{F}}^{\approx} - \mathbf{Z}_{\mathrm{F}}^{\approx} \left(\mathbf{Z}_{\mathrm{F}}^{=}\right)^{+} \boldsymbol{z}_{\mathrm{F}}^{=}}_{=:\,\tilde{\boldsymbol{z}}_{\mathrm{F}}}. \tag{3.28}$$

Eq. (3.28) is solvable if and only if (3.5a) is solvable – which we assume not to be the case, in this section. Hence, (3.28) is not solvable and an approximate minimum norm solution can, again, be found by solving

$$\tilde{\mathbf{Z}}_{\mathrm{F}}^{\top} \tilde{\mathbf{Z}}_{\mathrm{F}} \boldsymbol{q}_{\mathrm{F}} = \tilde{\mathbf{Z}}_{\mathrm{F}}^{\top} \tilde{\boldsymbol{z}}_{\mathrm{F}}. \tag{3.29}$$

However, by not directly approximating the original constraints (3.5a), we manage to exactly maintain the controller structure imposed by the constraints (3.26b).

Remark 12. The idea presented above can be refined to an arbitrary extent in two ways. Both of them require good knowledge about the constraint system (3.5a) and the impact of its components on the structure $\mathbf{G}_{\mathrm{w}}^{*}$ of the closed loop.

1. An ordered chain of constraint systems $\mathbf{Z}_{\mathrm{F}k}^{=} \boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{z}_{\mathrm{F}k}^{=}$ with $k \in \mathcal{I}_{1,n_{\mathrm{zf}}}$, $n_{\mathrm{zf}} \in \mathbb{N}$ can be introduced, such that

$$\begin{bmatrix} \mathbf{Z}_{\mathrm{F}1}^{=} \\ \vdots \\ \mathbf{Z}_{\mathrm{F}n_{\mathrm{zf}}}^{=} \end{bmatrix} \boldsymbol{f}_{\boldsymbol{y}} = \begin{bmatrix} \boldsymbol{z}_{\mathrm{F}1}^{=} \\ \vdots \\ \boldsymbol{z}_{\mathrm{F}n_{\mathrm{zf}}}^{=} \end{bmatrix}$$

   is equivalent to (3.5a). This chain is solved iteratively analogously to the procedure shown above, i.e. for increasing indices k, until a critical $k = k'$ is reached. The constraint system $\mathbf{Z}_{\mathrm{F}k'}^{=} \boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{z}_{\mathrm{F}k'}^{=}$ is not solvable and all constraints with $k \geq k'$ are approximated. All constraints with $k < k'$ are satisfied exactly.

2. The objective function J_{F} in Eq. (3.23a) can be adapted by introducing a positive definite weighting matrix $\mathbf{Q}_{\mathrm{F}} \in \mathbb{R}^{n_{\boldsymbol{e}_{\mathrm{F}}} \times n_{\boldsymbol{e}_{\mathrm{F}}}}$ with $n_{\boldsymbol{e}_{\mathrm{F}}} = \dim(\boldsymbol{e}_{\mathrm{F}})$. Using $J_{\mathrm{F}} = \boldsymbol{e}_{\mathrm{F}}^{\top} \mathbf{Q}_{\mathrm{F}} \boldsymbol{e}_{\mathrm{F}}$ shifts the controller structure such that it *softly* prioritizes specific constraints over others according to the weights in $\mathbf{Q}_{\mathrm{F}}$.

Remark 13. The techniques shown above for shaping the approximations of Eq. (3.5) are tailored to the specific situations occurring while designing $\mathbf{F}_{\boldsymbol{y}}$ or $\mathbf{L}$, respectively. E.g., it is of great importance to avoid $\mathbf{L} = \mathbf{0}$, which is not inherently needed for $\mathbf{F}_{\boldsymbol{y}}$ as long as $\mathbf{F}_{\boldsymbol{y}} = \mathbf{0}$ establishes a desired structure. However, depending on the user-defined design goals and the considered plant, it might be beneficial to use the approximation procedure initially designed for $\mathbf{F}_{\boldsymbol{y}}$ in the design for $\mathbf{L}$, and vice versa.

3.5 Stabilizability Analysis for Control Structures

We have put large effort in determining the structure of the control law (3.4) such that it establishes a closed-loop structure $\mathbf{G}_{\mathrm{w}}^{*}$ and fulfills secondary criteria as, e.g., a decentralization of the control action. Stability has not been considered in this process, so far. However, both domains, i.e. structuring the control loop and stabilizing it, are simultaneous tasks. Hence, it is necessary to discuss the dynamic implications of the control structure. This includes two important points:

1. We discuss whether closed-loop eigenvalues are fixed by imposing a controller structure $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}}) = \mathbf{F}_{\boldsymbol{y}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{C}_{\mathrm{m}}$. The set of fixed eigenvalues is given by [41]

$$\sigma_{\mathrm{fix}} = \bigcap_{\boldsymbol{q}_{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}}} \sigma(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})),$$

   where σ denotes the matrix spectrum and $\boldsymbol{q}_{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}}$ represents the degrees of freedom in solving Eq. (3.5a) as

$$\boldsymbol{f}_{\boldsymbol{y}} = (\mathbf{Z}_{\mathrm{F}})^{+}\, \boldsymbol{z}_{\mathrm{F}} + (\mathbf{Z}_{\mathrm{F}})^{\perp}\, \boldsymbol{q}_{\mathrm{F}}$$

   with, again, $(\mathbf{Z}_{\mathrm{F}})^{+}$ as the pseudo inverse and $(\mathbf{Z}_{\mathrm{F}})^{\perp}$ as a kernel representation of $\mathbf{Z}_{\mathrm{F}}$. We have assumed that Eq. (3.5a) has been designed such that it is solvable.

   The situation of structurally fixed eigenvalues is well-known to occur for diagonal decoupling if the relative degree δ of the system is smaller than its state dimension n. When establishing the structure by choosing a decoupling controller, $n - \delta$ eigenvalues are automatically fixed at the positions of invariant zeros of the plant. If such fixations occur it is important to know whether the corresponding eigenvalues are stable or unstable. The latter renders a stable control design infeasible.

2. Furthermore, we will investigate whether there exist enough degrees of freedom within the control law structure for stabilizing the plant, i.e. shifting any unstable, non-fixed open-loop eigenvalues to the negative complex half plane.

Let us briefly recall stabilizability for linear systems such that we can refer to a concise definition of it:

Definition 21 (see [48, Def. 8.5])**.** *A system of the form* (3.2) *is said to be stabilizable if all its uncontrollable modes are asymptotically stable, i.e. if the uncontrollable eigenvalues* $\lambda \in \sigma(\mathbf{A})$ *fulfill* $\mathrm{Re}(\lambda) < 0$.

The aim of this section is in fact to investigate if the parameter-dependent matrix $\mathbf{A}_{\mathrm{F}} := \mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})$ admits at least one parameterization such that its spectrum lies in the negative complex half plane. This question can be approached in various different ways.

A natural approach is to compute the characteristic polynomial of $\mathbf{A}_{\mathrm{F}}$ as

$$P_{\mathrm{F}}(s) := \det(s\mathbf{I}_n - \mathbf{A}_{\mathrm{F}}) = s^n + a_{n-1}(\boldsymbol{q}_{\mathrm{F}})s^{n-1} + \cdots + a_1(\boldsymbol{q}_{\mathrm{F}})s + a_0(\boldsymbol{q}_{\mathrm{F}}) \tag{3.30}$$

whose coefficients a_i, $i \in \mathcal{I}_{0,n-1}$ are functions of the parameters $\boldsymbol{q}_\mathrm{F}$. The problem is now to prove the existence of some $\boldsymbol{q}_\mathrm{F}$ satisfying, e.g., the Hurwitz criterion [65, 96]. Depending on the system, this can be a challenging task since the coefficients a_i are itself polynomials in the components $q_{\mathrm{F}j}$ of $\boldsymbol{q}_\mathrm{F} = [q_{\mathrm{F}1} \quad \cdots \quad q_{\mathrm{F}n_\mathrm{F}}]^\top \in \mathbb{R}^{n_\mathrm{F}}$. That is, they are nonlinear functions in $\boldsymbol{q}_\mathrm{F}$.

A related approach is to formulate a Lyapunov stability criterion, i.e. [97]

$$\mathbf{A}_\mathrm{F}^\top(\boldsymbol{q}_\mathrm{F})\mathbf{P} + \mathbf{P}\mathbf{A}_\mathrm{F}(\boldsymbol{q}_\mathrm{F}) \prec 0, \tag{3.31}$$

where the symbol $\prec$ indicates negative definiteness of the left side. Then, the problem is to find a positive definite matrix $\mathbf{P} \succ 0$ and a parameter $\boldsymbol{q}_\mathrm{F}$ satisfying the condition. This is by no means an easy task because the search for $\mathbf{P}$ is conducted for a set of matrices $\mathbf{A}_\mathrm{F}$ instead of a single matrix. To the best of the authors knowledge, no applicable results for handling this problem class are present in the literature.

As we are interested on controllability properties of the plant under the restriction of a pre-structured controller, we can also interpret this question as a *structural controllability problem*. The idea of structural controllability analysis and some results are collected, e.g., by Lunze [97]. Using graph representations of the signal flows in the system, the whole class of systems possessing the same structure in terms of these signal flows can be analyzed for controllability. Structural controllability is given if there exists at least one parameterization of the respective system structure that is controllable. This analysis technique has its strength in proving the absence of structural controllability. I.e., it can be shown that a given system structure does not admit any parameterization that is controllable. This is a valuable statement in system design. However, for our purpose, it is rather a disadvantage because we want to confirm controllability of the present system structure. Such a confirmation is indeed given by the structural controllability techniques, but it is not clear whether the present parameterization is also a controllable one. Furthermore, it is not possible to map parameter dependencies that occur between the controller parameters into the framework. Such dependencies occur due to the nature of Eq. (3.5a) and can have an impact on the ability of the structured controller to assign the system eigenvalues.

The approaches presented are chosen to illustrate the difficulty of making a statement about the ability of an arbitrarily structured controller (3.5a) to stabilize the plant. In the sequel, we will therefore also use information about the origin of the controller structure. Using the invariance properties that are established by a controller helps us to check the systems for a priori fixed eigenvalues.

3.5.1 Structures Establishing Invariance

The theoretical results used for identifying eigenvalues that are automatically assigned by establishing a certain invariance structure to the control loop are taken from Trentelman [160]. For a deeper insight, the interested reader is therein referred to Sections 4.4 and 4.5 covering pole placement under invariance constraints by state feedback. At first, let us introduce the concept of a controllability subspace.

Definition 22 (Controllability subspace [160, Def. 4.11]). *A subspace $\mathcal{C} \subseteq \mathcal{X}$ is called a controllability subspace if for every $\boldsymbol{x}_0 \in \mathcal{C}$ there exists a time $T > 0$ and an input function $\boldsymbol{u}\colon [0;\, T) \to \mathbb{R}^m$ such that the system solution $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, t) \in \mathcal{C}$ for all $t \in [0;\, T]$ and $\boldsymbol{\Phi}(\boldsymbol{x}_0,\, \boldsymbol{u},\, T) = \mathbf{0}$.*

A controllability subspace is, by Definition 22, a control invariant subspace. However, it is designed such that there exists an input trajectory driving any state from $\mathcal{C}$ to the origin in finite time, i.e. an input stabilizing the system. This property has not been included in the control invariant subspaces $\mathcal{V}$, $\mathcal{V}_j^*$, or $\mathcal{V}_{\boldsymbol{\theta} j}^*$ that we have used so far. Due to [160, Th. 4.17], we can compute the maximal controllability subspace included in some control invariant subspace $\mathcal{V} \subseteq \mathcal{X}$. It reads

$$\mathcal{C}^*(\mathcal{V}) := \langle \mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}} \mid \mathrm{im}(\mathbf{B}) \cap \mathcal{V} \rangle\,,$$

where the state feedback $\mathbf{F}_{\boldsymbol{x}}$ can be chosen as an arbitrary friend of $\mathcal{V}$. Further introducing an auxiliary subspace

$$\mathcal{D} := \mathcal{V} + \langle \mathbf{A} \mid \mathrm{im}(\mathbf{B}) \rangle$$

establishes a *chain* of subspaces $\mathcal{C}^* \subseteq \mathcal{V} \subseteq \mathcal{D} \subseteq \mathcal{X}$. This chain allows for a specific decomposition of the system matrices. To obtain it, let us define the regular state transformation

$$\boldsymbol{x} := \mathbf{T}\tilde{\boldsymbol{x}} := \begin{bmatrix} \mathbf{T}_1 & \mathbf{T}_2 & \mathbf{T}_3 & \mathbf{T}_4 \end{bmatrix} \tilde{\boldsymbol{x}}$$

whose components are adapted to the chain. I.e., we choose them such that $\mathrm{im}(\mathbf{T}_1) = \mathcal{C}^*$, $\mathrm{im}\left(\begin{bmatrix} \mathbf{T}_1 & \mathbf{T}_2 \end{bmatrix}\right) = \mathcal{V}$, and $\mathrm{im}\left(\begin{bmatrix} \mathbf{T}_1 & \mathbf{T}_2 & \mathbf{T}_3 \end{bmatrix}\right) = \mathcal{D}$. Matrix $\mathbf{T}_4$ finally needs to establish regularity of $\mathbf{T}$. Transforming the closed-loop system matrix to the adapted coordinate system admits the decomposition [160]

$$\mathbf{T}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}})\mathbf{T} =: \tilde{\mathbf{A}} + \tilde{\mathbf{B}}\tilde{\mathbf{F}}_{\boldsymbol{x}} := \begin{bmatrix} \tilde{\mathbf{A}}_{11} & \tilde{\mathbf{A}}_{12} & \tilde{\mathbf{A}}_{13} & \tilde{\mathbf{A}}_{14} \\ \tilde{\mathbf{A}}_{21} & \tilde{\mathbf{A}}_{22} & \tilde{\mathbf{A}}_{23} & \tilde{\mathbf{A}}_{24} \\ \tilde{\mathbf{A}}_{31} & \tilde{\mathbf{A}}_{32} & \tilde{\mathbf{A}}_{33} & \tilde{\mathbf{A}}_{34} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \tilde{\mathbf{A}}_{44} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}}_1 \\ \tilde{\mathbf{B}}_2 \\ \tilde{\mathbf{B}}_3 \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{F}}_{\boldsymbol{x},1} & \tilde{\mathbf{F}}_{\boldsymbol{x},2} & \tilde{\mathbf{F}}_{\boldsymbol{x},3} & \tilde{\mathbf{F}}_{\boldsymbol{x},4} \end{bmatrix} \tag{3.32}$$

which has a specific structure due to the invariance properties of the different subspaces. This is a similar situation as in Eq. (3.17) when deriving the structural constraints for the controller. E.g., the zero blocks in the last row of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$ are present since $\mathcal{D}$ represents all controllable directions of the system. Hence, once the system state lies in $\mathcal{D}$ it will naturally stay in $\mathcal{D}$ for all times, i.e., $\mathcal{D}$ is control invariant. Now, as $\mathbf{F}_{\boldsymbol{x}}$ should establish invariance of $\mathcal{V}$, we have to actively ensure $\tilde{\mathbf{A}}_{31} + \tilde{\mathbf{B}}_3\tilde{\mathbf{F}}_{\boldsymbol{x},1} = \mathbf{0}$ and $\tilde{\mathbf{A}}_{32} + \tilde{\mathbf{B}}_3\tilde{\mathbf{F}}_{\boldsymbol{x},2} = \mathbf{0}$. Choosing any $\tilde{\mathbf{F}}_{\boldsymbol{x},2}$ fulfilling the latter assigns a fixed spectrum σ to the block $\tilde{\mathbf{A}}_{22} + \tilde{\mathbf{B}}_2\tilde{\mathbf{F}}_{\boldsymbol{x},2}$ [160, Th. 4.18]. Together with the set of inherently uncontrollable eigenvalues of the plant, i.e. $\sigma(\tilde{\mathbf{A}}_{44})$, we can identify all fixed eigenvalues that emerge when establishing invariance of $\mathcal{V}$.

This procedure can be conducted for every member of the set of subspaces $\mathcal{V}_j^*$, or $\mathcal{V}_{\boldsymbol{\theta} j}^*$, respectively, with $j \in \mathcal{I}_{1,p}$ and $\boldsymbol{\theta} \in \Theta$ used in Sections 3.3.1 and 3.3.2. Any eigenvalue that is fixed by establishing invariance of one subspace $\mathcal{V}_j^*$, or $\mathcal{V}_{\boldsymbol{\theta} j}^*$ will also be fixed if invariance of the whole set of subspaces is established. Hence, from this collection of eigenvalues we can derive whether

unstable eigenvalues are inevitably assigned by establishing invariance. If this is the case, no corresponding stabilizing controller exists.

If all a priori fixed eigenvalues are stable, the question is certainly whether the remaining controllable eigenvalues can be assigned stable. In view of the decomposition (3.32), this can be confirmed for the single-plant case with

$$\mathcal{V}_i^* = \mathcal{V}_j^* \qquad \forall i, j \in \mathcal{I}_{1,p}, \tag{3.33}$$

that is the case where each column $\boldsymbol{g}_{\mathrm{w}j}^*$ of the transfer structure induces the same invariance constraints. Then, the eigenvalues of $\tilde{\mathbf{A}}_{11} + \tilde{\mathbf{B}}_1\tilde{\mathbf{F}}_{\boldsymbol{x},1}$ and $\tilde{\mathbf{A}}_{33} + \tilde{\mathbf{B}}_3\tilde{\mathbf{F}}_{\boldsymbol{x},3}$ can be assigned freely [160]. Eq. (3.33) is quite restrictive. However, for single-plant structures embodying state or output synchronization tasks by state feedback, we can obtain powerful results. For all other cases, this is not given since the controller cannot freely be chosen while establishing invariance of one single subspace $\mathcal{V}_j^*$, or $\mathcal{V}_{\boldsymbol{\theta}j}^*$, as it needs to *simultaneously* ensure invariance for the set of subspaces.

3.5.2 Parametric Structures

In this section, we will tackle the problem of giving stabilizability statements for controller constraints that do not originate from invariance considerations, but are simply given in the form (3.5a) not allowing for system theoretical interpretation. We choose a fully numerical approach now and provide answers to both questions of interest. I.e., whether there exist fixed eigenvalues due to choosing the structured controller, and whether the degrees of freedom in the controller suffice for finding at least one parameterization stabilizing all plant models considered. The results regarding the latter are sufficient but not necessary, as they are obtained by a bisection algorithm that can only provide statements for a bounded set of parameters. The presented methods have been gathered in the Master thesis [57].

3.5.2.1 Identifying Fixed Eigenvalues

In 1977, Davison et al. [41] have developed a procedure for checking controllability properties of high-dimensional linear time-invariant systems. The main goal of this work was to avoid evaluating rank conditions since they were computationally expensive at this time. Such rank conditions would occur in view of, e.g., the well-known Kalman criterion

$$\operatorname{rank}\left(\begin{bmatrix}\mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{bmatrix}\right) = n,$$

or the Hautus criterion [61]

$$\operatorname{rank}\left(\begin{bmatrix}\lambda\mathbf{I}_n - \mathbf{A} & \mathbf{B}\end{bmatrix}\right) = n \qquad \forall\, \lambda \in \sigma(\mathbf{A}).$$

Their proposed algorithm is based on the observation stated in the following Lemma. Its proof is built on prior results from Davison and Wang [40].

Lemma 3 (see [41, Lem. 1]). *Given a system of the form* (3.2)*, the class of feedback matrices* $\mathbf{F}_{\boldsymbol{x}} \in \mathbb{R}^{m\times n}$*, which does not result in the fixed eigenvalues of* (3.2) *being equal to those eigenvalues of* $\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}$ *which are common with the eigenvalues of* $\mathbf{A}$*, is either empty or lies on a hyper-surface in the parameter space of* $\mathbf{F}_{\boldsymbol{x}}$*.*

The Lemma sounds quite involved but its statement can in fact easily be understood. Given the spectrum of the system matrix $\sigma(\mathbf{A})$, the set of all controllers $\mathbf{F}_{\boldsymbol{x}}$ producing a closed-loop system matrix with $\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}})$ such that

$$\sigma(\mathbf{A}) \cap \sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}) \neq \sigma_{\text{fix}} \tag{3.34}$$

is a hyperplane in the parameter space of $\mathbf{F}_{\boldsymbol{x}}$, or an empty set. We have discussed in Section 2.2.4 that a non-full-dimensional hyperplane in some space has Lebesgue measure zero. Therefore, the probability of randomly choosing an $\mathbf{F}_{\boldsymbol{x}}$ such that Eq. (3.34) is fulfilled equals zero. Due to this observation, a simple way to identify fixed eigenvalues that are created by constraining a controller $\mathbf{F}_{\boldsymbol{y}}$ according to (3.5a) is given by computing the spectra $\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\mathbf{0}))$ and $\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\text{F}}))$, where the degrees of freedom $\boldsymbol{q}_{\text{F}}$ are randomly chosen. Their intersection $\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\mathbf{0}))\cap\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\text{F}}))$ equals the set of fixed eigenvalues σ_{fix} *with probability one.* This means that checking two specific parameters suffices almost surely for correctly computing σ_{fix}. If the result does not seem trustworthy the test can simply be repeated [41].

In contrast to Lemma 3, $\mathbf{F}_{\boldsymbol{x}}=\mathbf{0}$ might not be an admissible controller w.r.t. Eq. (3.5a). Therefore, we have substituted this choice by an appropriately structured controller with all free parameters set to zero. Since $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\text{F}})$ has an affine dependence on $\boldsymbol{q}_{\text{F}}$, i.e., it can be written as

$$\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\text{F}}) = \mathbf{F}_{\boldsymbol{x},0} + \sum_{i=1}^{n_{\text{F}}} \mathbf{F}_{\boldsymbol{x},i} q_{\text{F}i},$$

the argument of Lemma 3 still holds because $\mathbf{F}_{\boldsymbol{x},0}$ can be used to create an auxiliary system matrix $\mathbf{A}' = \mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x},0}$. Choosing $\boldsymbol{q}_{\text{F}}=\mathbf{0}$ therefore results in investigating the open-loop system matrix of the auxiliary system, whence the alignment with Lemma 3.

Following this procedure, the eigenvalues assigned by establishing the structure defined by (3.5a) can be identified for each parameterization (3.21) with $\boldsymbol{\theta}\in\Theta$ in the multiple-plant case. If unstable fixed eigenvalues occur for any $\boldsymbol{\theta}\in\Theta$, the family of plants is not simultaneously stabilizable using this controller structure.

3.5.2.2 Existence of Stabilizing Structured Controllers

As can be seen from the previous section, it is possible to confidently identify structurally fixed eigenvalues with a low computational effort. In this section, we present an algorithm for finding a set of parameters $\mathcal{Q}_{\text{F}} \subseteq \mathbb{R}^{n_{\text{F}}}$ rendering the closed-loop characteristic polynomial $P_{\text{F}}(s)$ defined in Eq. (3.30) a Hurwitz polynomial. That is the natural approach we introduced first. As already discussed, we need to analyze the coefficients $a_i(\boldsymbol{q}_{\text{F}})$ of $P_{\text{F}}(s)$ which are themselves polynomials in $\boldsymbol{q}_{\text{F}}$. More concrete, the Hurwitz criterion states

Theorem 10 (see [65, p. 274] and [96, Th. 8.5]). *Given the polynomial $P_{\mathrm{F}}(s)$ defined in Eq. (3.30), let*

$$\Delta_k := \det\left(\begin{bmatrix} a_{n-1}^{\Delta} & a_{n-3}^{\Delta} & a_{n-5}^{\Delta} & a_{n-7}^{\Delta} & \cdots & \\ a_n^{\Delta} & a_{n-2}^{\Delta} & a_{n-4}^{\Delta} & a_{n-6}^{\Delta} & \cdots & \\ 0 & a_{n-1}^{\Delta} & a_{n-3}^{\Delta} & a_{n-5}^{\Delta} & \cdots & \\ 0 & a_n^{\Delta} & a_{n-2}^{\Delta} & a_{n-4}^{\Delta} & \cdots & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \\ & & & & & a_{n-k}^{\Delta} \end{bmatrix}\right) \quad \textit{with} \quad a_i^{\Delta} = \begin{cases} a_i & i \in \mathcal{I}_{0,n-1} \\ 1 & i = n \\ 0 & \text{otherwise} \end{cases}.$$

All roots of $P_{\mathrm{F}}(s)$ have negative real parts if and only if $\Delta_k > 0$ for all $k \in \mathcal{I}_{1,n}$.

By Theorem 10, $\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})$ is stabilizable if and only if there exists $\boldsymbol{q}_{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}}$ such that the determinants Δ_k are positive for all $k \in \mathcal{I}_{1,n}$. In view of the fact that all coefficients of $P_{\mathrm{F}}(s)$ are themselves polynomials, it is reasonable to simplify the search for admissible parameters $\boldsymbol{q}_{\mathrm{F}}$ as much as possible. By [65], we have $\Delta_n = a_n \cdot \Delta_{n-1}$. Hence, the condition $\Delta_n > 0$ can be replaced by $a_n > 0$, since $\Delta_{n-1} > 0$ is itself checked. However, we have $a_n = 1$ in our case, and therefore, $a_n > 0$ can be omitted because it is trivially fulfilled. Furthermore, there exists an extension to the Hurwitz criterion, i.e. the criterion by Liénard and Chipart which further simplifies the analysis, especially in view of the present polynomial coefficients. It reads

Theorem 11 ([13, Th. 1]). *Necessary and sufficient conditions for $P_{\mathrm{F}}(s)$ to be Hurwitz are that*

$$a_{n-1} > 0, \quad a_{n-3} > 0, \quad a_{n-5} > 0, \quad \cdots, \quad a_0 > 0 \tag{3.35a}$$

$$\Delta_1 > 0, \quad \Delta_3 > 0, \quad \Delta_5 > 0, \quad \cdots, \quad \Delta_\alpha > 0 \tag{3.35b}$$

with

$$\alpha = n - \begin{cases} 3 & n \text{ even} \\ 2 & n \text{ odd} \end{cases}.$$

This criterion simplifies the check for stability of $\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})$ because it simplifies the structure of the polynomial inequalities that need to be evaluated to this end. The complexity of about a half of the inequalities $\Delta_k > 0$ is now reduced because the coefficients a_k of the characteristic polynomial are considered directly via Eq. (3.35a). These inequalities are easier to solve because their polynomial degree is, in general, lower.

The solution set of the parametric inequalities in Eq. (3.35) can, e.g., be approximated by means of a bisection algorithm. In Section A.2, an algorithm using Bernstein polynomials is briefly described which can be used for this task. In the bisection algorithm, a finite parameter set $\mathcal{Q}_{\mathrm{F}}^0$ is checked for including parameters $\boldsymbol{q}_{\mathrm{F}}$ rendering $\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})$ stable. Hereby, we loose globality of the results. However, a parameter set of interest $\mathcal{Q}_{\mathrm{F}}^0$ can be estimated by considering appropriate controller parameters that produce feasible input signals for system states within the expected operating range.

3.6 Example

In this section, we will show the steps for designing structured controllers, as we have covered it in this chapter. We will demonstrate these steps for the linearized version of a continuous-time *four*-tank system. I.e., we consider an extension of the system whose dynamics is described by Eq. (2.24). The plant dynamics is given by the state space model

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.36a}$$

with the matrices $\mathbf{A}$ and $\mathbf{B}$ defined by

$$\mathbf{A} = \begin{bmatrix} -\frac{q_1}{A_1}t_{12} & \frac{q_1}{A_1}t_{12} & 0 & 0 \\ \frac{q_1}{A_2}t_{12} & -\frac{1}{A_2}\left(q_1 t_{12} + q_2 t_{23}\right) & \frac{q_2}{A_2}t_{23} & 0 \\ 0 & \frac{q_2}{A_3}t_{23} & -\frac{1}{A_3}\left(q_2 t_{23} + q_3 t_{34}\right) & \frac{q_3}{A_3}t_{34} \\ 0 & 0 & \frac{q_3}{A_4}t_{34} & -\frac{1}{A_4}\left(q_3 t_{34} + q_4 t_4\right) \end{bmatrix} \tag{3.36b}$$

$$\mathbf{B} = \begin{bmatrix} \frac{1}{A_1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{A_3} & 0 \\ 0 & 0 & \frac{1}{A_4} \end{bmatrix}, \tag{3.36c}$$

where we have seamlessly extended the nomenclature from Section 2.2.5. Adding one state to the plant model allows for more expressive examples. As for Eq. (3.2), no state or input constraints are considered, i.e. $\boldsymbol{x}(t) \in \mathcal{X} = \mathbb{R}^n$, $\boldsymbol{u}(t) \in \mathcal{U} = \mathbb{R}^m$ with $n = 4$ and $m = 3$. The reference outputs have not been defined yet as they depend on the control task that should be accomplished. Throughout the example, we will consider the restriction that the state x_3 cannot be measured, i.e., we use the measurement output

$$\boldsymbol{y}_{\mathrm{m}} := \mathbf{C}_{\mathrm{m}}\boldsymbol{x} := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{x}. \tag{3.37}$$

3.6.1 Synchronizing Control Law

Let us design a control law that allows for a synchronized operation of the tanks. To this end, we need to choose a configuration for the reference outputs, and a corresponding structure matrix $\mathbf{G}_{\mathrm{w}}^*$. Defining

$$\boldsymbol{y} = \mathbf{C}\boldsymbol{x} + \mathbf{D}\boldsymbol{u} := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \boldsymbol{x} + \mathbf{0} \cdot \boldsymbol{u} \tag{3.38}$$

and

$$\mathbf{G}_{\mathrm{w1}}^* := \begin{bmatrix} * & 0 & 0 \\ 0 & * & 0 \\ 0 & 0 & * \end{bmatrix} \tag{3.39}$$

demands for a flexible synchronization of the tanks one, three, and four, while being able to control all reference outputs in a fully decoupled way. I.e., the deviation $y_2 = x_1 - x_3$ between the first and third tank should be adjustable by the second reference input w_2 while being independent from all other reference inputs. The same holds analogously for the deviation $y_3 = x_1 - x_4$ between the first and the fourth tank, as well as $y_1 = x_1$. Computing the controller and prefilter constraints (3.5) by applying the procedure described in Section 3.3.1 does not lead to a feasible set of constraints using the measurement output (3.37). The maximal control invariant subspaces are

$$\mathcal{V}_1^* = \mathrm{im}\left(\begin{bmatrix} 1 & 1 \\ 1 & -3 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}\right), \qquad \mathcal{V}_2^* = \mathrm{im}\left(\begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}\right), \qquad \mathcal{V}_3^* = \mathrm{im}\left(\begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}\right), \tag{3.40}$$

when choosing the system parameters that have been used in Section 2.2.5. The new system parameters have, furthermore, be chosen as $q_4 = 1\mathrm{cm}^2$, $A_4 = 100\mathrm{cm}^2$, and $\tilde{x}_{4\mathrm{OP}} = 24\mathrm{cm}$. The subspace $\mathcal{V}_2^*$ is not conditioned invariant w.r.t. the measurement $\boldsymbol{y}_\mathrm{m}$. This has been verified applying Condition (4.1.2) from [14] which is listed in Section A.3. This causes that there does not exist a controller matrix $\mathbf{F}_{\boldsymbol{y}}$ simultaneously rendering all of the above $\mathcal{V}_j^*$ with $j \in \mathcal{I}_{1,3}$ control invariant. Interestingly, using a full state feedback, structure $\mathbf{G}_{\mathrm{w}1}^*$ is realizable, which can also be checked by verifying

$$\det(\mathbf{D}^*) = \det\left(\begin{bmatrix} 0.01 & 0 & 0 \\ 0.01 & -0.01 & 0 \\ 0.01 & 0 & -0.01 \end{bmatrix}\right) \neq 0.$$

Creating a transfer structure feasible to an output feedback is connected to changing at least the second column of $\mathbf{G}_{\mathrm{w}1}^*$ because the corresponding subspace $\mathcal{V}_2^*$ was not conditioned invariant. It turns out that setting $g_{\mathrm{w}32}^* = g_{\mathrm{w}23}^* = *$ in $\mathbf{G}_{\mathrm{w}2}^*$ suffices for establishing solvability of the problem. Applying these relaxations, we define

$$\mathbf{G}_{\mathrm{w}2}^* := \begin{bmatrix} * & 0 & 0 \\ 0 & * & * \\ 0 & * & * \end{bmatrix}. \tag{3.41}$$

Using this structure, we allow for an interaction between the two synchronization tasks. Hence, if a change of an offset $y_2 = x_1 - x_3$, or $y_3 = x_3 - x_4$, is commanded by w_2 or w_3, both synchronization outputs will be affected. However, synchronization is still fully decoupled from the tank level x_1.

Setting up the control law constraints for $\mathbf{G}_{\mathrm{w}2}^*$, we again obtain $\mathcal{V}_1^*$ from Eq. (3.40) as control invariant subspace. This is trivial because we did not change the first column of $\mathbf{G}_{\mathrm{w}}^*$. The other two subspaces are now identical because the respective columns of the transfer matrix are equal. They now read

$$\mathcal{V}_2^* = \mathcal{V}_3^* = \mathrm{im}\left(\begin{bmatrix} \mathbf{0}_{1\times 3} \\ \mathbf{I}_3 \end{bmatrix}\right). \tag{3.42}$$

For rendering the above subspaces invariant, the controller matrix $\mathbf{F}_y$ must satisfy the constraints

$$\begin{bmatrix} 1 & 0 & -1 & & 0 & -1 \\ 0 & 1 & -1 & \mathbf{0}_{2\times 4} & 1 & -1 \\ \mathbf{0}_{4\times 3} & & & \mathbf{I}_4 & \mathbf{0}_{4\times 2} & \end{bmatrix} \boldsymbol{f}_y = \begin{bmatrix} 4.52 \\ 4.52 \\ -9.04 \\ -9.04 \\ \mathbf{0}_{2\times 1} \end{bmatrix}, \tag{3.43}$$

where $\boldsymbol{f}_y = \mathrm{vec}(\mathbf{F}_y)$. For avoiding any signal leakages from w_1 to $(\mathcal{V}_1^*)^\perp$ and from w_2, w_3 to $(\mathcal{V}_2^*)^\perp = (\mathcal{V}_3^*)^\perp$, respectively, the prefilter coefficients must satisfy

$$\begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \boldsymbol{\ell} = \mathbf{0}_{4\times 1}, \tag{3.44}$$

where $\boldsymbol{\ell} = \mathrm{vec}(\mathbf{L})$. Both constraint sets are solvable, i.e., structure (3.41) is realized by choosing any pair $\mathbf{F}_y$, $\mathbf{L}$ fulfilling the above constraints. The corresponding matrix structures are given by

$$\mathbf{F}_x = \mathbf{F}_y \mathbf{C}_\mathrm{m} = \begin{bmatrix} f_{31} + f_{34} + 4.52 & -9.04 & 0 & 0 \\ f_{31} - f_{24} + f_{34} + 4.52 & -9.04 & 0 & f_{24} \\ f_{31} & 0 & 0 & f_{34} \end{bmatrix} \tag{3.45a}$$

$$\mathbf{L} = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{11} & l_{22} & l_{23} \\ l_{11} & l_{23} & l_{33} \end{bmatrix}. \tag{3.45b}$$

We can see that realizing $\mathbf{G}_{\mathrm{w}2}^*$ occupies quite some degrees of freedom in the controller. Only the parameters f_{31}, f_{24} and f_{34} are freely assignable. For the prefilter, the direction of the first column is fixed and can only be adapted in length through the parameter l_{11}. The lower right parameters are freely assignable.

It can further be deduced that the measurement of x_4 can also be dropped while still being able to realize $\mathbf{G}_{\mathrm{w}2}^*$ as the fourth column of $\mathbf{F}_x$ can be set to zero. However, large attention must be paid regarding the stabilizability of the plant under the structured and reduced state feedback. Having only one free controller parameter left, i.e. f_{31}, will likely result in the creation of fixed eigenvalues which may well be unstable. Methods for analyzing this issue have been discussed in Section 3.5.

3.6.2 Identifying Structural Robustness

Let us continue the example and analyze the structure $\mathbf{G}_{\mathrm{w}2}^*$ from Eq. (3.41) in more detail. We want to focus on the structure's robustness properties w.r.t. parameter changes of the plant, see Section 3.3.2. The system dynamics Eq. (3.36) is given in terms of a set of possibly uncertain parameters $\boldsymbol{\theta} = [q_1 \;\; q_2 \;\; q_3 \;\; q_4 \;\; \tilde{x}_{1\mathrm{OP}} \;\; \tilde{x}_{2\mathrm{OP}} \;\; \tilde{x}_{3\mathrm{OP}} \;\; \tilde{x}_{4\mathrm{OP}} \;\; A_1 \;\; A_2 \;\; A_3 \;\; A_4]^\top$. We are interested in the question whether we can use a single controller $\mathbf{F}_y$ that maintains the structure $\mathbf{G}_{\mathrm{w}2}^*$

even if the plant parameters $\boldsymbol{\theta}$ vary. To this end, the control invariant subspaces $\mathcal{V}_j^*$ and the conditions $\tilde{\mathbf{A}}_{\mathrm{F}21}^{\,j} = \mathbf{0}$ with $j \in \mathcal{I}_{1,3}$ from Eq. (3.17) are computed symbolically. This is more involved than computing it purely numerically, but doable for our scenario. The easiest way for symbolically obtaining $\mathcal{V}_j^*$ is using Algorithm 4.1.1 and Eq. (4.1.8) in Basile and Marro [14]. This way, symbolic matrix inversions can be avoided, yielding a feasible computation procedure.

It turns out that all subspaces $\mathcal{V}_j^*$, i.e. $\mathcal{V}_1^*$ from Eq. (3.40) and $\mathcal{V}_2^*$, $\mathcal{V}_3^*$ from Eq. (3.42), that need to be made control invariant using $\mathbf{F}_{\boldsymbol{y}}$ are fully parameter independent. For $j = 1$, Eq. (3.17) contains four conditions

$$\begin{bmatrix} \frac{0.29}{A_1} & -\frac{0.39}{A_3} & \frac{0.11}{A_4} & \frac{0.29}{A_1} & -\frac{0.39}{A_3} & \frac{0.11}{A_4} \\ \frac{0.29}{A_1} & \frac{0.11}{A_3} & -\frac{0.39}{A_4} & \frac{0.29}{A_1} & \frac{0.11}{A_3} & -\frac{0.39}{A_4} \\ \frac{0.17}{A_1} & -\frac{0.23}{A_3} & \frac{0.061}{A_4} & -\frac{0.5}{A_1} & \frac{0.68}{A_3} & -\frac{0.18}{A_4} \\ \frac{0.17}{A_1} & \frac{0.061}{A_3} & -\frac{0.23}{A_4} & -\frac{0.5}{A_1} & -\frac{0.18}{A_3} & \frac{0.68}{A_4} \end{bmatrix} \quad \cdots$$

$$\cdots \quad \begin{bmatrix} \frac{0.29}{A_1} & -\frac{0.39}{A_3} & \frac{0.11}{A_4} \\ \frac{0.29}{A_1} & \frac{0.11}{A_3} & -\frac{0.39}{A_4} \\ \frac{0.17}{A_1} & -\frac{0.23}{A_3} & \frac{0.061}{A_4} \\ \frac{0.17}{A_1} & \frac{0.061}{A_3} & -\frac{0.23}{A_4} \end{bmatrix} \boldsymbol{f}_{\boldsymbol{y}} = \begin{bmatrix} \frac{0.11 q_4 t_4}{A_4} \\ -\frac{0.39 q_4 t_4}{A_4} \\ \frac{0.67 q_1 t_{12}}{A_1} + \frac{0.061 q_4 t_4}{A_4} - \frac{0.91 q_2 t_{23}}{A_3} \\ \frac{0.67 q_1 t_{12}}{A_1} - \frac{0.23 q_4 t_4}{A_4} + \frac{0.24 q_2 t_{23}}{A_3} \end{bmatrix}. \tag{3.46}$$

As the controller parameters f_{ij} in $\boldsymbol{f}_{\boldsymbol{y}}$ are constants, we can generally deduce that the parameters in $\boldsymbol{\theta}$ may only vary in a way such that a fixed set $\boldsymbol{f}_{\boldsymbol{y}}$ of controller parameters still solves the altered set of conditions. The above equation is fully independent of, e.g., A_2. Hence, it does not need to be known for rendering $\mathcal{V}_1^*$ control invariant and may therefore be uncertain. However, if, for example, q_1 is uncertain and we have a fixed $\boldsymbol{f}_{\boldsymbol{y}}$, it is likely that the altered constraints are violated. Therefore, all parameters occurring in such a way need to be known precisely for maintaining $\mathcal{V}_1^*$ control invariant, i.e., this subspace is not robust w.r.t. such parameters.

No new information results from setting up $\tilde{\mathbf{A}}_{\mathrm{F}21}^{\,j} = \mathbf{0}$ for $j = 2, 3$ and the prefilter constraints (3.19). Hence, we omit displaying them here. We can therefore expect that $\mathbf{G}_{\mathrm{w}2}^*$ can be realized by a constant controller $\mathbf{F}_{\boldsymbol{y}}$ even if the areas q_3 and A_2 are fully unknown. Also, the operating point $\tilde{x}_{1\mathrm{OP}}$ can be chosen arbitrarily, as long as the other tank levels maintain the nominal deviation, i.e., the terms t_{ij} and t_i must stay constant. The parameters mentioned only influence the closed-loop dynamic behavior, i.e. the closed-loop eigenvalues, but not the eigenstructure.

The knowledge obtained above can be used if parametric uncertainties should explicitly be taken into account in the structured control design. The conditions (3.5) resulting from the multiple-plant approach discussed in Section 3.3.2 will be infeasible if the underlying values manipulate (3.46) and the parameter set Θ is sufficiently large. Then, control design must be downgraded to an approximate procedure, as we have introduced it in Section 3.4.

3.6.3 Approximation of the Synchronizing Structure

Such an approximation approach can also be considered for setting up the structure $\mathbf{G}_{\mathrm{w}1}^*$ that we first sought to realize but which was infeasible. We can try to take an influence on the coupling

effects between the two synchronization tasks with the aim of minimizing them. Either, we simply use a minimum norm solution for all constraints arising in view of $\mathbf{G}^*_{\mathrm{w}1}$, or we prioritize the creation of the substructure $\mathbf{G}^*_{\mathrm{w}2}$ over realizing $g^*_{\mathrm{w}32} = g^*_{\mathrm{w}23} = 0$. Let us demonstrate the latter.

First, note that the prefilter constraints for $\mathbf{G}^*_{\mathrm{w}1}$ are solvable, i.e., no approximation is needed here. Hence, the key task for defining our priorities is separating Eq. (3.5a) such that we obtain conditions of the form (3.26). As we want to prioritize the substructure $\mathbf{G}^*_{\mathrm{w}2}$, its corresponding constraints (3.43) define Eq. (3.26b) for $\mathbf{F}_{\boldsymbol{y}}$. The approximated set of constraints corresponds to setting up

$$\mathbf{G}^*_{\mathrm{w}1\approx} := \begin{bmatrix} * & * & * \\ * & * & 0 \\ * & 0 & * \end{bmatrix},$$

which yields the infeasible controller constraint set (3.26a) as

$$\begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix} \boldsymbol{f}_{\boldsymbol{y}} = \begin{bmatrix} 9.04 \\ 1 \\ 8.27 \\ 0 \\ -12.79 \\ 12.79 \end{bmatrix}. \tag{3.47}$$

We then first solve (3.26b) given by Eq. (3.43), and use the remaining controller parameters to approximate (3.47), i.e. computing Eqs. (3.28) and (3.29). Hereby, we optimally suppress the coupling effects between w_2 and y_3, and w_3 and y_2, respectively, where *optimal* is defined in the sense of the objectives (3.23).

Solving (3.26b) yields (3.27) as

$$\boldsymbol{f}_{\boldsymbol{y}} = \begin{bmatrix} 1.13 \\ 0.57 \\ -1.70 \\ -9.04 \\ -9.04 \\ \mathbf{0}_{2\times 1} \\ 0.57 \\ -1.70 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 1 \\ 1 & -1 & 1 \\ 1 & 0 & 0 \\ & \mathbf{0}_{4\times 3} & \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_{\mathrm{F}1} \\ q_{\mathrm{F}2} \\ q_{\mathrm{F}3} \end{bmatrix}$$

with the remaining degrees of freedom collected in $\boldsymbol{q}_{\mathrm{F}} = \begin{bmatrix} q_{\mathrm{F}1} & q_{\mathrm{F}2} & q_{\mathrm{F}3} \end{bmatrix}^\top$. Inserting this expression into (3.47) encoding the low priority parts of the closed-loop structure $\mathbf{G}^*_{\mathrm{w}1}$, we obtain

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix} \boldsymbol{q}_{\mathrm{F}} = \begin{bmatrix} 0 \\ 1 \\ 12.79 \\ 0 \\ -13.35 \\ 13.35 \end{bmatrix}.$$

Due to the second and third row, the resulting system of equations is not solvable. Therefore, the normal equations (3.29) are used to define the structure of the controller matrix. They read

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{q}_{\mathrm{F}} = \begin{bmatrix} 0 \\ -26.7 \\ 0 \end{bmatrix}$$

forcing $q_{\mathrm{F2}} = -13.35$ and, hence, resulting in the applied structured controller

$$\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}}) = \mathbf{F}_{\boldsymbol{y}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{C}_{\mathrm{m}} = \begin{bmatrix} q_{\mathrm{F1}} + q_{\mathrm{F3}} + 1.13 & -9.04 & 0 & 0 \\ q_{\mathrm{F1}} + q_{\mathrm{F3}} + 13.92 & -9.04 & 0 & -12.79 \\ q_{\mathrm{F1}} - 1.70 & 0 & 0 & q_{\mathrm{F3}} - 1.70 \end{bmatrix}. \tag{3.48}$$

As the original prefilter constraints are solvable, $\mathbf{L}$ has the form (3.45b).

Performing the above steps, we obtain a strictly prioritized structural control design because we choose $\mathbf{F}_{\boldsymbol{y}}$ such that it *exactly* establishes $\mathbf{G}^*_{\mathrm{w}2}$. The remaining parameters are chosen in order to fulfill (3.47) as good as possible. This is in contrast to computing a softly prioritized structure by using the weighted approach described in Remark 12.2. There, none of the constraints will exactly be fulfilled. The solution is rather designed to approximate the entirety of the constraints.

Applying the controller (3.48) results in the placement of two eigenvalues $\lambda_1 = -0.1809$ and $\lambda_2 = -0.2185$ which can easily be verified by the procedure discussed in Section 3.5.2.1, i.e. choosing random numbers for the degrees of freedom $\boldsymbol{q}_{\mathrm{F}}$. By this random number test, we also see that the plant is stabilizable under the applied constraints because all randomly generated eigenvalues that have been tested have negative real parts.

3.7 Summary

Let us sum up the content of this chapter. The main intention is to study the properties of linear time-invariant systems w.r.t. the design of control laws establishing a given structure $\mathbf{G}^*_{\mathrm{w}}$ of the closed-loop transfer matrix $\mathbf{G}_{\mathrm{w}}(s)$. The focus lies on structures $\mathbf{G}^*_{\mathrm{w}}$ that are not necessarily standard. I.e., the results are not limited to, e.g., diagonal or triangular decoupling structures. Mixtures of such structures or fully arbitrary structures are covered by the tools presented in this chapter. The control law itself may also be pre-structured, e.g., for designing decentralized control laws or dynamic controllers, such as PI controllers.

Following Chapter 2, we are interested in statements about the robustness of control structures w.r.t. parameter uncertainties of the plant, i.e., statements whether a single and constant controller can keep up a closed-loop structure even under parameter changes of the plant. Such answers can be given while following a constructive method for establishing the questionable control structure $\mathbf{G}^*_{\mathrm{w}}$. Therefore, large parts of the chapter are devoted to developing methods for extracting the implications of the control loop structure $\mathbf{G}^*_{\mathrm{w}}$ on the structure of the control law (3.4).

As an introductory example, we compute in Section 3.1 the mentioned implications for the well-known diagonal decoupling control design by Falb and Wolovich [49]. We fully separate the

dynamics design from the structural design and obtain linear equality constraints in the controller parameters. Every controller satisfying these constraints establishes the desired diagonal decoupling structure $\mathbf{G}_{\mathrm{w}}^{*}$. The respective results are summarized in Proposition 1.

The main disadvantage of the above approach is that it is not extendable to arbitrary structures $\mathbf{G}_{\mathrm{w}}^{*}$. To this end, we switch to a fully geometric approach to our design problem. The main concepts, i.e. controlled and conditioned invariance, are introduced in Section 3.2.1. In the following Section 3.2.2, we establish the link between these invariance concepts and our original aim of designing a structured control loop. This includes the geometric conditions that the control law (3.4) must satisfy in order to establish $\mathbf{G}_{\mathrm{w}}^{*}$, see Proposition 2. The discussion is conducted for fully arbitrary control loop structures. Hence, the results incorporate the extension that could not be found in Section 3.1.

Section 3.3.1 focuses on the translation of the derived geometric constraints to algebraic constraints that can be computed numerically. The geometric conditions are mainly given in terms of control invariant subspaces. The procedure for transferring them to algebraic constraints is based on *adapting* the system description to these subspaces by an orthonormal state transformation. The linear equality constraints obtained can be used for a robustness analysis of the control structure. It can be investigated which controller parameters are numerically fixed by establishing $\mathbf{G}_{\mathrm{w}}^{*}$. Following the design steps symbolically, we can identify plant parameters that need to be known precisely for realizing $\mathbf{G}_{\mathrm{w}}^{*}$. At the same time, the parameters that may vary arbitrarily while not affecting the closed-loop structure are identified. Furthermore, in view of the controller structure, it is possible to derive statements about the importance of signal paths. In Section 3.3.2, we briefly discuss how the design procedure for single-plant scenarios can be extended to families of plants. Such families arise if parameter uncertainties are explicitly taken into account.

When choosing arbitrary closed-loop structures $\mathbf{G}_{\mathrm{w}}^{*}$, it can easily happen that no control law exists that satisfies the conditions from Proposition 2. This situation will also occur if the parameter configurations used for the multiple-plant design are not compatible, i.e., they do not admit a single control law establishing $\mathbf{G}_{\mathrm{w}}^{*}$ for all configurations. In Section 3.4, we propose a collection of measures for designing the control law such that it approximately realizes $\mathbf{G}_{\mathrm{w}}^{*}$. They are mainly based on choosing minimum norm solutions of the constraint set (3.5).

Furthermore, we propose ways to prioritize elements of $\mathbf{G}_{\mathrm{w}}^{*}$. A strict prioritization can be obtained by separating the equality constraints into a set of constraint systems. These can then be solved sequentially ensuring exact fulfillment of the highly prioritized constraints. Soft prioritization can be realized by solving a weighted quadratic optimization problem leading to a description of the control law in terms of weighted normal equations, see Remark 12.2.

Infeasibility of the construction steps proposed in Section 3.3.1 does not imply that the conditions from Proposition 2 cannot be met, i.e., the design steps depict sufficient but not necessary solvability conditions of the problem. This is because, in the controller design, we work with *maximal* control invariant subspaces. These are advantageous for our purposes because they can easily be obtained numerically. In Section 3.4, we review the subspace search that we *would* have to perform in order to obtain necessary solvability conditions. This subspace search is mathematically

complex since it involves a search over orthogonal matrices, see Eq. (3.22). We use this situation to justify application of the above mentioned approximate design strategies based on the objectives defined in Eq. (3.23).

The last part of the chapter covers the question of stabilizability of the plant under use of structured controllers. We distinguish between structures that fulfill invariance conditions and structures of unknown or mixed origin. The analyses cover the identification of eigenvalues that are fixed by establishing the structure $\mathbf{G}_{\mathrm{w}}^{*}$ and the ability to use the remaining controller parameters for stabilizing the plant, i.e. placing the controllable eigenvalues in the left complex half-plane. In Section 3.5.1, we present a method based on a Kalman decomposition of the plant exposing the assignable and fixed eigenvalues under the invariance constraints. Section 3.5.2 covers constraint systems of unknown origin and includes a stochastic identification method for fixed eigenvalues. Finding a set of free controller parameters stabilizing the plant is based on the LC criterion (Theorem 11) and the bisection algorithm from Section A.2.

4 Parameterization Methods for Structured Controllers

In the prior chapter, we have set up structures for the controller and prefilter matrices that establish a given closed-loop transfer matrix $\mathbf{G}_{\mathrm{w}}^{*}$, or at least approximate it. Depending on the number and the nature of the constraints in Eq. (3.5), there are parameters left that can be chosen freely, e.g., to stabilize the plant. They are given by the vectors $\boldsymbol{q}_{\mathrm{F}} \in \mathbb{R}^{n_{\mathrm{F}}}$ and $\boldsymbol{q}_{\mathrm{L}} \in \mathbb{R}^{n_{\mathrm{L}}}$ that appear when solving Eq. (3.5) as

$$\boldsymbol{f}_{\boldsymbol{y}} = (\mathbf{Z}_{\mathrm{F}})^{+}\, \boldsymbol{z}_{\mathrm{F}} + (\mathbf{Z}_{\mathrm{F}})^{\perp}\, \boldsymbol{q}_{\mathrm{F}} \tag{4.1a}$$

$$\boldsymbol{\ell} = (\mathbf{Z}_{\mathrm{L}})^{+}\, \boldsymbol{z}_{\mathrm{L}} + (\mathbf{Z}_{\mathrm{L}})^{\perp}\, \boldsymbol{q}_{\mathrm{L}} \tag{4.1b}$$

with, again, the pseudo inverse $(\cdot)^{+}$ and the kernel representation $(\cdot)^{\perp}$. How to choose them is subject of this chapter. Hereby, the main part is devoted to selecting $\boldsymbol{q}_{\mathrm{F}}$ such that we obtain a stable control loop. A strategy for completing the prefilter design is discussed afterwards in Section 4.3.

In Section 3.5, we have already briefly discussed some methods that are worth considering for designing $\boldsymbol{q}_{\mathrm{F}}$. Among them, pole assignment is a tractable technique that is well extendable to the multiple-plant case, as we will outline in Section 4.1.2. With this extension, it can handle explicit uncertainties of the plant model. We choose not to deepen the discussion about Lyapunov approaches as given by Eq. (3.31) because this is inherently connected with solving parameter-dependent linear matrix inequalities (LMIs). This is a difficult task as can be reviewed, e.g. in Apkarian and Tuan [10] and Peaucelle and Arzelier [127]. The difficulty results mainly from the situation that many available results on parametric LMIs require the parameters to have specifically structured forms, which are not present in our setting.

In this chapter, we introduce a novel approach that allows for parameterizing the questionable degrees of freedom by means of linear quadratic regulator design (LQR). In this view, we a priori deal with a constrained optimal control problem because the resulting controller needs to meet the constraints (3.5a). By interpreting this design task as an inverse optimal control problem, we are able to construct suitable ingredients, i.e. weighting matrices, of an unconstrained LQR problem solving the originally constrained optimization. Concluding, we provide two conceptually different approaches suitable for finalizing the structured controllers developed in Chapter 3.

4.1 Pole Assignment

The control design scheme in Chapter 3 has originally been designed for the use with pole region assignment, as outlined in [148]. Since pole region assignment is an optimization based method,

where the controller (and prefilter) coefficients are decision variables, we have initially chosen the form (3.5) for representing the control law structure. It is, therefore, relatively straight-forward to set up the corresponding parameterization problem. Before discussing this procedure, we briefly comment on the method of matching coefficients that is one of the most classical ways to choose the parameters of some state or output feedback law.

4.1.1 Matching Coefficients

The task we have in this context is shaping the spectrum $\sigma(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{y}}(\boldsymbol{q}_\mathrm{F})\mathbf{C}_\mathrm{m})$ by suitably choosing the parameters $\boldsymbol{q}_\mathrm{F} \in \mathbb{R}^{n_\mathrm{F}}$. In classical control design, it is common to compute the respective characteristic polynomial (3.30), i.e.

$$P_\mathrm{F}(s) = \det(s\mathbf{I}_n - \mathbf{A} - \mathbf{B}\mathbf{F}_{\boldsymbol{y}}(\boldsymbol{q}_\mathrm{F})\mathbf{C}_\mathrm{m}) = s^n + a_{n-1}(\boldsymbol{q}_\mathrm{F})s^{n-1} + \cdots + a_1(\boldsymbol{q}_\mathrm{F})s + a_0(\boldsymbol{q}_\mathrm{F}),$$

and choose the controller parameters such that $P_\mathrm{F}(s)$ attains a desired form. If the plant is controllable, the measurement output satisfies $\operatorname{rank}(\mathbf{C}_\mathrm{m}) = n$, and no structural constraints are imposed on $\mathbf{F}_{\boldsymbol{y}}$, then $n_\mathrm{F} = mn$ degrees of freedom are available in $\mathbf{F}_{\boldsymbol{y}}$, or $\boldsymbol{q}_\mathrm{F}$, respectively. This is sufficient for establishing every arbitrary eigenvalue configuration. Hence, a reference polynomial

$$P_\mathrm{F}^\mathrm{ref}(s) = s^n + a_{n-1}^\mathrm{ref}s^{n-1} + \cdots + a_1^\mathrm{ref}s + a_0^\mathrm{ref}$$

can be arbitrarily selected and it is guaranteed that the controller can be chosen such that $P_\mathrm{F}(s) = P_\mathrm{F}^\mathrm{ref}(s)$ for all $s \in \mathbb{C}$. This is achieved by choosing $\mathbf{F}_{\boldsymbol{y}}$ such that the coefficients satisfy

$$a_i^\mathrm{ref} = a_i(\boldsymbol{q}_\mathrm{F}) \qquad \forall\, i \in \mathcal{I}_{0,n-1}. \tag{4.2}$$

In general, this procedure can also be applied in the constrained setting where $n_\mathrm{F} < mn$ holds. No limitation occurs due to the complexity of $\mathbf{G}_\mathrm{w}^*$. It is, however, challenging to set up a feasible reference polynomial $P_\mathrm{F}^\mathrm{ref}(s)$. It must contain all fixed eigenvalues from σ_fix as zeros. Its remaining zeros must be attainable by some choice of $\boldsymbol{q}_\mathrm{F}$. Otherwise, no $\boldsymbol{q}_\mathrm{F}$ exists solving the set of equations (4.2). This situation is a clear disadvantage of this method in view of constrained controllers because the set of assignable eigenvalue *combinations* has to be known before selecting $P_\mathrm{F}^\mathrm{ref}(s)$. Depending on the plant, this can be involved.

A second strategy for direct eigenvalue placement has already been discussed in Section 3.5.1. It can easily be applied but is restricted to the design of state feedback laws and structures that only feature one control invariant subspace, i.e., Eq. (3.33) must hold. Using decomposition (3.32), the assignable eigenvalues are determined by the sub-controllers $\tilde{\mathbf{F}}_{\boldsymbol{x},1}$ and $\tilde{\mathbf{F}}_{\boldsymbol{x},3}$ [160, p. 87]. The system structure used here is advantageous since the freely assignable eigenvalues can be assigned by means of unstructured control design, which is well-known.

4.1.2 Pole Region Assignment

As mentioned earlier, transferring closed-loop structures $\mathbf{G}_\mathrm{w}^*$ into linear equality constraints of the form (3.5) has originally been motivated by the idea to combine structured control design with

pole region assignment. Pole region assignment is formulated as a static optimization procedure in the controller parameters f_{ij} of $\mathbf{F}_{\boldsymbol{y}}$ with $i \in \mathcal{I}_{1,m}$ and $j \in \mathcal{I}_{1,p_\mathrm{m}}$ in order to influence the spectrum $\sigma_\mathrm{F} := \sigma(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{y}}\mathbf{C}_\mathrm{m})$ of the closed-loop dynamics according to a given objective function $J(\mathbf{F}_{\boldsymbol{y}}) \in \mathbb{R}_{\geq 0}$. Therefore, the design procedure can be combined with our structured control design well since, here, the structural requirements imposed by $\mathbf{G}_\mathrm{w}^*$ are directly transformed into constraints on these parameters by Eq. (3.5a). These can be included in the optimization as

$$\min_{\mathbf{F}_{\boldsymbol{y}}} \quad J(\mathbf{F}_{\boldsymbol{y}}) \tag{4.3a}$$

$$\text{s.t.} \quad \mathbf{Z}_\mathrm{F}\boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{z}_\mathrm{F}. \tag{4.3b}$$

The advantage of an optimization based formulation of the eigenvalue assignment problem is that the set of assignable eigenvalues does not need to be known exactly beforehand. Shifting the closed-loop spectrum to a desired region $\Gamma \subset \mathbb{C}$ in the complex plane can either be encoded in the objective function J as in (4.3), or included in additional constraints via

$$\min_{\mathbf{F}_{\boldsymbol{y}}} \quad \tilde{J}(\mathbf{F}_{\boldsymbol{y}}) \tag{4.4a}$$

$$\text{s.t.} \quad \mathbf{Z}_\mathrm{F}\boldsymbol{f}_{\boldsymbol{y}} = \boldsymbol{z}_\mathrm{F} \tag{4.4b}$$

$$\sigma_\mathrm{F}(\mathbf{F}_{\boldsymbol{y}}) \subset \Gamma, \tag{4.4c}$$

where the objective $\tilde{J}$ can be used for formulating secondary requirements, or be omitted by setting $\tilde{J} = 0$.

The approach via the optimization problem (4.3) corresponds to a *soft* assignment since the resulting spectrum is not guaranteed to satisfy $\sigma_\mathrm{F} \subset \Gamma$. However, as long as the side constraints (3.5a) are feasible, the optimization is as well feasible. This approach is especially advantageous if it is unknown whether the chosen region Γ depicts a feasible set for the closed-loop eigenvalues under the structural constraints. In contrast, the way of encoding Γ into additional side constraints as in (4.4) guarantees $\sigma_\mathrm{F} \subset \Gamma$ if the optimization problem is feasible. In turn, more detailed knowledge concerning the possibilities on eigenvalues placement are needed.

If parameter uncertainties are existent in the plant model, i.e., we have a description of the form (3.21), a finite set Θ of realizations of the parameter vectors $\boldsymbol{\theta} \in \mathbb{R}^\theta$ can be included into the regional assignment procedure. This does not increase the number of decision variables as we still seek a single controller $\mathbf{F}_{\boldsymbol{y}}$ simultaneously shifting all spectrums $\sigma_\mathrm{F}(\boldsymbol{\theta}, \mathbf{F}_{\boldsymbol{y}})$ inside Γ. Oftentimes, the parameter uncertainty is a priori assumed to be restricted to some bounded set $\overline{\Theta} \subset \mathbb{R}^\theta$ with infinitely many elements. A truly robust control design must give stability guarantees for all $\boldsymbol{\theta} \in \overline{\Theta}$, which is not inherently the case if only finitely many parameters are sampled in $\Theta \subset \overline{\Theta}$ and included into the design procedures (4.3) or (4.4). Remedy to this situation is provided by the Bauer-Fike Theorem [33]. It can be used to estimate the perturbation of $\sigma_\mathrm{F}(\boldsymbol{\theta}, \mathbf{F}_{\boldsymbol{y}})$ in view of a perturbation of the parameter vector $\boldsymbol{\theta}$. Applying the theorem, statements about the intersampled systems can be made such that stability is verified for all $\boldsymbol{\theta} \in \overline{\Theta}$.

More detailed information regarding pole region assignment is presented by, e.g., Ackermann [2] and Vogt et al. [165]. This includes the construction of the objectives J and $\tilde{J}$, or the constraint $\sigma_{\mathrm{F}}(\mathbf{F}_{\boldsymbol{y}}) \subset \Gamma$, respectively, as well as numerical solution methods.

4.2 Linear Quadratic Control

A second standard tool for designing state feedback controllers for linear time-invariant plants of the form (3.2) is LQR design. The design task is formulated by

$$\min_{\boldsymbol{u}} \quad \int_0^\infty \boldsymbol{x}^\top(t)\mathbf{Q}\boldsymbol{x}(t) + \boldsymbol{u}^\top(t)\mathbf{R}\boldsymbol{u}(t)\mathrm{d}t \tag{4.5a}$$

$$\text{s.t.} \quad \dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{4.5b}$$

with positive definite matrices $\mathbf{Q} \succ 0$ and $\mathbf{R} \succ 0$ of appropriate dimensions. For every quadratic objective of the above form and the plant (4.5b) being stabilizable, the optimal control problem (OCP) is solved by a state feedback control law $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$. With $\mathbf{Q}$ and $\mathbf{R}$ given, the time-invariant feedback matrix can be computed by means of variational calculus, see e.g. [98]. It is uniquely defined by

$$\mathbf{F}_{\boldsymbol{x}} = -\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}, \tag{4.6}$$

where $\mathbf{P}$ is the unique positive definite solution of the algebraic Riccati equation

$$\mathbf{Q} = \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} - \mathbf{P}\mathbf{A} - \mathbf{A}^\top\mathbf{P}. \tag{4.7}$$

The derivation of Eqs. (4.6) and (4.7) is not trivial, but the final result is a significant simplification w.r.t. the original problem. It admits finding a solution by comparably simple numeric algorithms. The remaining challenge is solving Eq. (4.7) by some $\mathbf{P} \succ 0$. This problem can be transferred to a generalized eigenvalue problem, as presented by Arnold and Laub [12].

A significant advantage over pole assignment techniques is that LQR techniques inherently provide an intuition to control design. While for pole assignment, we explicitly have to define the closed-loop spectrum σ_{F} prior to the design step, we can encode a desired system behavior in the objective function in LQR design. In many cases, especially if the system states and inputs resemble physical quantities and n and m are large numbers, this is supported by a good interpretability. Hence, the abstract choice of eigenvalues can be substituted by a more heuristic and natural access to control design.

The key challenge is that LQR is an optimal control formulation and naturally provides the optimal input function $\boldsymbol{u}\colon [0;\infty) \to \mathbb{R}^m$ as opposed to the optimal state feedback $\mathbf{F}_{\boldsymbol{x}}$. The fact that this optimal input function can be expressed as a state feedback law $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$ is only a secondary result. That is, since the design problem is fully formulated in terms of state and input signals, it is not possible to directly influence the structure of the resulting optimal controller $\mathbf{F}_{\boldsymbol{x}}$, as we intend to do it. At this point, modal synthesis techniques are better suitable for including predefined controller structures, or control-loop structures $\mathbf{G}_{\mathrm{w}}^*$. They are inherently designed such that, if

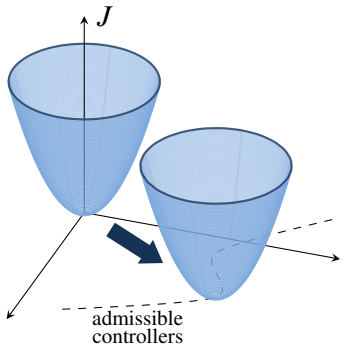


Figure 4.1: Illustration of the objective shift such that the minimizing controller automatically satisfies Eq. (4.8c). The objective function is denoted by J.

$m > 1$, they allow for a specific usage of the degrees of freedom that are not needed for pole placement. A feedback $\mathbf{F}_{\boldsymbol{x}}$ designed by LQR is, in contrast, fully parameterized by the previously chosen weights $\mathbf{Q}$ and $\mathbf{R}$.

Let us continue the above discussion in view of the optimization itself. Directly including the controller structure into (4.5) leads to a constrained dynamic optimization problem given by

$$\min_{\mathbf{F}_{\boldsymbol{x}}} \quad \int_0^\infty \boldsymbol{x}^\top(t)\mathbf{Q}\boldsymbol{x}(t) + \boldsymbol{u}^\top(t)\mathbf{R}\boldsymbol{u}(t)\mathrm{d}t \tag{4.8a}$$

$$\text{s.t.} \quad \dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t) + \mathbf{B}\boldsymbol{u}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{4.8b}$$

$$\mathbf{Z}_{\mathrm{F}\boldsymbol{x}}\,\boldsymbol{f}_{\boldsymbol{x}} = \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}} \tag{4.8c}$$

$$\boldsymbol{u}(t) = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}(t). \tag{4.8d}$$

with $\boldsymbol{f}_{\boldsymbol{x}} = \mathrm{vec}(\mathbf{F}_{\boldsymbol{x}})$.[1] It may well be possible to solve it approximately by means of numeric optimization algorithms. However, it has to be noted that the design problem turns nonlinear because the product in the constraint (4.8d) depends on two unknowns. This way, we lose the rich theory available to solve the original problem. To circumvent this, the idea to include the constraints (4.8c) into problem (4.5) is shifting the objective of the unconstrained problem such that its minimizer automatically satisfies Eq. (4.8c). This aim is graphically shown in Fig. 4.1, where the objective, named J, is drawn over a set of controllers $\mathbf{F}_{\boldsymbol{x}}$. On this set, there exists a subset of admissible controllers satisfying the constraints which is highlighted by the dashed line. The objective is shifted such that its minimum lies on the dashed line.

Summarizing, the problem that we want to solve in this section is given by

Definition 23 (LQR with linear constraints, LQRLC [146]). *For the optimal control problem* (4.5)*, find matrices* $\mathbf{Q} \succ 0$ *and* $\mathbf{R} \succ 0$*, such that the minimizer* $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$ *satisfies the constraints* (4.8c).

[1] Since the LQR design procedure returns a complete state feedback $\mathbf{F}_{\boldsymbol{x}}$ as opposed to an output feedback $\mathbf{F}_{\boldsymbol{y}}$, we assume throughout this section that the controller structure defined by Eq. (3.5a) is formulated in terms of state feedback parameters. For a derivation of Eq. (4.8c) from Eq. (3.5a), see Section A.4. Since both representations are equivalent, the degrees of freedom $\boldsymbol{q}_{\mathrm{F}}$ that occur when solving them, are the same.

4.2.1 Inverse Optimal Control

Definition 23 asks for the design of the unconstrained optimal control problem such that it naturally fulfills the imposed constraints. Such a problem is called an *inverse* optimal control problem. The problem's input is the signal $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$ defined by $\mathbf{F}_{\boldsymbol{x}}$ that is assumed to be optimal w.r.t. an optimization problem of the form (4.5). The output is the objective function that $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x}$ is a minimizer of. In case of LQR, this objective is assumed to be quadratic and, hence, fully defined by the parameters $\mathbf{Q}$ and $\mathbf{R}$.

Motivation for starting research in this area was given by the question if a present, manually tuned, controller was optimal in some sense. This question was first analyzed by Kalman [76] in order to increase the acceptance of control design by means of LQR in the engineering community. In the mentioned work, the author focuses on single-input linear time-invariant plants, i.e. a quite restricted system class. Kreindler and Jameson [87] picked up this research and stated that every stabilizing control law (4.8d) is optimal if the objective (4.5a) is extended by a cross product term $\boldsymbol{x}^\top\mathbf{N}\boldsymbol{u}$ with some $\mathbf{N} \in \mathbb{R}^{n\times m}$. With this finding comes the fact that there exist state feedback matrices that are *not* optimal w.r.t. the standard problem formulation (4.5). Hence, we can already note now that LQRLC according to Definition 23 is not always solvable.

Inverse optimal control problems also arise in various other settings. For example, Menner et al. [104] use such a problem formulation to estimate the objective and the constraints for a human manipulation task that should be carried out in cooperation with a robotic arm. To this end, multiple human-triggered trajectories are recorded. As they are assumed to be optimal, these data are then used for identifying the corresponding parameters of the overlying optimal control problem. Such a procedure becomes more involved the less information is available about the structure of the OCP. Hence, assumptions on the structure of the objective or the type of constraints are commonly made. They must be individualized for the respective application. Other examples where inverse optimal control problems are used are given by Priess et al. [130] in a biological setting as well as Petrosian et al. [128] who aim at identifying the human driving behavior and encoding it into an OCP. This is beneficial for using human-created driving trajectories as a reference for further automatic trajectory planning.

4.2.2 Solvability

The main reason why our problem formulation, i.e. LQRLC, is different from classical inverse optimal control problems is that the controller assumed to be optimal w.r.t. the unknown objective is only partially parameterized. We only have its structure defined by Eq. (4.8c) which does usually not admit a unique solution. In classic works as the one by Jameson and Kreindler [68], $\mathbf{F}_{\boldsymbol{x}}$ is assumed to be fully known. Hence, we can use results from the literature but have to extend them such that they include the parametric structure at hand. First, let us state a set of necessary conditions for solving LQRLC.

Proposition 3 ([68, Th. 2.2] and [146, Proposition 1])**.** *LQRLC is solvable only if there exists a parameter vector* $\boldsymbol{q}_\mathrm{F} \in \mathbb{R}^{n_\mathrm{F}}$ *and, therefore, a structured controller* $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F}) \in \mathbb{R}^{m\times n}$ *in accordance with the constraint set* (4.8c) *such that the following conditions hold:*

1. $\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})$ *possesses only eigenvalues with negative real parts*
2. $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B} \in \mathbb{R}^{m\times m}$ *possesses m linear independent, real eigenvectors*
3. $\mathrm{rank}(\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}) = \mathrm{rank}(\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})) = \mathrm{rank}(\mathbf{B})$
4. $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ *possesses only non-positive eigenvalues*

Proof. Condition 1 is a stabilizability condition. Assuming $\mathbf{Q} \succ 0$ and $\mathbf{R} \succ 0$ in the LQR problem (4.5), the optimizer $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})$ must always render the closed loop stable because the objective function attains a finite value at its minimum. This is only possible if $\lim_{t\to\infty} \boldsymbol{x}(t) = \mathbf{0}$, i.e., the closed-loop system is stable. The remaining Conditions 2 – 4 are transferred from [68] and expressed in terms of the structured version of $\mathbf{F}_{\boldsymbol{x}}$, i.e. $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})$. They are necessary and sufficient for ensuring the existence of positive definite matrices $\mathbf{P}$ and $\mathbf{R}$ solving Eq. (4.6). □

The reason for Proposition 3 only embodying a set of necessary conditions is that it remains unclear whether there exists a matrix $\mathbf{Q} \succ 0$ for a compatible triple $\mathbf{F}_{\boldsymbol{x}}$, $\mathbf{P}$, $\mathbf{R}$. This matrix is uniquely determined via the algebraic Riccati equation (4.7). The map preserves symmetry, i.e., there sufficiently exists a symmetric matrix $\mathbf{Q} = \mathbf{Q}^\top$ for every pair $\mathbf{P}$, $\mathbf{R}$, both being positive definite. This can easily be understood by transposing the right side of (4.7). Positive definiteness of the left side is, however, not guaranteed. For the common case $m < n$, the first term $\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}$ in (4.7) is only positive *semi*-definite. This can be seen by decomposing it in a square form $\mathbf{M}\mathbf{M}^\top$ with $\mathbf{M} = \mathbf{P}\mathbf{B}(\mathbf{R}^{-1})^{1/2} \in \mathbb{R}^{n\times m}$ where $(\mathbf{R}^{-1})^{1/2}(\mathbf{R}^{-1})^{1/2} = \mathbf{R}$, and $\mathrm{rank}(\mathbf{M}) \leq m$.[2] Together with the Lyapunov-type term $-\mathbf{P}\mathbf{A} - \mathbf{A}^\top\mathbf{P}$ about whose definiteness little can be said for a given $\mathbf{P}$, this is not sufficient for positive definiteness of $\mathbf{Q}$.

A second challenge regarding Proposition 3 is the difficulty to check the conditions. The eigenstructure of the product $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ plays a significant role. In general, it is dependent on the choice of $\boldsymbol{q}_\mathrm{F}$. Therefore, it has to be analyzed analytically. Depending on the structure of the plant model, this can be mathematically involved.

Under the assumption that $\mathrm{rank}(\mathbf{B}) = m$, we can set up an alternative algebraic formulation of Proposition 3. The assumption made is quite common for dynamic systems and mild. It demands that every system input triggers a unique movement in the state space. In this case, Condition 3 condenses to $\mathrm{rank}(\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}) = m$. This implies that this product cannot possess eigenvalues at zero. Therefore, Condition 4 changes accordingly. Writing the remaining conditions by algebraic means results in the alternative formulation as

[2] Such a decomposition can be found, e.g., by computing a singular value decomposition of $\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}$.

Proposition 4. *Under the assumption* $\operatorname{rank}(\mathbf{B}) = m$*, LQRLC is solvable only if there exist matrices* $\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3 \succ 0$*, a parameter vector* $\boldsymbol{q}_\mathrm{F} \in \mathbb{R}^{n_\mathrm{F}}$ *and, therefore, a structured controller* $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F}) \in \mathbb{R}^{m\times n}$ *in accordance with the constraint set* (4.8c) *such that the following conditions hold:*

1. $(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F}))\mathbf{S}_1 + \mathbf{S}_1(\mathbf{A} + \mathbf{B}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F}))^\top \prec 0$
2. $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}\mathbf{S}_2 - \mathbf{S}_2\mathbf{B}^\top\mathbf{F}_{\boldsymbol{x}}^\top(\boldsymbol{q}_\mathrm{F}) = \mathbf{0}$
3. $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}\mathbf{S}_3 + \mathbf{S}_3\mathbf{B}^\top\mathbf{F}_{\boldsymbol{x}}^\top(\boldsymbol{q}_\mathrm{F}) \prec 0.$

Proof. We have to show equivalence to the conditions from Proposition 3. As already mentioned, $\operatorname{rank}(\mathbf{B}) = m$ yields Condition 3 as $\operatorname{rank}(\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}) = m$. By Condition 4 of Proposition 3, this results in the requirement of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ only having strictly negative eigenvalues. The Lyapunov inequality in Condition 3 of Proposition 4 accounts for this, see [23, p. 25]. Analogously, Condition 1 in Proposition 4 is equivalent to the stability condition from Proposition 3. Furthermore, the proof for Condition 2, Proposition 4, being equivalent to Condition 2 from Proposition 3 is derived in [44, Theorem 1]. □

Proposition 4 is stated here primarily for pointing out the difficulty of establishing solvability conditions that are simple to be checked. The new conditions transfer a parametric eigenstructure analysis to checking solvability of parametric linear matrix inequalities (LMIs) and the search for a positive definite matrix $\mathbf{S}_2$ that $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ commutes with. Strictly speaking, all conditions are nonlinear due to the product terms including the parameters $\boldsymbol{q}_\mathrm{F}$ and $\mathbf{S}_i$, $i \in \mathcal{I}_{1,3}$.

4.2.3 A Construction Scheme

Although it might be difficult to prove solvability of the LQRLC design, we now propose a design scheme for obtaining a solution. Recalling Definition 23, it is our aim to find matrices $\mathbf{Q}$ and $\mathbf{R}$ for the standard LQR procedure (4.5) that automatically lead to a structured controller of the desired form (4.8c). To achieve this, the general procedure is as follows. First, we transfer the controller constraints (4.8c) into constraints in the parameters of $\mathbf{P}$ that must solve Eq. (4.7). This is done by use of Eq. (4.6). The obtained constraints must necessarily be fulfilled for solving (4.5) and (4.8c) simultaneously. The construction of the input weights in $\mathbf{R}$ is crucial in this step because it influences the feasibility of the subsequent steps. After this, we can include the constraints in the parameters of $\mathbf{P}$ into the Riccati equation (4.7). This yields a parametric Riccati equation whose solution set has to be computed. I.e., we have to find the free parameters in $\mathbf{P}$ that render $\mathbf{P}$ and $\mathbf{Q}$ positive definite.

4.2.3.1 Transferring the Controller Constraints

The step for expressing the controller constraints (4.8c) in the parameters p_{ij} of $\mathbf{P}$, where $i, j \in \mathcal{I}_{1,n}$, is rather simple. Denoting $\boldsymbol{p} := \operatorname{vec}(\mathbf{P})$, we can vectorize Eq. (4.6) and insert it into (4.8c).

By [153, p. 296], this yields

$$\begin{aligned} & \operatorname{vec}(\mathbf{F}_{\boldsymbol{x}}) = -\operatorname{vec}(\mathbf{R}^{-1}\mathbf{B}^{\top}\mathbf{P}) \\ \Leftrightarrow \qquad & \boldsymbol{f}_{\boldsymbol{x}} = -\left(\mathbf{I}_n \otimes \mathbf{R}^{-1}\mathbf{B}^{\top}\right)\boldsymbol{p} \end{aligned}$$

resulting in the transferred constraints

$$\mathbf{Z}_{\mathrm{P}}\boldsymbol{p} = \boldsymbol{z}_{\mathrm{P}} \tag{4.9}$$

with $\boldsymbol{z}_{\mathrm{P}} = \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}}$ and

$$\mathbf{Z}_{\mathrm{P}}(\mathbf{R}) = -\mathbf{Z}_{\mathrm{F}\boldsymbol{x}}\left(\mathbf{I}_n \otimes \mathbf{R}^{-1}\mathbf{B}^{\top}\right).$$

As the solution $\mathbf{P}$ of Eq. (4.7) must be positive definite, it must necessarily be symmetric. We can directly add these symmetry constraints to (4.9). This is beneficial because it constrains a set of $\frac{1}{2}(n^2 - n)$ degrees of freedom from our design problem significantly reducing the problem size. The symmetry constraints are of the form $p_{ij} - p_{ji} = 0 \; \forall \; i > j$. They can be gathered in matrix form as $\mathbf{Z}_{\mathrm{sym}}\boldsymbol{p} = \mathbf{0}$. The construction of the coefficient matrix $\mathbf{Z}_{\mathrm{sym}}$ is shown in Section A.5. Combining both equations, we obtain

$$\begin{bmatrix} \mathbf{Z}_{\mathrm{P}}(\mathbf{R}) \\ \mathbf{Z}_{\mathrm{sym}} \end{bmatrix} \boldsymbol{p} = \begin{bmatrix} \boldsymbol{z}_{\mathrm{P}} \\ \mathbf{0} \end{bmatrix},$$

which we will abbreviate in the sequel by

$$\tilde{\mathbf{Z}}_{\mathrm{P}}(\mathbf{R})\,\boldsymbol{p} = \tilde{\boldsymbol{z}}_{\mathrm{P}}. \tag{4.10}$$

Each minimizer of the optimization problem (4.5) must necessarily solve Eq. (4.10). Otherwise, the controller constraints will not be met, or $\mathbf{P}$ cannot be positive definite.

4.2.3.2 Feasible Choice of the Input Weights

The next step is to discuss the solvability of Eq. (4.10) w.r.t. the specific choice of $\mathbf{R}$. There are two ways of finding a solution to (4.10). Either, we interpret the equation as a nonlinear problem, i.e., we allow $\mathbf{R}$ and $\boldsymbol{p}$ to be design parameters, or we choose $\mathbf{R}$ a priori such that we can expect (4.10) to have a solution. The latter aligns with the fact that the control designer does usually not have any requirements in the parameters of $\mathbf{P}$, but rather in $\mathbf{R}$ as this is the weighting matrix for the input signals. Therefore, we choose the second way of a priori choosing $\mathbf{R}$ in a specific way. This is outlined next.

In order to proceed, we assume that Conditions 2 – 4 of Proposition 3 or, equivalently, Conditions 2 and 3 of Proposition 4 are fulfilled. Doing so, we assume existence of positive definite matrices $\mathbf{P}$ and $\mathbf{R}$ solving Eq. (4.6). How strong this assumption is corresponds directly to the amount and structure of the controller constraints, and their interplay with the system dynamics. Consider the extremes: If there are no constraints, the problem is always solvable. Then, the set of feasible controllers $\mathbf{F}_{\boldsymbol{x}}$ w.r.t. Eq. (4.8c) is equivalent to the whole space $\mathbb{R}^{m\times n}$. Hence, any pair $\mathbf{Q}$, $\mathbf{R}$ will result in an optimal controller lying in this feasible set. If, however, all controller parameters are constrained, i.e. $\boldsymbol{q}_{\mathrm{F}} \in \{\}$, it is less likely that this specific controller $\mathbf{F}_{\boldsymbol{x}}$ is optimal

w.r.t. problem (4.5) because, now, the set of feasible controllers condenses to a single point in the solution space of (4.5). However, LQRLC might still be solvable, even in this case.

Relying on results from [68], let us construct $\mathbf{R}$. Assume, for now, that $\mathbf{F}_{\boldsymbol{x}}$ is fixed, i.e., $\boldsymbol{q}_{\mathrm{F}}$ has been chosen in accordance with Conditions 2 – 4 of Proposition 3. This brings us back to the original inverse optimal control problem discussed by Jameson and Kreindler [68]. By regularity, multiplying Eq. (4.6) by $\mathbf{R}$ from the left, i.e., writing

$$\mathbf{R}\mathbf{F}_{\boldsymbol{x}} = -\mathbf{B}^{\top}\mathbf{P}, \tag{4.11}$$

does not change the equation. Every triple $\mathbf{R}$, $\mathbf{F}_{\boldsymbol{x}}$, $\mathbf{P}$ must as well satisfy

$$\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B} = -\mathbf{B}^{\top}\mathbf{P}\mathbf{B}, \tag{4.12}$$

i.e. Eq. (4.11) multiplied by $\mathbf{B}$ from the right. Note that, as $\mathbf{B}$ is usually not regular, satisfying the latter is necessary but not sufficient for solving Eq. (4.11). The reason for writing Eq. (4.6) in this form is that we can deduce structural information from (4.12). As $\mathbf{P}$ is symmetric by its positive definiteness, we know that the product $\mathbf{B}^{\top}\mathbf{P}\mathbf{B}$ is symmetric and, likewise, $\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ must be *made* symmetric by choice of $\mathbf{R}$.

Satisfaction of Condition 2, Proposition 3, suffices for $\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ being diagonalizable. Hence, we can express this product as

$$\mathbf{F}_{\boldsymbol{x}}\mathbf{B} = \mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}\mathbf{V}_{\Lambda}$$

with $\mathbf{V}_{\Lambda}^{-1}$ as the eigenvector matrix, and $\boldsymbol{\Lambda}$ as the diagonal eigenvalue matrix. Replacing $\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ in Eq. (4.12) yields

$$\mathbf{R}\mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}\mathbf{V}_{\Lambda} = -\mathbf{B}^{\top}\mathbf{P}\mathbf{B}.$$

From this, we can infer that the choice $\mathbf{R} = \mathbf{V}_{\Lambda}^{\top}\mathbf{V}_{\Lambda}$ causes $\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ to be symmetric. Simultaneously, $\mathbf{R}$ is positive definite since $\operatorname{rank}(\mathbf{V}_{\Lambda}) = m$ by Condition 2 of Proposition 3. The authors of [68] state that there exist more degrees of freedom in the choice of $\mathbf{R}$. These are contained in a positive definite matrix $\boldsymbol{\Gamma} \succ 0$ that must commute with the eigenvalue matrix $\boldsymbol{\Lambda}$, i.e.

$$\boldsymbol{\Lambda}\boldsymbol{\Gamma} = \boldsymbol{\Gamma}\boldsymbol{\Lambda} \tag{4.13}$$

must hold. The matrix $\mathbf{R}$ can then be chosen as

$$\mathbf{R} = \mathbf{V}_{\Lambda}^{\top}\boldsymbol{\Gamma}\mathbf{V}_{\Lambda}. \tag{4.14}$$

It can easily be verified that $\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ is still symmetric. We have

$$\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B} = \underbrace{(\mathbf{V}_{\Lambda}^{\top}\boldsymbol{\Gamma}\mathbf{V}_{\Lambda})}_{=\mathbf{R}}\underbrace{(\mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}\mathbf{V}_{\Lambda})}_{=\mathbf{F}_{\boldsymbol{x}}\mathbf{B}} = \mathbf{V}_{\Lambda}^{\top}\boldsymbol{\Gamma}\boldsymbol{\Lambda}\mathbf{V}_{\Lambda},$$

which is equal to its own transpose $(\mathbf{R}\mathbf{F}_{\boldsymbol{x}}\mathbf{B})^{\top}$ in view of (4.13). As $\boldsymbol{\Gamma} \succ 0$, $\mathbf{R} \succ 0$ follows as well. It is shown by Jameson and Kreindler [68] that this choice of $\mathbf{R}$, in combination with the satisfaction of Conditions 2 – 4 of Proposition 3, is not only necessary for the solvability of Eq. (4.6), but

also sufficient. The latter, in turn, implies solvability of our structured design problem, as we have chosen $\mathbf{F}_{\boldsymbol{x}}$ in accordance with Eq. (4.8c).

Another important statement given in [68] is the fact that, given a fixed $\mathbf{F}_{\boldsymbol{x}}$, the choice (4.14) does not only parameterize feasible weighting matrices $\mathbf{R}$. It rather defines *all* feasible weighting matrices. The authors' results are based on the fixed controller $\mathbf{F}_{\boldsymbol{x}}$, whereas ours is only fixed in structure. Therefore, we need to discuss how the choice of $\boldsymbol{q}_{\mathrm{F}}$ that has been assumed before starting the design procedure influences the set of attainable matrices $\mathbf{R}$. This is not a simple question because this depends on the eigenstructure of the parametric matrix $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$. Hence, if this question should be answered for a specific LQRLC problem, the eigenstructure needs to be analyzed analytically in order to verify any dependencies on $\boldsymbol{q}_{\mathrm{F}}$.

A first case for which we can make a rigorous statement are problems where any choice of $\boldsymbol{q}_{\mathrm{F}}$ causes $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ to have the same eigenvectors, and some distinct eigenvalues:

Proposition 5 (see [146, Proposition 5]). *Let $\boldsymbol{q}_{\mathrm{F}}$ be a parameter vector that is compatible with Conditions 2 – 4 of Proposition 3. Assume that $\boldsymbol{q}_{\mathrm{F}}$ produces some matrix $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ with $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}\mathbf{V}_{\Lambda}^{-1} = \mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}$, where $\mathbf{V}_{\Lambda}^{-1}$ contains the eigenvectors of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$, and the diagonal matrix $\boldsymbol{\Lambda}$ contains its distinct eigenvalues, respectively. Then, the characterization $\mathbf{R} = \mathbf{V}_{\Lambda}^{\top}\boldsymbol{\Gamma}\mathbf{V}_{\Lambda}$ with $\boldsymbol{\Gamma} \succ 0$ and $\boldsymbol{\Lambda}\boldsymbol{\Gamma} = \boldsymbol{\Gamma}\boldsymbol{\Lambda}$ contains all matrices $\mathbf{R}$ which are attainable by choosing any other compatible $\tilde{\boldsymbol{q}}_{\mathrm{F}}$ producing some $\mathbf{F}_{\boldsymbol{x}}(\tilde{\boldsymbol{q}}_{\mathrm{F}})\mathbf{B}$ with distinct eigenvalues and the eigenvector matrix $\mathbf{V}_{\Lambda}^{-1}$.*

Proof. Let $\boldsymbol{q}_{\mathrm{F}i}$ with $i = 1, 2$ denote two parameter vectors compatible with Conditions 2 – 4 of Proposition 3 which produce the same eigenvector matrix $\mathbf{V}_{\Lambda}^{-1}$ and a set of distinct eigenvalues of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}i})\mathbf{B}$. That is, we have $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}i})\mathbf{B}\mathbf{V}_{\Lambda}^{-1} = \mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}_i$. Let $\mathbf{R}_i$ be constructed as $\mathbf{R}_i = \mathbf{V}_{\Lambda}^{\top}\boldsymbol{\Gamma}_i\mathbf{V}_{\Lambda}$ with $\boldsymbol{\Gamma}_i\boldsymbol{\Lambda}_i = \boldsymbol{\Lambda}_i\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Gamma}_i \succ 0$. As $\boldsymbol{\Lambda}_i$ is assumed to possess distinct diagonal elements, we know that $\boldsymbol{\Gamma}_i$ must be a diagonal matrix in order to commute with $\boldsymbol{\Lambda}_i$. But since all diagonal matrices commute with any diagonal matrix, $\boldsymbol{\Gamma}_i$ can be chosen arbitrarily diagonal and positive definite. If we choose them equally, i.e. $\boldsymbol{\Gamma}_1 = \boldsymbol{\Gamma}_2$, we obtain $\mathbf{R}_1 = \mathbf{R}_2$. □

The last sentence of the proof implies that, regardless which $\boldsymbol{q}_{\mathrm{F}i}$ was chosen, we can construct the same set of weighting matrices $\mathbf{R}$ from it. Hence, if the eigenvector matrix $\mathbf{V}_{\Lambda}^{-1}$ does not depend on $\boldsymbol{q}_{\mathrm{F}}$ and, for all $\boldsymbol{q}_{\mathrm{F}}$, no multiple eigenvalues occur in $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$, the specific choice of $\boldsymbol{q}_{\mathrm{F}}$ does not play any role for the set of attainable matrices $\mathbf{R}$. It can, therefore, be chosen randomly.

Next, let us consider cases where $\boldsymbol{q}_{\mathrm{F}}$ can be chosen such that the multiplicity of the eigenvalues of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ can change. I.e., there exist $\boldsymbol{q}_{\mathrm{F}}$ such that $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ only possesses distinct eigenvalues, but $\boldsymbol{q}_{\mathrm{F}}$ can also be chosen such that $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ possesses some multiple eigenvalues. Still assuming that $\mathbf{V}_{\Lambda}^{-1}$ is independent of $\boldsymbol{q}_{\mathrm{F}}$, the question arises, which scenario offers more degrees of freedom in the design of $\mathbf{R}$. We can record

Proposition 6. *Let $\boldsymbol{q}_{\mathrm{F}1}$, $\boldsymbol{q}_{\mathrm{F}2}$ be two parameter vectors compatible with Conditions 2 – 4 of Proposition 3. Assume that both $\boldsymbol{q}_{\mathrm{F}i}$ with $i = 1, 2$ produce some matrices $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}i})\mathbf{B}$ with $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}i})\mathbf{B}\mathbf{V}_{\Lambda}^{-1} = \mathbf{V}_{\Lambda}^{-1}\boldsymbol{\Lambda}_i$, where the eigenvectors in $\mathbf{V}_{\Lambda}^{-1}$ are independent of $\boldsymbol{q}_{\mathrm{F}i}$. Let $\boldsymbol{\Lambda}_1$ contain some multiple*

eigenvalues, whereas $\mathbf{\Lambda}_2$ only contains distinct eigenvalues. Then the set of attainable matrices $\mathbf{R}$ is strictly larger for the choice $\boldsymbol{q}_{\mathrm{F}1}$.

Proof. We need to show that, for $i = 1$, all matrices $\mathbf{R}$ are attainable that are attainable for $i = 2$, and that there is at least one additional choice. As we assume $\mathbf{V}_\Lambda^{-1}$ to be independent of $\boldsymbol{q}_{\mathrm{F}}$, all degrees of freedom for shaping $\mathbf{R}$ are contained in the choice of $\mathbf{\Gamma}$. Since $\mathbf{\Lambda}_2$ contains distinct eigenvalues, $\mathbf{\Gamma}_2$ must necessarily be a diagonal matrix such that it commutes with $\mathbf{\Lambda}_2$. However, if $\mathbf{\Gamma}_2$ is diagonal, it commutes with the diagonal matrix $\mathbf{\Lambda}_1$. This verifies that, for $i = 1$, we can create all $\mathbf{R}$ that we can create for $i = 2$.

For verifying that there are additional choices for $\mathbf{\Gamma}_1$ we write $\mathbf{\Lambda}_1$ in a block structure as

$$\mathbf{\Lambda}_1 = \begin{bmatrix} \mathbf{\Lambda}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{12} \end{bmatrix}$$

with $\mathbf{\Lambda}_{11} \in \mathbb{R}^{n_\lambda \times n_\lambda}$ and $\mathbf{\Lambda}_{12} \in \mathbb{R}^{(m-n_\lambda)\times(m-n_\lambda)}$ both being diagonal matrices. Here, $n_\lambda > 1$ denotes the multiplicity of some multiple eigenvalue $\lambda \in \mathbb{R}$. Without loss of generality assuming sorted eigenvalues, we can write $\mathbf{\Lambda}_{11} = \mathbf{I}_{n_\lambda}\lambda$. Then, we can partition $\mathbf{\Gamma}_1$ as

$$\mathbf{\Gamma}_1 = \begin{bmatrix} \mathbf{\Gamma}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_{12} \end{bmatrix}$$

with $0 \prec \mathbf{\Gamma}_{11} \in \mathbb{R}^{n_\lambda \times n_\lambda}$ and $0 \prec \mathbf{\Gamma}_{12} \in \mathbb{R}^{(m-n_\lambda)\times(m-n_\lambda)}$. Then, to satisfy the commutation requirement, only $\mathbf{\Gamma}_{12}$ must be diagonal, $\mathbf{\Gamma}_{11}$ can be chosen arbitrarily positive definite. Due to

$$\begin{bmatrix} \mathbf{I}_{n_\lambda}\lambda & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{\Gamma}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_{12} \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma}_{11}\lambda & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{12}\mathbf{\Gamma}_{12} \end{bmatrix} = \begin{bmatrix} \mathbf{\Gamma}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_{12} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{n_\lambda}\lambda & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{12} \end{bmatrix},$$

Eq. (4.13) still holds, verifying the claim of Proposition 6. □

Due to Proposition 6, if we have the option of choosing an initial $\boldsymbol{q}_{\mathrm{F}}$ such that $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ has some multiple eigenvalues as opposed to having only distinct eigenvalues, it is beneficial to choose the option with multiple eigenvalues. Assuming that both versions of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ have the same eigenvectors, there are more possibilities to choose the input weights $\mathbf{R}$ because more degrees of freedom are available via $\mathbf{\Gamma}$.

For other scenarios, e.g., for cases where both eigenvectors and eigenvalues depend on $\boldsymbol{q}_{\mathrm{F}}$, it is difficult to make general suggestions how to choose the initial parameters $\boldsymbol{q}_{\mathrm{F}}$. In such cases, $\boldsymbol{q}_{\mathrm{F}}$ can be seen as an additional design parameter.

Remark 14. Given a controller *structure*, we have based the search for weighting matrices $\mathbf{R}$ on one specific parameterization, i.e. one fixed controller $\mathbf{F}_{\boldsymbol{x}}$ satisfying the structure. As discussed above in detail, this sometimes does not change the set of attainable matrices $\mathbf{R}$. In some cases, degrees of freedom in the choice of $\mathbf{R}$ might be lost by this procedure. It is, however, important to note that the initial choice of $\boldsymbol{q}_{\mathrm{F}}$ does, in general, not fix the final result, i.e. the result of the unconstrained LQR procedure (4.5), as degrees of freedom are left in the design of $\mathbf{R}$ and $\mathbf{Q}$.

4.2.3.3 Obtaining the Structure of P and Q

Choosing the input weights $\mathbf{R}$ such that Eq. (4.10) is solvable is the crucial step towards solving LQRLC. Having fixed these weights, we can now proceed to integrating Eq. (4.10) into the Riccati equation (4.7). This will result in a fixed structure of the Riccati matrix $\mathbf{P}$ as well as the state weights $\mathbf{Q}$.

The solution of Eq. (4.10) is given by

$$\boldsymbol{p} = \tilde{\mathbf{Z}}_{\mathrm{P}}^{+}\tilde{\boldsymbol{z}}_{\mathrm{P}} + \tilde{\mathbf{Z}}_{\mathrm{P}}^{\perp}\boldsymbol{q}_{\mathrm{P}}, \tag{4.15}$$

where, again, $\tilde{\mathbf{Z}}_{\mathrm{P}}^{+}$ denotes the pseudo inverse and $\tilde{\mathbf{Z}}_{\mathrm{P}}^{\perp}$ is a kernel representation of $\tilde{\mathbf{Z}}_{\mathrm{P}}$. The remaining degrees of freedom are contained in the free choice of the vector $\boldsymbol{q}_{\mathrm{P}} \in \mathbb{R}^{n_{\mathrm{P}}}$. Based on our choice of $\mathbf{R}$, Eq. (4.15) describes all symmetric matrices $\mathbf{P}(\boldsymbol{q}_{\mathrm{P}})$ that produce, via Eq. (4.6), a structured controller of the form (4.8c).

The associated set of state weights is obtained by evaluating the algebraic Riccati equation Eq. (4.7), which yields

$$\mathbf{Q}(\boldsymbol{q}_{\mathrm{P}}) = \mathbf{P}(\boldsymbol{q}_{\mathrm{P}})\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{\top}\mathbf{P}(\boldsymbol{q}_{\mathrm{P}}) - \mathbf{P}(\boldsymbol{q}_{\mathrm{P}})\mathbf{A} - \mathbf{A}^{\top}\mathbf{P}(\boldsymbol{q}_{\mathrm{P}}). \tag{4.16}$$

By Eq. (4.7), $\mathbf{Q}(\boldsymbol{q}_{\mathrm{P}})$ is automatically symmetric. However, as already discussed in Section 4.2.2, this does not hold for positive definiteness. In fact, it is not known whether there exist any parameters $\boldsymbol{q}_{\mathrm{P}}$ rendering $\mathbf{P}$ and $\mathbf{Q}$ positive definite simultaneously, whence the lack of sufficiency of Propositions 3 and 4. Actively searching parameters $\boldsymbol{q}_{\mathrm{P}}$ establishing $\mathbf{P}, \mathbf{Q} \succ 0$ can be done based on

Theorem 12 (Sylvester's Criterion [54, p. 45])**.** *A real, symmetric matrix is positive definite if and only if all its principal minors are positive.*

The principal minors of a matrix

$$\mathbf{M} = \begin{bmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nn} \end{bmatrix} \in \mathbb{R}^{n\times n}$$

are defined by the set of determinants

$$\Delta_k(\mathbf{M}) := \det\left(\begin{bmatrix} m_{11} & \cdots & m_{1k} \\ \vdots & \ddots & \vdots \\ m_{k1} & \cdots & m_{kk} \end{bmatrix}\right)$$

for all $k \in \mathcal{I}_{1,n}$. In our case, i.e. parametric matrices $\mathbf{P}(\boldsymbol{q}_{\mathrm{P}})$, $\mathbf{Q}(\boldsymbol{q}_{\mathrm{P}})$, the principal minors are polynomials in the elements of $\boldsymbol{q}_{\mathrm{P}}$. Therefore, our parameter search problem is transformed in two sets of n polynomial inequalities

$$\begin{matrix} \Delta_k(\mathbf{P}(\boldsymbol{q}_{\mathrm{P}})) > 0 \\ \Delta_k(\mathbf{Q}(\boldsymbol{q}_{\mathrm{P}})) > 0 \end{matrix} \qquad \forall\, k \in \mathcal{I}_{1,n}. \tag{4.17}$$

We are interested in finding those parameters $\boldsymbol{q}_\mathrm{P}$ simultaneously satisfying all of the above polynomial inequalities. In general, numeric methods need to be used to solve this problem. One possibility is to use a bisection algorithm based on so-called Bernstein polynomials. Garloff and Graf [51] have developed such an algorithm. The underlying concepts of the algorithm are briefly outlined and explained in Section A.2.

4.2.3.4 Summary and Discussion of the Design Procedure

To conclude this section, we provide a compact overview over the steps for constructing a structured controller by means of LQR techniques. They are given in Algorithm 4, where solvability of the LQRLC problem is assumed.

Algorithm 4 Linear Quadratic Regulator Design with Linear Controller Constraints.

1: **Input:** $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Z}_{\mathrm{F}\boldsymbol{x}}$, $\boldsymbol{z}_{\mathrm{F}\boldsymbol{x}}$
2: Solve $\mathbf{Z}_{\mathrm{F}\boldsymbol{x}}\,\boldsymbol{f}_{\boldsymbol{x}} = \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}}$ and establish a parametric controller $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})$.
3: If doable, investigate the eigenstructure of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ for its dependence on $\boldsymbol{q}_\mathrm{F}$.
4: Choose an initial $\boldsymbol{q}_\mathrm{F}$ in accordance with Proposition 3. If there exists a feasible $\boldsymbol{q}_\mathrm{F}$ such that $\mathbf{F}_{\boldsymbol{x}}\mathbf{B}$ possesses multiple eigenvalues, this is a rather beneficial choice, see Proposition 6.
5: Design $\mathbf{R} = \mathbf{V}_\Lambda^\top \boldsymbol{\Gamma}\,\mathbf{V}_\Lambda$ according to Eq. (4.14) with $\boldsymbol{\Gamma} \succ 0$ and $\boldsymbol{\Gamma}\,\boldsymbol{\Lambda} = \boldsymbol{\Lambda}\,\boldsymbol{\Gamma}$.
6: Compute the structure of the Riccati matrix $\mathbf{P}$ via Eqs. (4.10) and (4.15).
7: Compute the structure of the state weights $\mathbf{Q}$ via Eq. (4.16).
8: Set up the polynomial inequalities (4.17).
9: Find feasible parameters $\boldsymbol{q}_\mathrm{P}$ rendering $\mathbf{P}$ and $\mathbf{Q}$ positive definite, e.g., by applying a bisection algorithm as the one outlined in Section A.2.
10: Choose any feasible parameter $\boldsymbol{q}_\mathrm{P}$ to design $\mathbf{Q} \succ 0$.
11: Compute the structured controller via Eq. (4.6).
12: **Output:** $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{F}_{\boldsymbol{x}}$, $\mathbf{P}$

LQRLC combines the design of arbitrary control-loop structures $\mathbf{G}_\mathrm{w}^*$, or arbitrarily pre-structured controllers, respectively, with a widely-used control design scheme. Using LQRLC for control design might especially be considered if the location of closed-loop eigenvalues providing a *good* dynamics is not known, i.e., if the aims of the dynamics design can be better described by designing the LQR weighting matrices $\mathbf{Q}$ and $\mathbf{R}$. In this case, LQRLC provides a set of input and state weights that, via Eq. (4.5), lead to a controller of the prescribed structure. A significant difference, e.g., w.r.t. the work by Hirzinger [64], is that the controller structure defined by Eq. (4.8c) is met exactly, not approximately. Depending on the exact design task, this is beneficial because this situation offers more transparency during the control design. A disadvantage of LQRLC is the search for a suitable input weighting matrix $\mathbf{R}$. For cases with a complex eigenstructure in $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_\mathrm{F})\mathbf{B}$, i.e. strong dependencies on $\boldsymbol{q}_\mathrm{F}$, obtaining a good intuition for the available degrees of freedom in the design of $\mathbf{R}$ can be difficult.

4.3 Prefilter Design

Design of the prefilter matrix $\mathbf{L}$ has been omitted so far and will be discussed now. Classically, the prefilter has the task to scale the reference signals $\boldsymbol{w}$ fed to the closed loop such that the steady-state tracking error

$$\boldsymbol{e}_\infty := \lim_{t\to\infty} \boldsymbol{w}(t) - \boldsymbol{y}(t)$$

satisfies $\boldsymbol{e}_\infty = \mathbf{0}$. Assuming a stabilizing controller $\mathbf{F}_x = \mathbf{F}_y\mathbf{C}_\mathrm{m}$, this is usually achieved by solving

$$\lim_{s\to 0} \mathbf{G}_\mathrm{w}(s) = \left(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_x)^{-1}\mathbf{B}+\mathbf{D}\right)\mathbf{L} \stackrel{!}{=} \mathbf{I}_p \tag{4.18}$$

for $\mathbf{L}$ yielding

$$\mathbf{L} = \left(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_x)^{-1}\mathbf{B}+\mathbf{D}\right)^{-1}. \tag{4.19}$$

This inversion exists for each non-degenerate system without invariant zeros at $s = 0$, and $p = m$, i.e. every system with $\operatorname{rank}(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_x)^{-1}\mathbf{B}+\mathbf{D}) = p$.

Designing a suitable structured control law involves obeying the constraints (3.5b) that appear for establishing the desired structure $\mathbf{G}_\mathrm{w}^*$. Therefore, $\mathbf{L}$ can in general not be chosen by Eq. (4.19) because there is no guarantee that the resulting matrix $\mathbf{L}$ solves (3.5b). Hence, the constraints must explicitly be taken into account.

Let us define a matrix $\mathbf{G}_\mathrm{w}^\infty \in \mathbb{R}^{p\times p}$ that encodes some desired steady-state gain values for $\mathbf{G}_\mathrm{w}(s)$. It must be compatible to the structure matrix $\mathbf{G}_\mathrm{w}^*$, i.e., the elements $g_{\mathrm{w}ij}^\infty$, $i, j \in \mathcal{I}_{1,p}$, of $\mathbf{G}_\mathrm{w}^\infty$ are constrained as

$$g_{\mathrm{w}ij}^\infty = \begin{cases} 0, & g_{\mathrm{w}ij}^* = 0 \\ g_{ij}, & g_{\mathrm{w}ij}^* = * \end{cases}$$

with some finite numbers $g_{ij} \in \mathbb{R}\backslash\{0\}$. Choosing $\mathbf{G}_\mathrm{w}^\infty$ and $\mathbf{G}_\mathrm{w}^*$ not compatible may result in conflicting design goals. E.g., if $g_{\mathrm{w}ij}^* = 0$ but $g_{\mathrm{w}ij}^\infty \neq 0$, then, assuming solvability of the structural control problem defined in Definition 20, we will have $g_{\mathrm{w}ij}(s) \equiv 0$ by structure of the control law. But, we would simultaneously try to use $\mathbf{L}$ to scale the steady-state gain of this element to a non-zero value. This is obviously not feasible using a finite $\mathbf{L}$.

Analog to problem (4.18), we need to solve

$$\left(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_x)^{-1}\mathbf{B}+\mathbf{D}\right)\mathbf{L}(\boldsymbol{q}_\mathrm{L}) \stackrel{!}{=} \mathbf{G}_\mathrm{w}^\infty, \tag{4.20}$$

where $\mathbf{L}$ solves Eq. (3.5b) as described by Eq. (4.1b). Due to $\mathbf{L}$ being structured, it is unclear whether Eq. (4.20) admits a feasible solution. For this reason, we turn it into a minimization problem in the free parameters $\boldsymbol{q}_\mathrm{L} \in \mathbb{R}^{n_\mathrm{L}}$. This can most conveniently be set up by iterating through the columns of Eq. (4.20). The j-th column of the equation is

$$\left(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_x)^{-1}\mathbf{B}+\mathbf{D}\right)\mathbf{L}(\boldsymbol{q}_\mathrm{L})\boldsymbol{\alpha}_j \stackrel{!}{=} \mathbf{G}_\mathrm{w}^\infty\boldsymbol{\alpha}_j$$

with $\boldsymbol{\alpha}_j \in \mathbb{R}^p$ as the canonical unit vector in the j-th coordinate direction. Using Eq. (4.1b), this can be written directly in the parameters $\boldsymbol{q}_\mathrm{L}$ as

$$\underbrace{\left(-\mathbf{C}(\mathbf{A}+\mathbf{B}\mathbf{F}_{\boldsymbol{x}})^{-1}\mathbf{B}+\mathbf{D}\right)\begin{bmatrix}\mathbf{0}_{m\times(j-1)m} & \mathbf{I}_m & \mathbf{0}_{m\times(p-j)m}\end{bmatrix}}_{=:\,\mathbf{G}_j}\left((\mathbf{Z}_\mathrm{L})^{+}\,\boldsymbol{z}_\mathrm{L}+(\mathbf{Z}_\mathrm{L})^{\perp}\,\boldsymbol{q}_\mathrm{L}\right)\overset{!}{=}\mathbf{G}_\mathrm{w}^{\infty}\boldsymbol{\alpha}_j. \tag{4.21}$$

The corresponding error equations are defined by

$$\boldsymbol{e}_{\boldsymbol{q}_\mathrm{L}j}(\boldsymbol{q}_\mathrm{L}) := \mathbf{G}_\mathrm{w}^{\infty}\boldsymbol{\alpha}_j - \mathbf{G}_j\,(\mathbf{Z}_\mathrm{L})^{+}\,\boldsymbol{z}_\mathrm{L} - \mathbf{G}_j\,(\mathbf{Z}_\mathrm{L})^{\perp}\,\boldsymbol{q}_\mathrm{L} \tag{4.22}$$

for all $j \in \mathcal{I}_{1,p}$. The design goal is to choose $\boldsymbol{q}_\mathrm{L}$ such that

$$J_{\boldsymbol{q}_\mathrm{L}}(\boldsymbol{q}_\mathrm{L}) = \sum_{j=1}^{p} \boldsymbol{e}_{\boldsymbol{q}_\mathrm{L}j}^{\top}\boldsymbol{e}_{\boldsymbol{q}_\mathrm{L}j} \tag{4.23}$$

is minimized. This step can be solved analytically. To this end, all equations of the form (4.21) are stacked as

$$\begin{bmatrix}\mathbf{G}_1\\ \vdots\\ \mathbf{G}_p\end{bmatrix}(\mathbf{Z}_\mathrm{L})^{\perp}\,\boldsymbol{q}_\mathrm{L} = \mathrm{vec}(\mathbf{G}_\mathrm{w}^{\infty}) - \begin{bmatrix}\mathbf{G}_1\\ \vdots\\ \mathbf{G}_p\end{bmatrix}(\mathbf{Z}_\mathrm{L})^{+}\,\boldsymbol{z}_\mathrm{L}$$

which are optimally solved in the sense of (4.23) by

$$\boldsymbol{q}_\mathrm{L} = \left(\begin{bmatrix}\mathbf{G}_1\\ \vdots\\ \mathbf{G}_p\end{bmatrix}(\mathbf{Z}_\mathrm{L})^{\perp}\right)^{+}\cdot\left(\mathrm{vec}(\mathbf{G}_\mathrm{w}^{\infty}) - \begin{bmatrix}\mathbf{G}_1\\ \vdots\\ \mathbf{G}_p\end{bmatrix}(\mathbf{Z}_\mathrm{L})^{+}\,\boldsymbol{z}_\mathrm{L}\right). \tag{4.24}$$

The proposed design of the prefilter solves Eq. (4.20) exactly whenever this is feasible. Otherwise, an error-minimizing alternative is chosen, while obeying the structural constraints (3.5b). The design can simply be extended to the multiple-plant case introduced in Section 3.3.2 by additionally setting up and minimizing the errors (4.22), newly named $\boldsymbol{e}^{\boldsymbol{\theta}}_{\boldsymbol{q}_\mathrm{L}j}$, for each parameter instance $\boldsymbol{\theta} \in \Theta$ providing an objective function of the form

$$J^{\Theta}_{\boldsymbol{q}_\mathrm{L}}(\boldsymbol{q}_\mathrm{L}) = \sum_{\boldsymbol{\theta}\in\Theta}\sum_{j=1}^{p}(\boldsymbol{e}^{\boldsymbol{\theta}}_{\boldsymbol{q}_\mathrm{L}j})^{\top}\boldsymbol{e}^{\boldsymbol{\theta}}_{\boldsymbol{q}_\mathrm{L}j}.$$

4.4 Example

Let us continue the example from Section 3.6 and parameterize the control law (3.41) that establishes the structured control loop $\mathbf{G}^{*}_{\mathrm{w}2}$. The main focus lies on showing the steps of LQRLC for a single plant. However, we will also demonstrate multiple-plant design by means of pole region assignment.

4.4.1 Single-Plant Design

For the structure $\mathbf{G}_{\mathrm{w}2}^{*}$, the design problem is exactly solvable, as previously discussed in Section 3.6.1. The controller and prefilter constraints are given by Eqs. (3.43) and (3.44) inducing the structured matrices given by Eq. (3.45). In the sequel, we will use LQRLC to finalize these structured matrices.

First, the conditions for the existence of an optimal control law $\boldsymbol{u} = \mathbf{F}_{\boldsymbol{x}}\boldsymbol{x} = \mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}}\boldsymbol{y}_{\mathrm{m}}$ satisfying the constraints (3.43) given in Proposition 3 should be checked. As discussed before, they are not sufficient. However, due to their necessity, they are a good estimator for the chances of success. For our scenario, we have

$$\begin{aligned}\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B} &= \begin{bmatrix} f_{31}+f_{34}+4.52 & -9.04 & 0 & 0 \\ f_{31}-f_{24}+f_{34}+4.52 & -9.04 & 0 & f_{24} \\ f_{31} & 0 & 0 & f_{34} \end{bmatrix} \begin{bmatrix} \frac{1}{A_1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{A_3} & 0 \\ 0 & 0 & \frac{1}{A_4} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{A_1}(f_{31}+f_{34}+4.52) & 0 & 0 \\ \frac{1}{A_1}(f_{31}-f_{24}+f_{34}+4.52) & 0 & \frac{f_{24}}{A_4} \\ \frac{1}{A_1}f_{31} & 0 & \frac{f_{34}}{A_4} \end{bmatrix}.\end{aligned}$$

It is clear to see that the rank condition from Proposition 3, i.e. Condition 3, is violated. This is due to

$$\max_{\boldsymbol{q}_{\mathrm{F}}} \operatorname{rank}(\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}) = 2 < 3 = \operatorname{rank}(\mathbf{B})$$

caused by the second column of $\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}})\mathbf{B}$ being zero. This zero column exists because the third column of $\mathbf{F}_{\boldsymbol{x}}$ is forced to zero by our demand of creating an output feedback. Hence, the structure $\mathbf{G}_{\mathrm{w}2}^{*}$ cannot be parameterized by LQRLC if we demand for the design of this output feedback configuration. No such structured output feedback is optimal w.r.t. the problem (4.5).

For this reason, let us allow for a full state feedback, instead. The corresponding constraints have not been displayed yet. Written in terms of the state feedback parameters, they read

$$\begin{bmatrix} 1 & 0 & -1 & & 0 & -1 & 0 & 0 & -1 \\ 0 & 1 & -1 & \mathbf{0}_{2\times 4} & 1 & -1 & 0 & 1 & -1 \\ \multicolumn{2}{c}{\mathbf{0}_{4\times 3}} & & \mathbf{I}_4 & \multicolumn{4}{c}{\mathbf{0}_{4\times 5}} & \\ & & & \mathbf{0}_{1\times 9} & & & 1 & 0 & 0 \end{bmatrix} \boldsymbol{f}_{\boldsymbol{x}} = \begin{bmatrix} 4.52 \\ 4.52 \\ -9.04 \\ -9.04 \\ \mathbf{0}_{3\times 1} \end{bmatrix}$$

and imply the controller structure

$$\mathbf{F}_{\boldsymbol{x}}(\boldsymbol{q}_{\mathrm{F}}) = \begin{bmatrix} f_{31}+f_{33}+f_{34}+4.52 & -9.04 & 0 & 0 \\ f_{31}-f_{24}-f_{23}+f_{33}+f_{34}+4.52 & -9.04 & f_{23} & f_{24} \\ f_{31} & 0 & f_{33} & f_{34} \end{bmatrix}.$$

Using this structure, it can be verified that all conditions from Proposition 3 can be satisfied. One feasible initial parameterization of $\boldsymbol{q}_{\mathrm{F}}$ is given by $\boldsymbol{q}_{\mathrm{F}} = [f_{31} \quad f_{23} \quad f_{33} \quad f_{24} \quad f_{34}]^{\top} =$

$-10 \cdot [1 \quad 2 \quad 0 \quad 0 \quad 2]^\top$. These parameters are chosen with the aim of allowing for a nicely structured eigenvector matrix $\mathbf{V}_\Lambda^{-1}$ of $\mathbf{F}_x(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ and have been found iteratively. The resulting eigenstructure is defined by the matrices

$$\mathbf{V}_\Lambda^{-1} = \begin{bmatrix} 0.43 & 0 & 0 \\ 0.43 & 1 & 0 \\ 0.79 & 0 & 1 \end{bmatrix} \qquad \text{and} \qquad \mathbf{\Lambda} = \begin{bmatrix} -0.25 & & \\ & -0.2 & \\ & & -0.2 \end{bmatrix}.$$

Remark 15. In this example, the eigenstructure of $\mathbf{F}_x(\boldsymbol{q}_\mathrm{F})\mathbf{B}$ depends on $\boldsymbol{q}_\mathrm{F}$. Hence, $\boldsymbol{q}_\mathrm{F}$ is a degree of freedom in the design step. We use it explicitly to choose an eigenstructure of $\mathbf{F}_x\mathbf{B}$ that can easily be interpreted in the following steps. The simpler the structure of $\mathbf{V}_\Lambda^{-1}$, the more intuition is added to the choice of the matrix $\mathbf{\Gamma}$ and how it influences the input weights $\mathbf{R}$. The choices $f_{23} = f_{34}$ and $f_{33} = f_{24} = 0$ are responsible for this simple structure as it creates a diagonal block in $\mathbf{F}_x\mathbf{B}$. For an example where the product $\mathbf{F}_x\mathbf{B}$ is independent of $\boldsymbol{q}_\mathrm{F}$, the reader is referred to the prior publication [146]. In such scenarios, $\boldsymbol{q}_\mathrm{F}$ has no influence on the results of LQRLC which is due to Propositions 5 and 6.

As explained above, we can now adjust the input weights in $\mathbf{R} = \mathbf{V}_\Lambda^\top \mathbf{\Gamma} \mathbf{V}_\Lambda$ via the choice of $\mathbf{\Gamma}$. Two examples are given for $\mathbf{\Gamma}_1 = \mathbf{I}_3$ and $\mathbf{\Gamma}_2 = \mathrm{diag}(1,\, 0.5,\, 0.5)$ creating the respective input weights

$$\mathbf{R}_1 = \mathbf{V}_\Lambda^\top \mathbf{\Gamma}_1 \mathbf{V}_\Lambda = \begin{bmatrix} 9.66 & -1 & -1.83 \\ -1 & 1 & 0 \\ -1.83 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{R}_2 = \mathbf{V}_\Lambda^\top \mathbf{\Gamma}_2 \mathbf{V}_\Lambda = \begin{bmatrix} 7.50 & -0.50 & -0.91 \\ -0.50 & 0.50 & 0 \\ -0.91 & 0 & 0.50 \end{bmatrix}.$$

We see that the structure of $\mathbf{V}_\Lambda$ distorts the preliminary weights in $\mathbf{\Gamma}$. Nevertheless, it is possible to develop an intuition for the choice of $\mathbf{\Gamma}$ and its effect on $\mathbf{R}$. Weighting the actuators in $\mathbf{\Gamma}$ evenly by choice of $\mathbf{\Gamma}_1$, the resulting weights for u_1 on the one side and u_2 and u_3 on the other side deviate by a factor of roughly 10. Lowering the preliminary weights for u_2 and u_3 in $\mathbf{\Gamma}_2$ has an expected effect on $\mathbf{R}$. Although the change in $\mathbf{\Gamma}$ does not transfer one by one to $\mathbf{R}$, i.e., we do not have $\mathbf{R}_2 \neq \mathbf{R}_1\mathbf{\Gamma}_2$, we can see that the relative weighting between u_1 and the pair u_2, u_3 has been changed proportionally to the values in $\mathbf{\Gamma}_2$.

The next design step is setting up the constraint set (4.10) for the entries of $\mathbf{P}$. As this is a technicality, we only present the results as

$$\underbrace{\begin{bmatrix} & & & -1 & & a(\mathbf{R}) \\ \mathbf{I}_5 & & \mathbf{0}_{5\times4} & 0 & \mathbf{0}_{9\times4} & \mathbf{0}_{2\times1} \\ & & & 1 & & \\ & \mathbf{0}_{13\times1} & \mathbf{I}_4 & \mathbf{0}_{4\times1} & & 1.83 \\ & & & 1 & & \mathbf{0}_{6\times1} \\ \mathbf{0}_{8\times5} & & & & \mathbf{I}_4 & 1.83 \\ & & \mathbf{0}_{4\times4} & \mathbf{0}_{5\times1} & & \mathbf{0}_{2\times1} \end{bmatrix}}_{=\tilde{\mathbf{Z}}_\mathrm{P}(\mathbf{R})} \boldsymbol{p} = \tilde{\boldsymbol{z}}_\mathrm{P}(\mathbf{R})$$

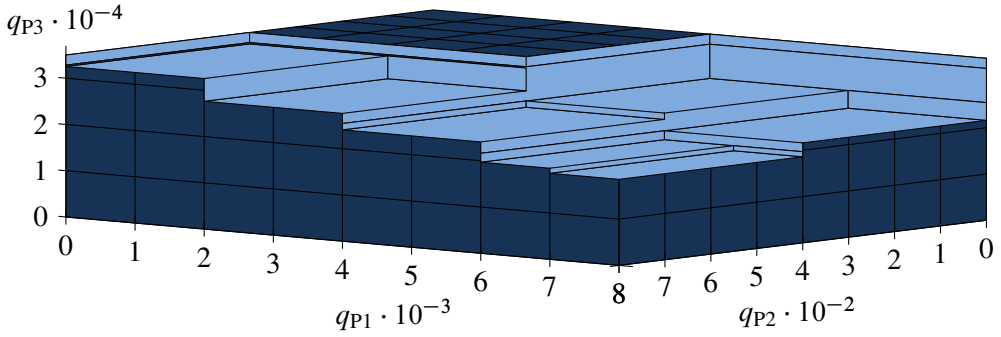


Figure 4.2: Illustration of the approximated feasible region for the parameters $\boldsymbol{q}_{\mathrm{P}}$.

with

$$\tilde{\boldsymbol{z}}_{\mathrm{P}}(\mathbf{R}_1) = \begin{bmatrix} 2920 & 7830 & 0 & 0 & 7830 & 0 & -1650 & 0 & 0 & 0 & 0 & -1650 & 0 \end{bmatrix}^{\top}$$
$$\tilde{\boldsymbol{z}}_{\mathrm{P}}(\mathbf{R}_2) = \begin{bmatrix} 2920 & 6330 & 0 & 0 & 6330 & 0 & -825 & 0 & 0 & 0 & 0 & -825 & 0 \end{bmatrix}^{\top}$$

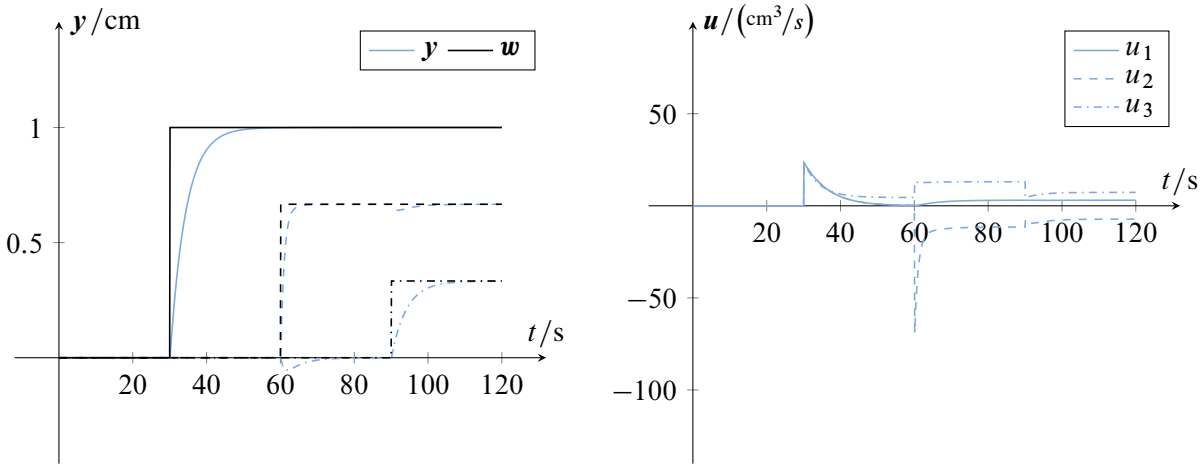
as well as $a(\mathbf{R}_1) = -8.66$ and $a(\mathbf{R}_2) = -13.99$, respectively[3]. By the choice of $\mathbf{R}$ and the initial parameters $\boldsymbol{q}_{\mathrm{F}}$ in accordance with the conditions from Proposition 3, solvability of the above constraints with some $\mathbf{P} \succ 0$ is ensured. Using their solution from Eq. (4.15), we are now able to obtain the structures of $\mathbf{P}$ and the weighting matrix $\mathbf{Q}$ in dependence of the free parameters $\boldsymbol{q}_{\mathrm{P}} \in \mathbb{R}^{n_{\mathrm{P}}}$. Due to $r_{\mathrm{P}} = \operatorname{rank}(\tilde{\mathbf{Z}}_{\mathrm{P}}(\mathbf{R})) = 13$, we have $n_{\mathrm{P}} = n^2 - r_{\mathrm{P}} = 16 - 13 = 3$ free parameters.

For the scenario based on $\mathbf{R}_1$, Fig. 4.2 gives an impression of the approximated solution set of the polynomial inequalities (4.17) depending on $\boldsymbol{q}_{\mathrm{P}}$. Choosing some randomly drafted admissible parameters $\boldsymbol{q}_{\mathrm{P1}} = [3000 \quad 450 \quad 27500]^{\top}$, we obtain $\mathbf{P}(\boldsymbol{q}_{\mathrm{P1}}) \succ 0$ and $\mathbf{Q}(\boldsymbol{q}_{\mathrm{P1}}) \succ 0$ as

$$\mathbf{P} = \begin{bmatrix} 18910 & 7832 & -8219 & -1637 \\ 7832 & 14130 & 0 & -1650 \\ -8219 & 0 & 8219 & 0 \\ -1637 & -1650 & 0 & 897.2 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 10130 & 855 & -9782 & 623 \\ 855 & 4415 & -1061 & -584.9 \\ -9782 & -1061 & 10340 & -1018 \\ 623 & -584.9 & -1018 & 390.9 \end{bmatrix},$$

respectively. We do not show $\mathbf{P}(\boldsymbol{q}_{\mathrm{P2}})$ and $\mathbf{Q}(\boldsymbol{q}_{\mathrm{P2}})$ here as little additional information can be drawn from them. Considering the above matrices, we find that they are heavily structured. Denoting $\boldsymbol{x} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^{\top}$, we infer that mixed terms $x_i x_j$, $i \neq j$, play an important role in the objective function of (4.5). This is because their coefficients which are encoded in the off-diagonal elements of $\mathbf{Q}$ have significant non-zero values. Therefore, interpretability of $\mathbf{Q}$ as a free-to-design state weighting matrix is not given for this example. Rather, we can conclude that LQRLC enables to parameterize the structured controller on the basis of well-designable input weights.

[3] In the derivation of Eq. (4.9), we have stated that $\boldsymbol{z}_{\mathrm{P}} = \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}}$ is independent of the choice of $\mathbf{R}$. The reason for this being different here is that we have displayed a reduced row echelon form of the original equation (4.9). This has been done for improving legibility. Although both forms are equivalent, due to the standardized coefficient matrix, the right side varies depending on different choices of $\mathbf{R}$.



(a) Output evolution. Solid lines (y_1, w_1), dashed lines (y_2, w_2), dash-dotted lines (y_3, w_3).

(b) Input evolution.

Figure 4.3: Output and input evolution for the control law obtained with $\mathbf{R}_1$.

Via Eq. (4.6), we finally arrive at the constrained controllers

$$\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P1}}) = \begin{bmatrix} -14.45 & -9.04 & 0 & 0 \\ 67.74 & -9.04 & -82.19 & 0 \\ -10.00 & 0 & 0 & -8.97 \end{bmatrix}$$

and, with $\boldsymbol{q}_{\mathrm{P2}} = [1559 \quad 502 \quad 28970]^\top$,

$$\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P2}}) = \begin{bmatrix} -23.47 & -9.04 & 0 & 0 \\ 153.14 & -9.04 & -176.60 & 0 \\ -10 & 0 & 0 & -17.98 \end{bmatrix}.$$

These controllers assign the closed-loop eigenvalue sets

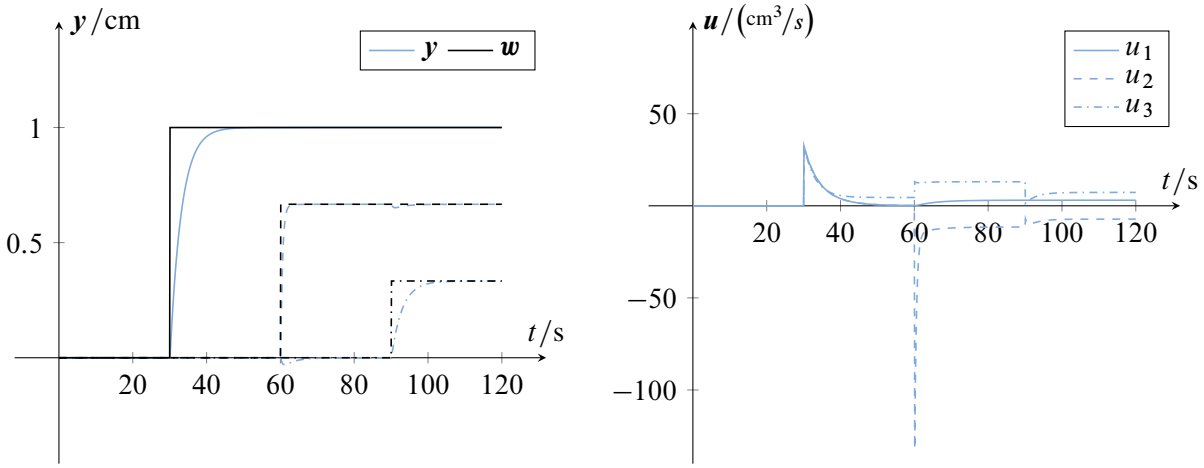
$$\sigma(\mathbf{A} + \mathbf{B}\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P1}})) = \{-1.07, -0.18, -0.23, -0.24\}$$
$$\sigma(\mathbf{A} + \mathbf{B}\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P2}})) = \{-1.99, -0.18, -0.32, -0.34\}.$$

Furthermore, with $\mathbf{G}_{\mathrm{w}}^{\infty} = \mathbf{I}_3$ and Eq. (4.24), we obtain the prefilter matrices

$$\mathbf{L}_1 = \begin{bmatrix} 23.49 & 0 & 0 \\ 23.49 & -103.99 & 12.80 \\ 23.49 & 12.82 & -26.27 \end{bmatrix} \quad \text{and} \quad \mathbf{L}_2 = \begin{bmatrix} 32.50 & 0 & 0 \\ 32.50 & -198.40 & 12.80 \\ 32.50 & 12.82 & -35.28 \end{bmatrix},$$

respectively.

The parameters $\boldsymbol{q}_{\mathrm{P2}}$ have been chosen such that they are in the same order of magnitude as $\boldsymbol{q}_{\mathrm{P1}}$, aiming for comparable results. Doing so, we see the effect of the input weights on the final controllers. By the changed weight balance in $\mathbf{R}_2$, the magnitudes of the signals u_2 and u_3 are expected to grow in relation to the magnitude of u_1.



(a) Output evolution. Solid lines (y_1, w_1), dashed lines (y_2, w_2), dash-dotted lines (y_3, w_3).

(b) Input evolution.

Figure 4.4: Output and input evolution for the control law obtained with $\mathbf{R}_2$.

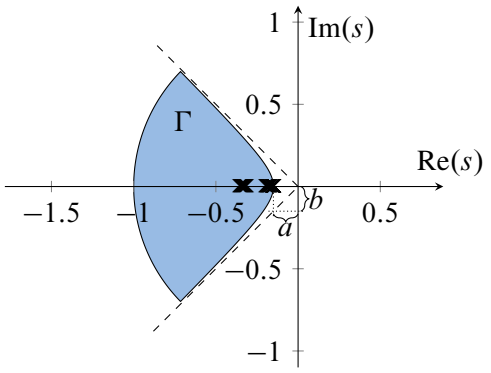
In the Figures 4.3 and 4.4, the step response plots are shown for the closed loops controlled by $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P1}})$, $\mathbf{L}_1$ and $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P2}})$, $\mathbf{L}_2$, respectively. In both cases, applying reference steps in $\boldsymbol{w}$, the corresponding output responses are decoupled according to the imposed structure $\mathbf{G}_{\mathrm{w2}}^*$. That is, a step in w_1 only influences y_1, but no other output. In turn, steps in w_2 and w_3 do not impact y_1. Furthermore, the stated expectations concerning the dynamics are visible. The reduction of the input weight for u_2 has the largest effect. Its magnitude rises significantly for the second step at $t = 60$s in Fig. 4.4b. The effect of changing the weight of u_3 is less apparent. The respective control signal only changes significantly for the first step at $t = 30$s. Overall, by the decrease of the input weights, the dynamics of the second control loop is faster. This can be seen by a comparison of the Figures 4.3a and 4.4a.

Remark 16. From the plots, besides successfully implementing $\mathbf{G}_{\mathrm{w2}}^*$, the controllers $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P1}})$ and $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P2}})$ significantly suppress the signal flows $w_2 \to y_3$ and $w_3 \to y_2$. The second controller $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P2}})$ is performing slightly better than $\mathbf{F}_x(\boldsymbol{q}_{\mathrm{P1}})$ for this task. Since the designed controllers naturally approximate $\mathbf{G}_{\mathrm{w1}}^*$, it seems unnecessary to use a dedicated approximating controller as the one from Eq. (3.48) in this specific case.

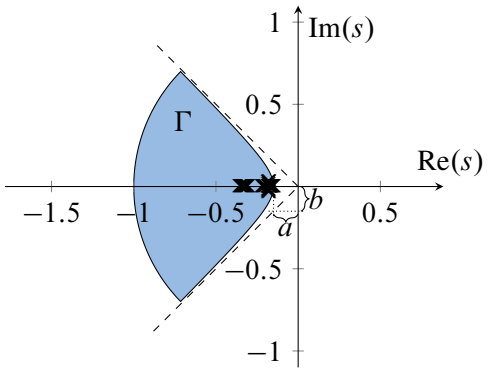
4.4.2 Multiple-Plant Design

If parametric uncertainties should be explicitly taken into account during structured control design, pole region assignment is the method that is best suited for this task. Furthermore, as this procedure is a numeric optimization in the controller parameters, it can handle the originally designed output feedback for realizing $\mathbf{G}_{\mathrm{w2}}^*$. Recall that this was not possible for LQRLC because no optimal controller in the sense of (4.5) exists having the required structure.

In Section 3.6.2, we have found that the structure $\mathbf{G}_{\mathrm{w2}}^*$ is structurally robust w.r.t. the plant parameters q_2 and A_2. This implies that the optimization problem (4.4) remains feasible if only these



(a) Closed-loop eigenvalues by the controller (4.25).



(b) Closed-loop eigenvalues for (4.27).

Figure 4.5: Pole region Γ all eigenvalues should be assigned to. The parameters are chosen as $a = b = 0.15$. The crosses indicate the closed-loop eigenvalue locations.

parameters are subject to uncertainty. Then, $\mathbf{G}_{\mathrm{w}2}^{*}$ can exactly be established by a single structured controller $\mathbf{F}_{\boldsymbol{y}}$ and a corresponding prefilter $\mathbf{L}$. In the sequel, the finite parameter set Θ is chosen as

$$\Theta = \left\{ \operatorname{diag}\left(\begin{bmatrix} 1 & 1+\lambda & \mathbf{1}_7^\top & 1+\lambda & \mathbf{1}_2^\top \end{bmatrix} \right) \boldsymbol{\theta}_0 \,\middle|\, \lambda \in \frac{1}{50}\mathcal{I}_{-4,4} \right\},$$

with $\boldsymbol{\theta}_0 = [1 \quad 1 \quad 1 \quad 1 \quad 39 \quad 33 \quad 27 \quad 24 \quad 100 \quad 100 \quad 100 \quad 100]^\top$. For obtaining the structured controller based on (4.4), we replace the constraint (4.4c) by its multiple-parameter version, i.e. $\sigma_{\mathrm{F}}(\boldsymbol{\theta}, \mathbf{F}_{\boldsymbol{y}}) \subset \Gamma \; \forall \, \boldsymbol{\theta} \in \Theta$, where the underlying pole region Γ, along with the later assigned closed-loop eigenvalues, is depicted in Fig. 4.5a. Setting $\tilde{J} = 0$, i.e. omitting any secondary design objectives, and using the initial value $\mathbf{F}_{\boldsymbol{y},0} = [\mathbf{1}_3 \quad \mathbf{0}_{3\times 2}]$ for the gradient based optimization procedure, we obtain

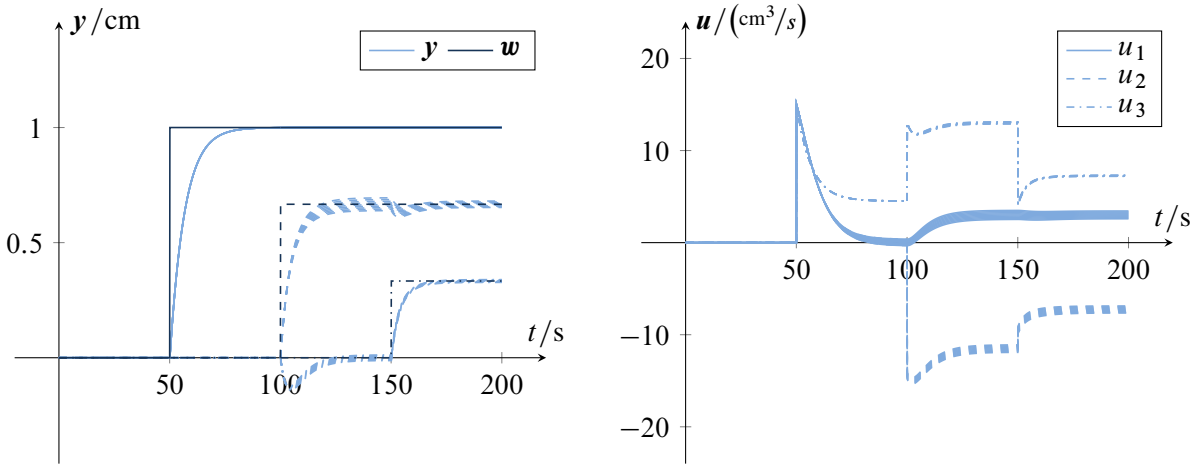
$$\mathbf{F}_{\boldsymbol{y}} = \begin{bmatrix} -5.96 & -9.04 & 0 \\ 0.88 & -9.04 & -6.84 \\ -1.37 & 0 & -9.11 \end{bmatrix} \tag{4.25}$$

as a result. Again choosing $\mathbf{G}_{\mathrm{w}}^{\infty} = \mathbf{I}_3$, the prefilter is determined via the multiple-plant extension of Eq. (4.24) as

$$\mathbf{L} = \begin{bmatrix} 15.00 & 0 & 0 \\ 15.00 & -21.78 & 5.90 \\ 15.00 & 12.73 & -26.36 \end{bmatrix}. \tag{4.26}$$

The implementation of pole region assignment applied here is introduced in detail by Vogt et al. [165], and also briefly in [148], whence its omission here. Fig. 4.6 shows the resulting step response plots. We can especially encounter the structural robustness w.r.t. the parameters q_2 and A_2. Although the dynamic and steady-state behavior changes for the different instances of $\boldsymbol{\theta}$, the structure $\mathbf{G}_{\mathrm{w}2}^{*}$ is maintained for all $\boldsymbol{\theta} \in \Theta$. Interestingly, the dynamics of the signal path $w_1 \to y_1$ is not at all affected by the investigated parameter changes.

Let us briefly demonstrate the control design of a non-compatible parameter set Θ, i.e. the situation when the multiple-plant constraints (3.5a) are not solvable. In this example, we choose to



(a) Output evolution. Solid lines (y_1, w_1), dashed lines (y_2, w_2), dash-dotted lines (y_3, w_3).

(b) Input evolution.

Figure 4.6: Output and input evolution for the multiple-plant design with Eq. (3.5a) solvable.

approximate them via Eq. (3.24a). The parameters are now collected in the set

$$\Theta = \left\{ \operatorname{diag}\left(\begin{bmatrix} \mathbf{1}_3^\top \cdot (1+\lambda) & \mathbf{1}_6^\top & 1+\lambda & \mathbf{1}_2^\top \end{bmatrix} \right) \boldsymbol{\theta}_0 \;\middle|\; \lambda \in \frac{1}{50}\mathcal{I}_{-4,4} \right\}.$$

It turns out that it is still possible to stabilize all plant instances. Using the same optimization settings as for computing the controller (4.25), we obtain the new control law parameterized as

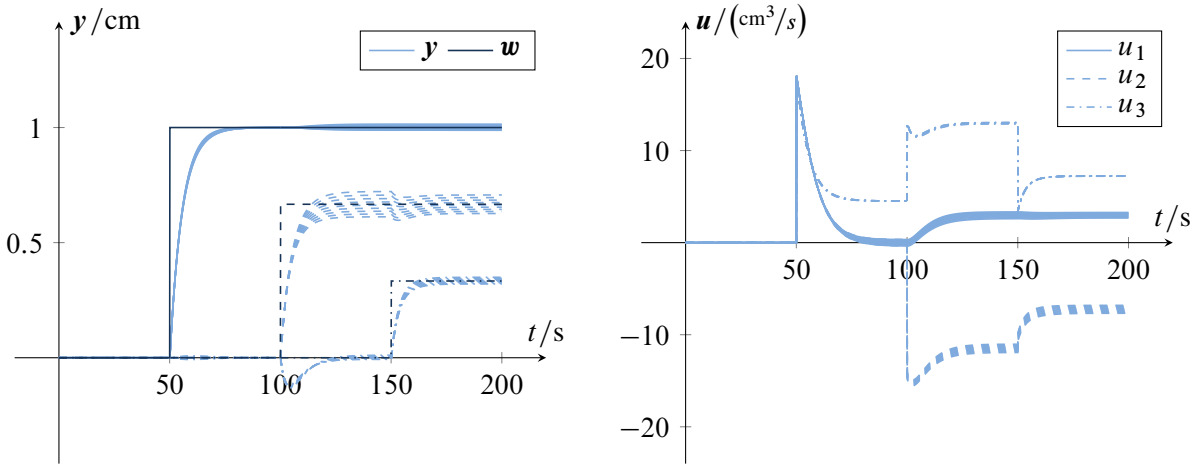
$$\mathbf{F}_y = \begin{bmatrix} -9.80 & -9.04 & 0 \\ 8.62 & -9.04 & -18.41 \\ 4.12 & 0 & -18.44 \end{bmatrix} \quad \text{and} \quad \mathbf{L} = \begin{bmatrix} 18.83 & 0 & 0 \\ 18.83 & -21.73 & -5.69 \\ 18.83 & 12.68 & -35.67 \end{bmatrix}. \tag{4.27}$$

The closed-loop eigenvalues are depicted in Fig. 4.5b. The changes are minor w.r.t. the controller designed before. Also, note that, in this specific case, the structure of $\mathbf{F}_y$ and $\mathbf{L}$ are unchanged w.r.t. (4.25) and (4.26). The lack of structural robustness of the transfer structure $\mathbf{G}_{w2}^*$ becomes apparent in view of the output evolution depicted in Fig. 4.7. The rejection of any signal transmission from w_1 to y_2, and y_3, respectively, as well as from w_2 to y_1 can no longer be maintained exactly. This can be see from the small deflections of the output signals caused by the respective reference steps in w_1 and w_2.

4.4.3 Comparison of LQRLC and Pole Region Assignment

Let us briefly compare and summarize advantages and disadvantages of the methods encountered during their application.

With LQRLC, we were not capable of designing the structured controller as intended. The reason for this lies, however, not in the method itself. Rather, no structured controller optimal in the sense of Eq. (4.5) exists for our showcase. This situation could be identified systematically checking the



(a) Output evolution. Solid lines (y_1, w_1), dashed lines (y_2, w_2), dash-dotted lines (y_3, w_3).

(b) Input evolution.

Figure 4.7: Output and input evolution for the multiple-plant design with Eq. (3.5a) approximated.

existence conditions from Proposition 3. Remedy, i.e. falling back to a full state feedback, could therefore be provided instantly. In the following, the method turned out to enable the parameterization without any knowledge about assignable eigenvalues. Instead, the input weights could be chosen for shaping the optimal control problem in a desired way. For the investigated example, the state weighting matrix $\mathbf{Q}$ turned out to be highly structured. This shows that, depending on the problem at hand, it might be difficult to have good access to the state weights during this type of LQR design.

Pole region assignment was used for finding controllers solving simultaneous stabilization problems for multiple-plant parameterizations. Since the parametric optimization is directly performed over the controller parameters, this method is well suitable for integrating constraints of the form (3.5). This also becomes visible by the fact that, with pole region assignment, we are able to include the originally intended output feedback law. Not knowing a feasible eigenvalue region Γ did not prove to be challenging for finding a solution. In critical cases, it is still an option to iteratively adjust Γ until a feasible region is found. The main challenge of this method lies in its implementation. The optimization of eigenvalues in dependence of the controller parameters is a nonlinear problem that has to be accounted for during implementation. Especially the provision of gradients and Hessians is involved, see [165].

4.5 Summary

After the computation of a control law structure suitable to realize a given closed-loop transfer structure $\mathbf{G}_{\mathrm{w}}^{*}$ in Chapter 3, the final parameterization of such control laws is covered in this chapter. Two main methods are described, i.e. pole region assignment as well as a linear quadratic design procedure obeying the linear controller constraints, named LQRLC.

Pole region assignment itself is a well-established control design method based on a parametric optimization problem in the controller parameters. Since the controller constraints arising from $\mathbf{G}_{\mathrm{w}}^{*}$ are linear equations of the form (3.5a), they can easily be integrated in the design process. Corresponding results can be reviewed, e.g. in [165]. Hence, for this parameterization technique, it is most challenging to establish Eq. (3.5a) at first, which has been covered in detail in Chapter 3. Accordingly, Section 4.1.2 contains only a brief general overview over the design method itself, along with a discussion about a stability proof for the intersampled systems. Intersampled systems occur if a parameter is genuinely assumed to attain uncertain values within a given interval, but only a finite number of parameter instances are taken into account for solving the stabilization problem. Answers to such questions can be given relying on the well-known Bauer-Fike Theorem [33].

Although only covered briefly in this thesis, pole region assignment is the method the structured control approach presented in Chapter 3 is originally designed for. As a result, structured control design is cleanly divided into a compact two-step method consisting of first finding the necessary controller constraints and then feeding the remaining degrees of freedom into the parameter optimization for finalizing the dynamics design. Providing as design variables the closed-loop structure $\mathbf{G}_{\mathrm{w}}^{*}$, the steady-state gains $\mathbf{G}_{\mathrm{w}}^{\infty}$, as well as a suitable pole region Γ, the design can well be automated enabling for a generally quick and effortless control design.

Linear quadratic regulator design subject to arbitrary linear controller constraints is, furthermore, developed in this chapter. Abbreviated LQRLC, the main motivation for this research is the fact that LQR is a widely-spread control design technique providing a good intuition to the designer about how to choose available degrees of freedom, i.e. classically the weighting matrices $\mathbf{Q} \succ 0$ and $\mathbf{R} \succ 0$. Hence, it is of interest whether this method can be extended such that it can parameterize structured controllers.

The map from given matrices $\mathbf{Q}$ and $\mathbf{R}$ in the standard problem (4.5) to a controller $\mathbf{F}_{\boldsymbol{x}}$ is not bijective. Therefore, although LQR will always create a stabilizing[4] $\mathbf{F}_{\boldsymbol{x}}$ for given matrices $\mathbf{Q}, \mathbf{R} \succ 0$, not every controller $\mathbf{F}_{\boldsymbol{x}} \in \mathbb{R}^{m \times n}$ relates to a set of positive definite weights $\mathbf{Q}, \mathbf{R}$. In short, not every controller $\mathbf{F}_{\boldsymbol{x}}$, and, therefore, not every structured controller, is optimal. This discussion reveals that LQRLC is equivalent to an inverse optimal control problem since the main question is whether a given controller structure is optimal w.r.t. Eq. (4.5), and how the respective matrix pair $\mathbf{Q}$, $\mathbf{R}$ is itself structured. Necessary solvability conditions are given in Proposition 3 and, in different form, Proposition 4.

A key point of the following constructive procedure is an admissible choice of the input weights in $\mathbf{R}$. The corresponding results are based on existing literature, mainly [68], where fully parameterized controllers are fed into the inverse optimal control problem. An extensive discussion is conducted about the available degrees of freedom at this stage of the design problem. The respective results are gathered in the Propositions 5 and 6. In the sequel, the controller constraints are transformed via Eq. (4.6) into constraints in the Riccati matrix $\mathbf{P}$. By solving these constraints, the structures of $\mathbf{P}$ and, via Eq. (4.7), of $\mathbf{Q}$ are known. Depending on the specific design problem, there

[4] Assuming stabilizability of the pair $\mathbf{A}$, $\mathbf{B}$

exist degrees of freedom $\boldsymbol{q}_\mathrm{P}$ in their final choice. They must establish positive definiteness of $\mathbf{P}$ and $\mathbf{Q}$. The existence of some $\mathbf{Q} \succ 0$ is, unfortunately, not guaranteed, whence the non-sufficiency of the Propositions 3 and 4. The use of a bisection algorithm, as briefly outlined in Section A.2, is proposed for support in finding admissible values of $\boldsymbol{q}_\mathrm{P}$.

LQRLC enables to finalize structured control design by means of optimal control techniques. It can especially be beneficial if feasible pole regions that could be used for pole region assignment are unknown. Furthermore, there exists the possibility to influence input and state weights directly, which might add a better intuition to the control design. How freely $\mathbf{Q}$ and $\mathbf{R}$ can be chosen varies with the plant model at hand as well as the imposed controller constraints. In the example given in this chapter, difficulties are encountered for intuitively choosing $\boldsymbol{q}_\mathrm{P}$ because $\mathbf{Q}$ is heavily structured. In the examples given in [146], this choice is simpler due to a less heavily structured problem.

5 Summary & Outlook

5.1 Summary

This thesis presents two areas of research. Chapter 2 contains the development of a novel method for characterizing redundancy properties of dynamic systems. In the Chapters 3 and 4, we focus on control design such that a given closed-loop transfer behavior can be achieved by a linear state or output feedback law. Both fields presented in the thesis are connected by the initially coordinate free, i.e. set-based, approach they are founded on. Analyses of the system behavior in non-standard situations are enabled. On the one hand, closed-loop transfer structures can be analyzed for their robustness w.r.t. uncertainties of the plant parameters. On the other hand, with redundancy analysis, statements can be obtained which points in the output space of the system can be reached in worst-case fault scenarios.

The main contributions are twofold. The novel redundancy analysis mechanism enables a systematic view on the abilities of a dynamic system under worst-case conditions. This is and will be an important element of system design, especially in view of the rising demand for fully automated, safety-critical applications. The knowledge gathered about the system can furthermore be used during operation where safe areas of the operation range can explicitly be favored. Equally important, the thesis makes a contribution on the field of structured control design. Providing a well-structured two-step design procedure, it can not only be used for investigations regarding parameter uncertainties. It also contributes to highly automated control design. This is due to the fact that, in the derivations of the control law structures, all occurring degrees of freedom are preserved for the dynamics design step. Especially the combination with pole region assignment provides a powerful option for a nearly autonomous design of structured controllers.

Redundancy Analysis Chapter 2 covers the assessment of a control system w.r.t. basic safety properties. It therefore covers the first research question posed in Section 1.1. In opposition to investigating *resilience*, i.e. the ability of a system to be unaffected of fault scenarios [25], our aim is to describe the worst-case abilities of the control system. In view of modern automated systems, often unmanned, such a *redundancy* analysis is useful for verifying a plant's safety in case of fault scenarios. This is because it can explicitly be proven that critical system quantities, e.g. temperatures or pressures, can be kept in safe operating regions under all considered circumstances.

The cornerstone of the chapter is laid by merging common intuitions about system redundancy, see e.g. [24], and existing system theoretic approaches, best summarized by Kreiss and Trégouët [89]. Doing so, a definition is formed that extends the present and is applicable to a wider class of systems, i.e. nonlinear systems with state and input constraints. Furthermore, it allows for consid-

eration of arbitrary fault scenarios differing from ordinary actuator losses. The connection of the newly developed and prior redundancy definitions is presented in detail in the thesis.

Based on the results thus far, it is possible to investigate a dynamic system for the presence of redundant actuators (Definition 5), or its redundancy w.r.t. a set of given fault scenarios (Definition 6). The next step is comprised of answering the question of how much system maneuverability is left despite occurrence of a fault scenario. Reachability techniques are applied to this end and two types of redundantly reachable sets are defined. One of them offers invariance properties. That is, it describes a tube of points in the output space that the system's outputs can be kept in for all times by applying an admissible control input, independently of the specific fault event. Based on these two types of redundantly reachable sets, an analysis can be conducted on whether the priorly mentioned critical system quantities can be kept in safe operating regions. At the same time, it can be identified if the system lacks redundancy features. Conclusions can be drawn for the choice of additional actuators, altered actuator limits, or other system components to maintain a minimum functionality.

A major challenge in computing redundantly reachable sets is posed by the consideration of infinitely many fault scenarios. These occur especially when taking into account actuators being stuck at an unknown position within their limits. As the main ingredients of each redundantly reachable set are the reachable set of the control system subject to each single fault scenario, it is not feasible to compute them if infinitely many fault scenarios are present. The consequences of this situation are discussed in detail in Section 2.2.2 and system classes are presented for which the computation can be simplified without inducing errors on the result. In these cases, only a finite number of specific fault scenarios needs to be taken into account as they define the shape of the redundantly reachable set. For any other system not belonging to the respective system class, a sampling approach is proposed, inevitably causing computation errors.

As supporting tools to redundancy analysis, two further elements are introduced. A redundancy measure composed of a tuple of scalars is proposed. The measure indicates the loss in volume as well as the loss in dimension of the redundantly reachable set w.r.t. the reachable set of the system under nominal conditions. The main benefit is that reachability information is condensed to two numbers that can be interpreted simpler than the reachable sets itself. The latter may be of high dimensions, depending on the system dimensions and its configuration. Furthermore, conditions are presented for testing a reference point in the output space for redundant stabilizability. That is, it is checked whether this reference point can be reached under all considered circumstances as well as being kept at this position.

Structured Control Design Chapter 3 serves two purposes: control design itself and enabling a robustness analysis of a given closed-loop transfer structure (see e.g. Eq. (3.1)). It therefore covers the research questions posed in Section 1.3. A transfer structure is considered robust w.r.t. uncertainties of plant parameters if the structure is maintained by a time-invariant control law despite varying parameters. The robustness analysis can be conducted during a constructive control design

procedure. Because of this, all results on structural analysis are embedded in the control design steps.

The approach to structured control design is based on a separation of structure design and dynamics design. This separation is conducted by application of geometric concepts as control invariant and conditioned invariant subspaces. It delivers a set of algebraic constraints that must be fulfilled by the controller and prefilter parameters of a time-invariant linear state or output feedback law. That is, a control law structure is derived which is required for realizing a given closed-loop transfer structure.

Besides the presentation of the underlying ideas and geometric concepts, an extension for robustifying the control design against parameter uncertainties is discussed in Section 3.3.2. Here, the algebraic constraints in the control law parameters are computed for a family of plants arising from variations in the plant parameters. Assuming a sufficiently large number of such parameterizations, structural robustness is present if the set of algebraic constraints remains solvable.

For cases that are not structurally robust, techniques are presented and discussed to approximate the closed-loop transfer structure. To this end, least squares approximations of the algebraic constraints are used. These can be applied in various ways and single elements of the transfer structure can be prioritized in this step if desired. The chapter closes with a discussion on the stabilizability of the plant under the structured controller. Methods for verifying stabilizability are presented for different types of controller structures. For structures purely establishing invariance of a single control invariant subspace, results can be adapted from the standard geometric approach literature. For mixed structures, i.e. structures rendering multiple subspaces control invariant, embodying output feedback, or decentralized feedback laws, respectively, these results are not applicable. However, criteria like Theorem 11 by Liénard and Chipart can be applied for verifying stabilizability in such cases.

Parameterization of Structured Controllers Performing structured control design in the way it is presented in Chapter 3 demands for suitable algorithms for populating the remaining degrees of freedom in the control law structure. As a point of entry into the matter, basic pole placement subject to controller constraints is shortly discussed as well as pole region assignment under constraints. The latter is well-suited to be combined with the design of structured controllers and embodies the method that the approach from Chapter 3 has originally be designed for. Since this combination has already been covered in detail in [148], the main focus of Chapter 4 is the detailed presentation of a parameterization strategy based on linear quadratic regulator design explicitly taking into account the required controller structure.

Instead of solving a constrained optimal control problem leading to a sub-optimal controller in the sense of the standard LQR problem (4.5), the idea is to choose the objective function of (4.5) such that the constraints are automatically satisfied. To this end, an inverse optimal control problem is formulated in terms of the structured controller. The weighting matrices of the objective function are the variables sought for. Although many results can be adopted from the literature, the controller only being present in its parametric form creates additional degrees of freedom in

the design process. How they can be chosen is discussed in detail. Their choice has an effect on the input weights of the LQR problem that can be obtained. Therefore, this is an important step.

Having fixed the input weights, the next step is to determine the structure of the Riccati matrix imposed by the controller structure. From this, the structure of the state weights of the objective function follows via the Riccati equation. Again, parameters must be chosen that render both, the Riccati matrix and the state weights, positive definite. Estimating the set of admissible parameters is proposed to be done by solving a set of polynomial inequalities by a bisection algorithm.

5.2 Open Research Questions

Redundancy Analysis The redundancy analysis procedure developed in Chapter 2 is generally applicable to all standard nonlinear systems of the form (2.1). No theoretic difficulties are induced if the set of fault scenarios has finitely many elements. However, for the prominent case of actuators stuck at an unknown position, i.e. a case with infinitely many fault scenarios, the analysis is hard to be conducted for such systems. Errors will be induced by the simplifications that can validly be made for, e.g., linear systems. This is due to the conditions of Theorem 5, i.e. the requirement of monotonicity properties and convex reachable sets for all times. It would be a major improvement to find ways to relax these conditions such that an error-less analysis is enabled for constant actuator faults, also for nonlinear systems. In order for this to be done, it is expected that gradients of reachable sets are required to be known. Using this information, a set of extremal reachable sets could potentially be identified that are crucial for shaping the redundantly reachable set.

A direct integration of the proposed redundancy analysis into system design is naturally possible by evaluating the redundancy properties of various system *candidates* possessing, e.g., variations of actuator configurations. The analysis results can be compared and the best performing candidate can be drawn. The performance could be measured, e.g., by the redundancy measure presented in Section 2.2.4. A better-directed search, possibly a gradient-based optimization procedure, for an optimal candidate is of interest at this point. In this way, the system designer could be aided in obtaining suitable system candidates instead of only comparing predefined candidates. This could help to identify and create awareness of degrees of freedom that can be used for improving the system's safety properties.

For linear unconstrained systems, an alternative to the priorly mentioned redundancy measure could be based on Eq. (2.8). There, we have associated a redundant actuator u_i, $i \in \mathcal{I}_{1,m}$, with a non-zero kernel $\boldsymbol{\alpha}_i^\top \mathbf{P}_2^\perp(s)$. If this kernel exists, it is possible to set $u_i = 0$ without rendering any tuple $(\boldsymbol{x}_0, \boldsymbol{y}) \in \mathcal{S}$ infeasible. Following this idea, we could measure how much effort is needed to compensate the given actuator u_i by allocating its nominal control action to the other actuators $u_j, i \neq j$. This measurement can be based on the relative increase in the overall control effort w.r.t. the nominal case, i.e. viewing some vector norm of the control signal in the fault case and the nominal case, respectively. Similarly, the norm of the degrees of freedom $\boldsymbol{q}(s)$ could be used. However, in the latter case, the physical interpretability of the redundancy measure might be

restricted. Challenges in the design of a measure based on the above arguments are how to choose $\boldsymbol{y}_{\mathrm{s}}$ and $\boldsymbol{x}_0$ in Eq. (2.8). Furthermore, the dependence of Eq. (2.8) on the Laplace variable $s \in \mathbb{C}$ must be handled. For example, viewing the limit $s \to 0$ restricts all statements to a steady state of the system. In contrast, this fosters interpretability.

Structured Control Design Solvability of the structured control problem posed in Definition 20 is given if and only if there exists a set of compatible control invariant subspaces fulfilling the stated conditions. In the design step described in Section 3.3.1, we do not search explicitly for such a compatible set. Rather, we use the *maximal* control invariant subspaces for each column of the transfer structure. Doing so, solvability of the resulting equality constraints is only sufficient for the problem to be solvable. If they are not solvable, it is unclear whether there exists a compatible set of subspaces. In Section 3.4, the feasibility problem (3.22) is presented that encodes the explicit search for such a set. Theory exists for optimizing over the manifold of orthonormal bases which occur in this problem. However, the problem is still challenging since the dimensions of the involved bases is a priori unknown and as well a decision variable. This case is not covered by the literature. There is a finite set of combinations of dimensions for the set of subspaces. However, depending on the system dimensions, solving the feasibility problem (3.22) for all possible combinations can result in a large computational effort. Theoretical results concerning this issue are, therefore, of interest.

LQRLC An initial parameterization of the structured controller is needed to start the LQRLC procedure. The Propositions 3 and 4 describe the influence of this initial parameterization on the degrees of freedom available for an admissible choice of input weights. In their view, it becomes clear that there exist systems and controller structures where some degrees of freedom are lost due to the choice of this initial parameterization. It would be beneficial to find a way how to avoid this loss at this early stage in the design procedure. Thereby, a complete characterization of the whole set of admissible input weights could be created. The advantage is obvious. The design problem becomes more transparent enabling for a more refined control design. A desirable way to achieve this could be adapted from Ko and Bitmead [82] where the structured control design problem is turned into an unstructured one by a suitable system transformation.

Appendix

A.1 Optimal Control Scheme

In the following, we document the optimal control scheme that is used for producing the results from Section 2.2.5.3 as well as Section 2.3.2. For the sake of legibility, we omit the indices z indicating the fault scenario for the states, inputs, and slack variables. The scheme used for obtaining the results from Fig. 2.8a is given by

$$\min_{\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{s}_k} \quad \sum_{k=0}^{N-1} \boldsymbol{e}_k^\top \mathbf{T}_{\boldsymbol{e}} \boldsymbol{e}_k + \boldsymbol{u}_k^\top \mathbf{T}_{\boldsymbol{u}} \boldsymbol{u}_k + \boldsymbol{s}_k^\top \mathbf{T}_{\boldsymbol{s},k} \boldsymbol{s}_k \tag{A.1a}$$

$$\text{s.t.} \quad \boldsymbol{x}_{k+1} = \bar{\mathbf{A}}_{z,k} \boldsymbol{x}_k + \bar{\mathbf{B}}_{z,k} \boldsymbol{u}_k, \qquad k \in \mathcal{I}_{0,N-1} \tag{A.1b}$$

$$\boldsymbol{x}_0 = \boldsymbol{x}(0) \tag{A.1c}$$

$$\bar{\mathbf{C}}_{z,N} \boldsymbol{x}_N = \boldsymbol{y}_{\text{ref}} \tag{A.1d}$$

$$\mathbf{H}_{z,k}^{\boldsymbol{x}} \boldsymbol{x}_k \leq \boldsymbol{h}_{z,k}^{\boldsymbol{x}}, \qquad k \in \mathcal{I}_{0,N} \tag{A.1e}$$

$$\mathbf{H}_{z,k}^{\boldsymbol{u}} \boldsymbol{u}_k \leq \boldsymbol{h}_{z,k}^{\boldsymbol{u}}, \qquad k \in \mathcal{I}_{0,N-1} \tag{A.1f}$$

$$\mathbf{H}_k^{\mathcal{Q}} \bar{\mathbf{C}}_{z,k} \boldsymbol{x}_k \leq \boldsymbol{h}_k^{\mathcal{Q}} + \boldsymbol{s}_k \qquad k \in \mathcal{I}_{0,N}. \tag{A.1g}$$

We have used the abbreviation $\boldsymbol{e}_k := \left(\boldsymbol{y}_{\text{ref}} - \bar{\mathbf{C}}_{z,k} \boldsymbol{x}_k\right)$ and will denote $n_{s,k} := \dim(\boldsymbol{s}_k)$ in the sequel. We have adapted the nomenclature for the state and input constraints (A.1e) and (A.1f) to Eq. (2.18). Analogously, the time-varying constraints (A.1g) are adapted to Eq. (2.22). For the construction of the results for Fig. 2.8b, they are replaced by

$$\mathbf{H}_k^{\hat{\mathcal{Q}}} \bar{\mathbf{C}}_{z,k} \boldsymbol{x}_k \leq \boldsymbol{h}_k^{\hat{\mathcal{Q}}} \qquad k \in \mathcal{I}_{0,N},$$

where no slack variables are needed ($n_{s,k} = 0$ for all $k \in \mathcal{I}_{0,N}$). The numeric values used for computation are

$$\mathbf{T}_{\boldsymbol{e}} = 10^3 \mathbf{I}_p, \qquad \mathbf{T}_{\boldsymbol{u}} = \mathbf{I}_m, \qquad \mathbf{T}_{\boldsymbol{s},k} = 10^9 \mathbf{I}_{n_{s,k}}.$$

For Section 2.3.2, no slack variables are used since we want to validate that the formulated OCP is feasible. Furthermore, the constraints (A.1d) and (A.1g) are replaced by $\boldsymbol{x}_k \in \mathcal{V}_z^*$ for all $k \in \mathcal{I}_{k_s,N}$. This way, the reference point $\boldsymbol{y}_{\text{ref}}$ is forced to be reached from $k = k_s$ onward. The weighting matrices are chosen as $\mathbf{T}_{\boldsymbol{e}} = \mathbf{I}_p$ and $\mathbf{T}_{\boldsymbol{u}} = \mathbf{I}_m$. The control invariant sets $\mathcal{V}_z^*$ for this example are given by the line segments $\mathcal{V}_{\underline{u}_3}^* = \{\boldsymbol{x} \in \mathbb{R}^3 \,|\, 0.16 \leq x_1 \leq 0.72,\ x_2 = 0.3,\ x_3 = 0.2\}$ for the fault scenario $u_3 = \underline{u}_3$ and $\mathcal{V}_{\overline{u}_3}^* = \{\boldsymbol{x} \in \mathbb{R}^3 \,|\, 0.04 \leq x_1 \leq 0.67,\ x_2 = 0.3,\ x_3 = 0.2\}$ for $u_3 = \overline{u}_3$.

Remark 17. Note that, in both cases, $p < n$. Therefore, not all states are penalized in the stage cost and it might be expected that this causes instability of the non-penalized states. Due to Raković and Levine [136, p. 8], it is, however, sufficient that the underlying dynamic system is observable and the prediction horizon N exceeds the number of states n. This is the case in our setup. Hence, obtaining a stabilizing solution by solving Eq. (A.1) is to be expected.

A.2 Bernstein Algorithm

In this thesis, there are two problems, i.e. Eqs. (3.35) and (4.17), that involve the solution of systems of polynomial inequalities having the form

$$\boldsymbol{\pi}(\boldsymbol{q}) > \mathbf{0} \tag{A.2}$$

with $\boldsymbol{\pi}(\boldsymbol{q}) \in \mathbb{R}^{n_\pi}$. Here, $\boldsymbol{q} = [q_1 \quad \cdots \quad q_{n_\mathrm{q}}]^\top \in \mathbb{R}^{n_\mathrm{q}}$ denotes the free parameters of the polynomials $\pi_i(\boldsymbol{q})$, $i \in \mathcal{I}_{1,n_\pi}$. More specifically, in the mentioned problems, we are interested in finding the largest set $\mathcal{Q}_\pi \subseteq \mathbb{R}^{n_\pi}$ such that (A.2) holds for all $\boldsymbol{q} \in \mathcal{Q}_\pi$, i.e., we want to find the solution set of (A.2), or at least an approximation $\mathcal{Q}_\pi^{\approx}$ of it. This is in contrast to finding one feasible point $\boldsymbol{q} \in \mathbb{R}^{n_\pi}$ solving the problem. For the latter, we could set up a (nonlinear) feasibility problem and make use of, e.g., gradient based optimization methods.

Let us outline the idea of the presented algorithm that is due to Garloff and Graf [51] for the simple case $n_\mathrm{q} = n_\pi = 1$, i.e. the scalar case with $\pi := \boldsymbol{\pi} = \pi_1$ and $q := \boldsymbol{q} = q_1$, respectively. This case is not conceptually simpler, but allows for a simpler presentation without the need of introducing special notation. Any scalar polynomial

$$\pi(q) = \sum_{j=0}^{k} a_j q^j,$$

$a_j \in \mathbb{R}$, can be written as

$$\pi(q) \equiv \sum_{j=0}^{k} b_j B_{j,k}(q) \tag{A.3}$$

with $k \in \mathbb{N}$ as the degree of the polynomial π, where $b_j \in \mathbb{R}$ denote so-called Bernstein coefficients, while the Bernstein polynomials are defined as

$$B_{j,k}(q) := \binom{k}{j} q^j (1-q)^{k-j} \tag{A.4}$$

with the binomial coefficient

$$\binom{k}{j} = \frac{k!}{j!(k-j)!}.$$

The Bernstein polynomials establish a basis of linearly independent polynomials of degree k, whence the possibility to establish the equivalence (A.3). Furthermore, it can easily be verified that all Bernstein polynomials (A.4) have the property $B_{j,k}(q) > 0$ for all free parameters $q \in (0;\, 1)$. Following [51], the Bernstein coefficients are given by

$$b_j = \sum_{\ell=0}^{j} \frac{\binom{j}{\ell}}{\binom{k}{\ell}} a_\ell.$$

These coefficients can be used for checking the satisfaction of (A.2) on the open interval (0; 1). Sufficiently, if all $b_j > 0$, $j \in \mathcal{I}_{0,k}$, then (A.2) is fulfilled by positivity of the Bernstein polynomials (A.4).

The investigation of $\pi(q)$ being positive is not restricted to parameters $q \in (0; 1)$. By applying the transformation

$$\begin{aligned} q &= (1 - \tilde{q})\underline{q} + \tilde{q}\overline{q} \\ &= \underline{q} + (\overline{q} - \underline{q})\tilde{q} \end{aligned}$$

with $\tilde{q} \in (0; 1)$, we can easily extend the results to parameters q from the open interval $(\underline{q}; \overline{q})$. In this case, the polynomial at hand transforms as

$$\begin{aligned} \tilde{\pi}(\tilde{q}) &= \pi\Big(\underline{q} + (\overline{q} - \underline{q})\tilde{q}\Big) \\ &= \sum_{j=0}^{k} a_j \Big(\underline{q} + (\overline{q} - \underline{q})\tilde{q}\Big)^j \end{aligned}$$

which can be expressed as [11, p. 77]

$$\begin{aligned} \tilde{\pi}(\tilde{q}) &= \sum_{j=0}^{k} a_j \sum_{\ell=0}^{j} \binom{j}{\ell} \underline{q}^{j-\ell} (\overline{q} - \underline{q})^{\ell} \tilde{q}^{\ell} \\ &= \sum_{j=0}^{k} a_j \left[\underbrace{\binom{j}{0} \underline{q}^{j} (\overline{q} - \underline{q})^{0} \tilde{q}^{0}}_{=\underline{q}^{j}} + \cdots + \binom{j}{\ell^*} \underline{q}^{j-\ell^*} (\overline{q} - \underline{q})^{\ell^*} \tilde{q}^{\ell^*} + \cdots + \underbrace{\binom{j}{j} \underline{q}^{0} (\overline{q} - \underline{q})^{j} \tilde{q}^{j}}_{=(\overline{q}-\underline{q})^{j}\tilde{q}^{j}} \right] \end{aligned}$$

for some $\ell^* \leq j$. This form can be used to obtain the coefficients $\tilde{a}_{\ell^*}$ of the transformed polynomial

$$\tilde{\pi}(\tilde{q}) = \sum_{\ell^*=0}^{k} \tilde{a}_{\ell^*} \tilde{q}^{\ell^*}$$

by collecting all terms of power l^*. We have

$$\tilde{a}_{\ell^*} = \sum_{j=\ell^*}^{k} a_j \binom{j}{\ell^*} \underline{q}^{j-\ell^*} (\overline{q} - \underline{q})^{\ell^*}.$$

Checking $\pi(q) > 0$ on the given interval $(\underline{q}; \overline{q})$ is finally done according to the steps described in Algorithm 5, where the bisection of the interval $(\underline{q}; \overline{q})$ with center q_c is defined as creating two sub-intervals $(\underline{q}; q_c)$ and $(q_c; \overline{q})$. The parameter $d \geq 0$ denotes the bisection depth, i.e. the number of bisections allowed to be performed during the procedure. For $d \to \infty$, the output $\mathcal{Q}_\pi^{\approx}$ of the algorithm converges to the true solution set $\mathcal{Q}_\pi$ that we ultimately aim at finding.

The generalization of the presented procedure to the multi-variate case with $n_q > 1$ is achieved by using a multi-variate form of Bernstein polynomials and Bernstein coefficients whose definitions

Algorithm 5 Bernstein Algorithm for $n_\pi = n_\mathrm{q} = 1$.

1: **Input:** $(\underline{q};\, \overline{q})$, $\pi(q)$, d
2: $\mathcal{Q}_\pi^{\{\}} \leftarrow \{\}$
3: $i \leftarrow 0$
4: $\mathcal{Q}_0^{\{\}} \leftarrow \{\}$
5: $\mathcal{Q}_0 \leftarrow (\underline{q};\, \overline{q})$
6: Append $\mathcal{Q}_0^{\{\}}$ by $\mathcal{Q}_0$
7: **while** $i \leq d$ **do**
8: $\quad \mathcal{Q}_{i+1}^{\{\}} \leftarrow \{\}$
9: $\quad$ **for all** $\mathcal{Q} \in \mathcal{Q}_i^{\{\}}$ **do**
10: $\quad\quad$ Compute $\tilde{\pi}(\tilde{q})$ for $\mathcal{Q}$
11: $\quad\quad$ Compute Bernstein coefficients b_j for $\tilde{\pi}(\tilde{q})$
12: $\quad\quad$ **if** $b_j > 0 \;\forall\; j \in \mathcal{I}_{0,k}$ **then**
13: $\quad\quad\quad$ Append $\mathcal{Q}_\pi^{\{\}}$ by $\mathcal{Q}$
14: $\quad\quad$ **else**
15: $\quad\quad\quad$ Bisect $\mathcal{Q}$ into $\mathcal{Q}_1$ and $\mathcal{Q}_2$
16: $\quad\quad\quad$ Append $\mathcal{Q}_{i+1}^{\{\}}$ by $\mathcal{Q}_1$ and $\mathcal{Q}_2$
17: $\quad\quad$ **end if**
18: $\quad$ **end for**
19: $\quad i \leftarrow i + 1$
20: **end while**
21: $\mathcal{Q}_\pi^{\approx} \leftarrow \bigcup_{\mathcal{Q} \in \mathcal{Q}_\pi^{\{\}}} \mathcal{Q}$
22: **Output:** $\mathcal{Q}_\pi^{\approx}$

can be reviewed in [51]. Since their use requires introduction of a complete multi-index notation, they are not presented here. For $n_\mathrm{q} > 1$, the evaluated parameter interval is extended to a multi-dimensional box $(0;\, 1)^{n_\mathrm{q}}$ which has an impact on the bisection step in Algorithm 5, line 15. Here, a number of 2^{n_q} sub-boxes is created due to the bisection. The adaption to multiple polynomials, i.e. $n_\pi > 1$, is rather straight-forward, as we only need to check positivity of all Bernstein coefficients for all $i \in \mathcal{I}_{1,n_\pi}$.

A.3 Verifying Conditioned Invariance

Theorem 13 (see Basile and Marro [17, Def. 4.1.2], Trentelman [160, Th. 5.5])**.** *Given a system of the form* (3.2)*, a subspace $\mathcal{V} \subseteq \mathcal{X}$ is conditioned invariant if and only if*

$$\mathbf{A}(\mathcal{V} \cap \ker(\mathbf{C})) \subseteq \mathcal{V}$$

holds.

A.4 Rewriting Controller Constraints

For the use with LQRLC, we reformulate the controller constraints (3.5a) in terms of the state controller parameters $\boldsymbol{f}_{\boldsymbol{x}}$ because the design problem is formulated based on these parameters. Let us outline the necessary transformation briefly. We assume

$$\mathbf{F}_{\boldsymbol{x}} = \mathbf{F}_{\boldsymbol{y}}\mathbf{C}_{\mathrm{m}} \tag{A.5}$$

with $\mathbf{C}_{\mathrm{m}} \in \mathbb{R}^{p_{\mathrm{m}} \times n}$ and $\operatorname{rank}(\mathbf{C}_{\mathrm{m}}) = p_{\mathrm{m}}$. The vectorization of (A.5) reads $\boldsymbol{f}_{\boldsymbol{x}} = \left(\mathbf{C}_{\mathrm{m}}^{\top} \otimes \mathbf{I}_m\right) \boldsymbol{f}_{\boldsymbol{y}}$, where, due to the assumption, $\operatorname{rank}\left(\mathbf{C}_{\mathrm{m}}^{\top} \otimes \mathbf{I}_m\right) = mp_{\mathrm{m}}$ holds. We can reformulate the relation in terms of a singular value decomposition as

$$\boldsymbol{f}_{\boldsymbol{x}} = \mathbf{U}\mathbf{S}\mathbf{V}^{\top}\boldsymbol{f}_{\boldsymbol{y}}$$

where $\mathbf{S} = \begin{bmatrix}\Sigma & \mathbf{0}\end{bmatrix}^{\top}$ with the diagonal singular value matrix $\Sigma \in \mathbb{R}^{mp_{\mathrm{m}} \times mp_{\mathrm{m}}}$, and the orthonormal matrices $\mathbf{U} \in \mathbb{R}^{mn \times mn}$, $\mathbf{V} \in \mathbb{R}^{mp_{\mathrm{m}} \times mp_{\mathrm{m}}}$. Multiplying $\mathbf{U}^{\top}$ from the left and partitioning $\mathbf{U} = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}$ with $\mathbf{U}_1 \in \mathbb{R}^{mn \times mp_{\mathrm{m}}}$, we can rewrite the above equation as

$$\begin{bmatrix}\mathbf{U}_1^{\top} \\ \mathbf{U}_2^{\top}\end{bmatrix} \boldsymbol{f}_{\boldsymbol{x}} = \begin{bmatrix}\Sigma\mathbf{V}^{\top}\boldsymbol{f}_{\boldsymbol{y}} \\ \mathbf{0}\end{bmatrix}$$

which is equivalent to

$$\begin{bmatrix}\mathbf{V}\Sigma^{-1}\mathbf{U}_1^{\top} \\ \mathbf{U}_2^{\top}\end{bmatrix} \boldsymbol{f}_{\boldsymbol{x}} = \begin{bmatrix}\boldsymbol{f}_{\boldsymbol{y}} \\ \mathbf{0}\end{bmatrix}.$$

From this, it follows that the constraint (3.5a) can be rewritten in terms of $\boldsymbol{f}_{\boldsymbol{x}}$ as $\mathbf{Z}_{\mathrm{F}\boldsymbol{x}}\boldsymbol{f}_{\boldsymbol{x}} = \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}}$ with

$$\mathbf{Z}_{\mathrm{F}\boldsymbol{x}} = \begin{bmatrix}\mathbf{Z}_{\mathrm{F}}\mathbf{V}\Sigma^{-1}\mathbf{U}_1^{\top} \\ \mathbf{U}_2^{\top}\end{bmatrix}, \qquad \boldsymbol{z}_{\mathrm{F}\boldsymbol{x}} = \begin{bmatrix}\boldsymbol{z}_{\mathrm{F}} \\ \mathbf{0}_{m(n-p_{\mathrm{m}})\times 1}\end{bmatrix}.$$

A.5 Construction of Symmetry Constraints

The symmetry constraint set $\mathbf{Z}_{\mathrm{sym}}\boldsymbol{p} = \mathbf{0}$ included in Eq. (4.10) is constructed by extracting all lower triangular elements of the original symmetry constraint equation $\mathbf{P} - \mathbf{P}^{\top} = \mathbf{0}$. The latter is equivalent to

$$\operatorname{vec}(\mathbf{P} - \mathbf{P}^{\top}) = \operatorname{vec}(\mathbf{P}) - \operatorname{vec}(\mathbf{P}^{\top}) = \mathbf{0}.$$

In order to rewrite this in terms of $\boldsymbol{p} = \operatorname{vec}(\mathbf{P})$, we use the commutation matrix [100, Th. 3.1]

$$\mathbf{K}^{(a,b)} := \sum_{i=1}^{a} \left(\boldsymbol{a}_i \otimes \mathbf{I}_b \otimes \boldsymbol{a}_i^{\top}\right) \in \mathbb{R}^{ab \times ab} \tag{A.6}$$

with the canonical unit vector $\boldsymbol{a}_i \in \mathbb{R}^a$ pointing in the i-th coordinate direction. For any matrix $\mathbf{M} \in \mathbb{R}^{a \times b}$, the expression $\operatorname{vec}(\mathbf{M}^{\top}) = \mathbf{K}^{(a,b)}\operatorname{vec}(\mathbf{M})$ holds [100]. Hence, the symmetry constraint reads $(\mathbf{I}_{n^2} - \mathbf{K}^{(n,n)})\boldsymbol{p} = \mathbf{0}$. This notation contains linearly dependent rows, i.e. equations

$p_{ii} - p_{ii} = 0$ as well as equations of the form $p_{ij} - p_{ji} = 0$ with $i < j$. These are removed using a selection matrix

$$\mathcal{S} = \begin{bmatrix} \mathcal{S}_1 & & \\ & \ddots & \\ & & \mathcal{S}_{n-1} \end{bmatrix} \mathbf{0}_{\frac{1}{2}(n^2-n)\times n} \in \mathbb{R}^{\frac{1}{2}(n^2-n)\times n^2}$$

with $\mathcal{S}_i = \begin{bmatrix} \mathbf{0}_{(n-i)\times i} & \mathbf{I}_{n-i} \end{bmatrix} \in \mathbb{R}^{(n-i)\times n}$ and the zero matrix $\mathbf{0}_{a\times b} \in \mathbb{R}^{a\times b}$. The selection matrix is designed such that all elements below the main diagonal are selected from the left side of the initial symmetry constraint $\mathbf{P} - \mathbf{P}^\top = \mathbf{0}$. We finally obtain

$$\mathbf{Z}_{\text{sym}} = \mathcal{S}(\mathbf{I}_{n^2} - \mathbf{K}^{(n,n)}) \in \mathbb{R}^{\frac{1}{2}(n^2-n)\times n^2}$$

with $\text{rank}(\mathbf{Z}_{\text{sym}}) = \frac{1}{2}(n^2 - n)$. The latter results from the fact that we have filtered out all linearly dependent equations.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008. DOI: 10.1515/9781400830244.

[2] J. Ackermann. *Robust Control – Systems with Uncertain Physical Parameters*. Springer, 1993. DOI: 10.1007/978-1-4471-3365-0.

[3] J. Adamy. *Nonlinear Systems and Controls*. 1st ed. Springer Vieweg Berlin, Heidelberg, 2022. ISBN: 978-3-662-65632-7.

[4] M. Althoff. *CORA*. 2023. URL: https://github.com/TUMcps/CORA.

[5] M. Althoff. "Reachability Analysis and its Application to the Safety Assessment of Autonomous Cars". PhD thesis. Technische Universität München, 2010.

[6] M. Althoff, G. Frehse, and A. Girard. "Set Propagation Techniques for Reachability Analysis". In: *Annual Review of Control, Robotics, and Autonomous Systems* 4 (2021), pp. 369–395. DOI: 10.1146/annurev-control-071420-081941.

[7] M. Althoff and B. H. Krogh. "Avoiding geometric intersection operations in reachability analysis of hybrid systems". In: *15th ACM International Conference on Hybrid Systems: Computation and Control*. 2012. DOI: 10.1145/2185632.2185643.

[8] M. Althoff, O. Stursberg, and M. Buss. "Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization". In: *47th IEEE Conference on Decision and Control*. 2008. DOI: 10.1109/cdc.2008.4738704.

[9] D. Angeli and E. D. Sontag. "Monotone control systems". In: *IEEE Transactions on automatic control* 48.10 (2003), pp. 1684–1698.

[10] P. Apkarian and H. D. Tuan. "Parameterized LMIs in Control Theory". In: *SIAM Journal on Control and Optimization* 38.4 (2000), pp. 1241–1264. DOI: 10.1137/S036301299732612X.

[11] T. Arens, F. Hettlich, C. Karpfinger, U. Kockelkorn, K. Lichtenegger, and H. Stachel. *Mathematik*. 3rd ed. Springer Spektrum Berlin, Heidelberg, 2015. DOI: 10.1007/978-3-642-44919-2.

[12] W. F. Arnold and A. J. Laub. "Generalized eigenproblem algorithms and software for algebraic Riccati equations". In: *Proceedings of the IEEE* 72.12 (1984), pp. 1746–1754. DOI: 10.1109/proc.1984.13083.

[13] S. Barnett. "A new formulation of the Liénard-Chipart stability criterion". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 70.2 (1971), pp. 269–274. DOI: 10.1017/S0305004100049872.

[14] G. Basile and G. Marro. “Controlled and conditioned invariant subspaces in linear system theory”. In: *Journal of Optimization Theory and Applications* 3.5 (1969), pp. 306–315. DOI: 10.1007/bf00931370.

[15] G. Basile and G. Marro. “On the robust controlled invariant”. In: *Systems & Control Letters* (1987), pp. 191–195.

[16] G. Basile and G. Marro. “Self-bounded controlled invariant subspaces: A straightforward approach to constrained controllability”. In: *Journal of Optimization Theory and Applications* 38 (1982), pp. 71–81. DOI: 10.1007/bf00934323.

[17] G. Basile and G. Marro. *Controlled and Conditioned Invariants in Linear System Theory*. Englewood Cliffs : Prentice-Hall, 1992. ISBN: 0131729748.

[18] L. Benvenuti and L. Farina. “Positive Dynamical Systems: New Applications, Old Problems”. In: *International Journal of Control, Automation and Systems* 21.3 (2023), pp. 837–844.

[19] S. P. Bhattacharyya. “Generalized Controllability, (A, B)-Invariant Subspaces and Parameter Invariant Control”. In: *SIAM Journal on Algebraic Discrete Methods* 4 (1983), pp. 529–533. DOI: 10.1137/0604053.

[20] F. Borrelli, A. Bemporad, and M. Morari. *Predictive Control for Linear and Hybrid Systems*. 2017. DOI: 10.1017/9781139061759.

[21] N. Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023. DOI: 10.1017/9781009166164.

[22] J.-B. Bouvier and M. Ornik. “Resilient Reachability for Linear Systems”. In: *IFAC PapersOnLine* 53 (2020), pp. 4409–4414. DOI: 10.1016/j.ifacol.2020.12.372.

[23] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Vol. 15. Studies in Applied Mathematics. Philadelphia, PA: SIAM, June 1994.

[24] Britannica. *Definition of Redundancy*. retrieved September 6th, 2023. Sept. 2023. URL: https://www.britannica.com/dictionary/redundancy.

[25] Britannica. *Definition of Resilience*. retrieved September 6th, 2023. Sept. 2023. URL: https://www.britannica.com/dictionary/resilience.

[26] J. M. Buffington. “Modular control law design for the innovative control effectors (ICE) tailless fighter aircraft configuration 101-3”. In: *Ohio Airforce Research Laboratory* (1999).

[27] J. M. Buffington and D. F. Enns. “Lyapunov stability analysis of daisy chain control allocation”. In: *Journal of Guidance, Control, and Dynamics* 19.6 (1996), pp. 1226–1230.

[28] J. M. Buffington, D. F. Enns, and A. R. Teel. “Control allocation and zero dynamics”. In: *Journal of Guidance, Control, and Dynamics* 21.3 (1998), pp. 458–464.

[29] Bundeszentrale für politische Bildung. *Demographischer Wandel*. Nr. 350/2022. Jan. 2022.

[30] A. Casavola and E. Garone. “Fault-tolerant adaptive control allocation schemes for overactuated systems”. In: *International Journal of Robust and Nonlinear Control* 20 (2010), pp. 1958–1980. DOI: 10.1002/rnc.1561.

[31] P. V. Chanekar, N. Chopra, and S. Azarm. “Optimal structured static output feedback design using generalized benders decomposition”. In: *2017 IEEE 56th Conference on Decision and Control*. 2017, pp. 4819–4824.

[32] M. C. Cheng. “General criteria for redundant and nonredundant linear inequalities”. In: *Journal of Optimization Theory and Applications* 53 (1987), pp. 37–42.

[33] E. Chu. “Generalization of the Bauer-Fike Theorem”. In: *Numerische Mathematik* (1986), pp. 685–691.

[34] T. Chu and L. Huang. “Mixed monotone decomposition of dynamical systems with application”. In: *Chinese Science Bulletin* 43 (1998), pp. 1171–1175. DOI: 10.1007/bf02883218.

[35] J. Cieslak, D. Efimov, A. Zolghadri, A. Gheorghe, P. Goupil, and R. Dayre. “A method for actuator lock-in-place failure detection in aircraft control surface servo-loops”. In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 10549–10554.

[36] G. Conte and A. M. Perdon. “Robust Disturbance Deoupling Problem for Parameter Dependent Families of Linear Systems”. In: *Automatica* 29.2 (1993), pp. 475–478.

[37] G. Conte, A. M. Perdon, and G. Marro. “Computing the maximum robust controlled invariant subspace”. In: *Systems & Control Letters* 17 (1991), pp. 131–135. DOI: 10.1016/0167-6911(91)90038-g.

[38] S. Coogan. “Mixed Monotonicity for Reachability and Safety in Dynamical Systems”. In: *59th IEEE Conference on Decision and Control*. 2020. DOI: 10.1109/cdc42340.2020.9304391.

[39] A. Cristofaro, M. M. Polycarpou, and T. A. Johansen. “Fault-Tolerant Control Allocation for Overactuated Nonlinear Systems”. In: *Asian Journal of Control* 20 (2018), pp. 621–634. DOI: 10.1002/asjc.1619.

[40] E. Davison and S. Wang. “Properties of linear time-invariant multivariable systems subject to arbitrary output and state feedback”. In: *IEEE Transactions on Automatic Control* 18.1 (1973), pp. 24–32. DOI: 10.1109/TAC.1973.1100219.

[41] E. J. Davison, W. Gesing, and S. H. Wang. “An algorithm for obtaining the minimal realization of a linear time-invariant system and determining if a system is stabilizable-detectable”. In: *1977 IEEE Conference on Decision and Control including the 16th Symposium on Adaptive Processes and A Special Symposium on Fuzzy Set Theory and Applications*. 1977, pp. 777–782. DOI: 10.1109/CDC.1977.271674.

[42] P. De Leenheer, D. Angeli, and E. D. Sontag. “A tutorial on monotone systems – with an application to chemical reaction networks”. In: *Proc. 16th Int. Symp. Mathematical Theory of Networks and Systems (MTNS)*. 2004, pp. 2965–2970.

[43] Demopædia. *Multilingual Demographic Dictionary*. unified second edition. English volume (retrieved September 6th, 2023). 1982. URL: http://en-ii.demopaedia.org.

[44] M. P. Drazin and E. V. Haynsworth. "Criteria for the reality of matrix eigenvalues." In: *Mathematische Zeitschrift* 78 (1962), pp. 449–452.

[45] W. C. Durham. "Constrained control allocation". In: *Journal of Guidance, Control, and Dynamics* 16 (1993), pp. 717–725. DOI: 10.2514/3.21072.

[46] A. Edelman, T. A. Arias, and S. T. Smith. "The Geometry of Algorithms with Orthogonality Constraints". In: *SIAM Journal on Matrix Analysis and Applications* 20 (1998), pp. 303–353. DOI: 10.1137/s0895479895290954.

[47] V. Ewald and U. Konigorski. "Model-Matching-Control of a Redundantly Actuated Steer-by-Wire-System". In: *2020 IEEE Conference on Control Technology and Applications*. 2020, pp. 194–200. DOI: 10.1109/CCTA41146.2020.9206283.

[48] M. S. Fadali and A. Visioli. "Chapter 8 – Properties of State-Space Models". In: *Digital Control Engineering*. Ed. by M. S. Fadali and A. Visioli. Boston: Academic Press, 2009, pp. 281–333. DOI: 10.1016/B978-0-12-374498-2.00008-4.

[49] P. L. Falb and W. A. Wolovich. "Decoupling in the Design and Synthesis of Multivariable Control Systems". In: *IEEE Transactions on Automatic Control* (Dec. 1967), pp. 651–659.

[50] S. Galeani, A. Serrani, G. Varano, and L. Zaccarian. "On input allocation-based regulation for linear over-actuated systems". In: *Automatica* 52 (2015), pp. 346–354. DOI: 10.1016/j.automatica.2014.10.112.

[51] J. Garloff and B. Graf. "Solving strict polynomial inequalities by Bernstein expansion". In: *The Use of Symbolic Methods in Control System Analysis and Design* (1999), pp. 339–352.

[52] E. Garone, L. Ntogramatzidis, and F. Padula. "A Structural Approach To State-to-Output-Decoupling". In: *SIAM* 56.5 (2018), pp. 3816–3847.

[53] E. G. Gilbert. "Controllability and Observability in Multivariable Control Systems". In: *Journal of the Society for Industrial and Applied Mathematics Series A Control* 1 (1963), pp. 128–151. DOI: 10.1137/0301009.

[54] G. T. Gilbert. "Positive Definite Matrices and Sylvester's Criterion". In: *The American Mathematical Monthly* 98 (1991), p. 44. DOI: 10.2307/2324036.

[55] M. A. Goberna, J. A. Mira, and G. Torregrosa. "Redundancy in linear inequality system". In: *Numerical Functional Analysis and Optimization* 19 (1998), pp. 529–548. DOI: 10.1080/01630569808816843.

[56] E. Goubault and S. Putot. "Forward Inner-Approximated Reachability of Non-Linear Continuous Systems". In: *20th International Conference on Hybrid Systems: Computation and Control*. 2017, pp. 1–10. DOI: 10.1145/3049797.3049811.

[57] L. Groß. "Polplatzierung für Zustandsregler mit linearen Gleichungsbeschränkungen". MA thesis. Technische Universität Darmstadt, Oct. 2021.

[58] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. 2nd ed. Springer, 2011, pp. 43–66. ISBN: 978-3-319-46024-6. DOI: 10.1007/978-0-85729-501-9_3.

[59] O. Härkegård. "Backstepping and Control Allocation with Applications to Flight Control". PhD thesis. Linköping University, 2003.

[60] O. Härkegård and S. T. Glad. "Resolving actuator redundancy – optimal control vs. control allocation". In: *Automatica* 41.1 (2005), pp. 137–144.

[61] M. Hautus. "Controllability and Observability Conditions of Linear Autonomous Systems". In: *Nederlandse Akademie van Wetenschappen. Proceedings. Series A. Indagationes Mathematicae* 72 (Jan. 1969).

[62] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari. "Multi-Parametric Toolbox 3.0". In: *Proceedings of the European Control Conference*. Zürich, Switzerland, July 2013, pp. 502–510. URL: http://control.ee.ethz.ch/~mpt.

[63] C. Himpe. "emgr – The Empirical Gramian Framework". In: *Algorithms* 11.7 (2018). DOI: 10.3390/a11070091.

[64] G. Hirzinger. "Decoupling multivariable systems by optimal control techniques". In: *International Journal of Control* 22.2 (1975), pp. 157–168.

[65] A. Hurwitz. "Ueber die Bedingungen, unter welchen eine Gleichung nur Wurzeln mit negativen reellen Theilen besitzt". In: *Mathematische Annalen* 46.2 (1895), pp. 273–284. DOI: 10.1007/BF01446812.

[66] Innovaphone AG. *White Paper Redundancy: How Fail-Safe is Your Communication Infrastructure?* Tech. rep. Jan. 2018.

[67] R. Isermann. *Fault-Diagnosis Systems – An Introduction from Fault Detection to Fault Tolerance*. Springer Berlin, Heidelberg, 2006. DOI: 10.1007/3-540-30368-5.

[68] A. Jameson and E. Kreindler. "Inverse Problem of Linear Optimal Control". In: *Siam Journal on Control* (1973), pp. 1–19.

[69] T. A. Johansen. "Optimizing nonlinear control allocation". In: *43rd IEEE Conference on Decision and Control*. 2004. DOI: 10.1109/cdc.2004.1429240.

[70] T. A. Johansen and T. I. Fossen. "Control allocation – A survey". In: 49 (2013), pp. 1087–1103. DOI: 10.1016/j.automatica.2013.01.035.

[71] C. Jones, E. C. Kerrigan, and J. Maciejowski. *Equality set projection: A new algorithm for the projection of polytopes in halfspace representation*. Tech. rep. Cambridge University Engineering Dept, 2004.

[72] C. N. Jones, E. C. Kerrigan, and J. M. Maciejowski. "On polyhedral projection and parametric programming". In: *Journal of Optimization Theory and Applications* 138 (2008), pp. 207–220.

[73] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Transactions of the ASME - Journal of Basic Engineering* 82 (1960), pp. 35–45. DOI: 10.1115/1.3662552.

[74] R. E. Kalman. “Canonical Structure of Linear Dynamical Systems”. In: *Proceedings of the National Academy of Sciences* 48 (1962), pp. 596–600. DOI: 10.1073/pnas.48.4.596.

[75] R. E. Kalman. “Mathematical Description of Linear Dynamical Systems”. In: *Journal of the Society for Industrial and Applied Mathematics Series A Control* 1 (1963), pp. 152–192. DOI: 10.1137/0301010.

[76] R. E. Kalman. “When Is a Linear Control System Optimal?” In: 86 (1964), pp. 51–60. DOI: 10.1115/1.3653115.

[77] R. E. Kalman and J. E. Bertram. “General synthesis procedure for computer control of single-loop and multiloop linear systems (an optimal sampling system)”. In: *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry* 77 (1959), pp. 602–609. DOI: 10.1109/tai.1959.6371508.

[78] R. E. Kalman and R. W. Koepcke. “Optimal Synthesis of Linear Sampling Control Systems Using Generalized Performance Indexes”. In: *Transactions of the American Society of Mechanical Engineers* 80 (1958), pp. 1820–1826. DOI: 10.1115/1.4012899.

[79] S. S. Keerthi and K. Sridharan. “Solution of parametrized linear inequalities by Fourier elimination and its applications”. In: *Journal of Optimization Theory and Applications* 65 (1990), pp. 161–169. DOI: 10.1007/bf00941167.

[80] A. Khelassi, D. Theilliol, and P. Weber. “Reconfigurable control design for over-actuated systems based on reliability indicators”. In: *2010 Conference on Control and Fault Tolerant Systems*. 2010. DOI: 10.1049/ic.2010.0339.

[81] S. Ko. *Performance Limitations in Linear Estimation and Control: Constraint and Quantization Effects*. University of California, San Diego, 2005.

[82] S. Ko and R. R. Bitmead. “Optimal control for linear systems with state equality constraints”. In: *Automatica* 43.9 (2007), pp. 1573–1582.

[83] N. Kochdumper and M. Althoff. “Computing Non-Convex Inner-Approximations of Reachable Sets for Nonlinear Continuous Systems”. In: *59th IEEE Conference on Decision and Control*. 2020. DOI: 10.1109/cdc42340.2020.9304022.

[84] U. Konigorski. “Ausgangsgrößenverkopplung bei linearen Mehrgrößensystemen”. In: *at – Automatisierungstechnik* (Apr. 1999), pp. 165–170.

[85] I. Koren and C. M. Krishna. *Information Redundancy*. Ed. by I. Koren and C. M. Krishna. Second Edition. San Francisco (CA): Morgan Kaufmann, 2021, pp. 59–114. ISBN: 978-0-12-818105-8. DOI: 10.1016/B978-0-12-818105-8.00013-9.

[86] B. Kouvaritakis and M. Cannon. *Model Predictive Control: Classical, Robust and Stochastic*. Springer Cham, Dec. 2015. ISBN: 978-3-319-24853-0.

[87] E. Kreindler and A. Jameson. “Optimality of linear control systems”. In: *IEEE Transactions on Automatic Control* 17 (1972), pp. 349–351. DOI: 10.1109/tac.1972.1099985.

[88] E. Kreindler and P. Sarachik. “On the concepts of controllability and observability of linear systems”. In: *IEEE Transactions on Automatic Control* 9 (1964), pp. 129–136. DOI: 10.1109/tac.1964.1105665.

[89] J. Kreiss and J.-F. Trégouët. “Input redundancy: Definitions, taxonomy, characterizations and application to over-actuated systems”. In: *Systems & Control Letters* 158 (2021), p. 105060. DOI: 10.1016/j.sysconle.2021.105060.

[90] V. Kučera. “Decoupling optimal controllers”. In: *Proc. 18th Internat. Conf. Process Control*. 2011, pp. 400–407.

[91] H. Lebesgue. “Intégrale, Longueur, Aire”. In: *Annali de Matematica* 7.3 (1902), pp. 231–359. DOI: 10.1007/bf02420592.

[92] L. Lessard and S. Lall. “Optimal controller synthesis for the decentralized two-player problem with output feedback”. In: *2012 American Control Conference*. 2012, pp. 6314–6321.

[93] F. Lin, M. Fardad, and M. R. Jovanović. “Design of Optimal Sparse Feedback Gains via the Alternating Direction Method of Multipliers”. In: *IEEE Transactions on Automatic Control* 58.9 (2013), pp. 2426–2431.

[94] L. Litz. “Modale Maße für Steuerbarkeit, Beobachtbarkeit, Regelbarkeit und Dominanz – Zusammenhänge, Schwachstellen, neue Wege / Modal measures of Controllability, Observability, Dominance – Connections, Drawbacks and a new Approach”. In: *at – Automatisierungstechnik* 31.1-12 (1983), pp. 148–158.

[95] J. Lückel and P. C. Müller. “Analyse von Steuerbarkeits-, Beobachtbarkeits- und Störbarkeitsstrukturen linearer zeitinvarianter Systeme / Analysis of controliability-, observability- and disturbability-structures in linear time invariant systems”. In: *at – Automatisierungstechnik* 23.1-12 (1975), pp. 163–171.

[96] J. Lunze. *Regelungstechnik 1*. Springer Vieweg, 2020. DOI: 10.1007/978-3-662-60746-6.

[97] J. Lunze. *Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung*. 9th ed. Berlin: Springer Vieweg, 2016. ISBN: 978-3-662-52675-0. DOI: 10.1007/978-3-662-52676-7.

[98] Z. Ma and S. Zou. *Optimal Control Theory: The Variational Method*. Springer Nature, 2021.

[99] J.-F. Magni. *Robust Modal Control with a Toolbox for Use with Matlab*. Springer Science + Business, 2002. ISBN: 978-1-4613-5170-2.

[100] J. R. Magnus and H. Neudecker. “The Commutation Matrix: Some Properties and Applications”. In: *The Annals of Statistics* 7.2 (1979), pp. 381–394.

[101] G. Marro and G. Basile. *Geometric Approach Matlab Toolbox*. Internet source. Archived version due to deletion of original source. Sept. 2010. URL: http://web.archive.org/web/20220125182917/http://www3.deis.unibo.it/Staff/FullProf/GiovanniMarro/geometric.htm.

[102] G. Marro, F. Morbidi, L. Ntogramatzidis, and D. Prattichizzo. “Geometric control theory for linear systems: a tutorial”. In: *Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems – MTNS*. Vol. 5. 9. 2010.

[103] H. Mayeda. “On structural controllability theorem”. In: *IEEE Transactions on Automatic Control* 26 (1981), pp. 795–798. DOI: 10.1109/tac.1981.1102707.

[104] M. Menner, P. Worsnop, and M. N. Zeilinger. “Constrained Inverse Optimal Control With Application to a Human Manipulation Task”. In: *IEEE Transactions on Control Systems Technology* 29.2 (2021), pp. 826–834. DOI: 10.1109/TCST.2019.2955663.

[105] P.-J. Meyer, A. Devonport, and M. Arcak. *Interval Reachability Analysis*. Springer Cham, Jan. 2021. ISBN: 978-3-030-65110-7.

[106] Y. Michellod. “Overactuated systems coordination”. PhD thesis. École Polythechnique Fédérale Lausanne, 2009. DOI: 10.5075/epfl-thesis-4299.

[107] D. E. Miller. “A Comparison of LQR Optimal Performance in the Decentralized and Centralized Settings”. In: *IEEE Transactions on Automatic Control* 61.8 (2016), pp. 2308–2311.

[108] D. E. Miller and E. J. Davison. “Near optimal LQR control in the decentralized setting”. In: *2011 50th IEEE Conference on Decision and Control and European Control Conference*. 2011, pp. 6204–6209.

[109] B. Moore. “On the flexibility offered by state feedback in multivariable systems beyond closed loop eigenvalue assignment”. In: *IEEE Transactions on Automatic Control* 21 (1976), pp. 689–692. DOI: 10.1109/tac.1976.1101355.

[110] A. S. Morse and W. H. Wonham. “Triangular Decoupling of Linear Multivariable Systems”. In: *IEEE Transactions on Automatic Control* (Aug. 1970), pp. 447–449.

[111] J. L. Musgrave, T.-H. Guo, E. Wong, and A. Duyar. “Real-time accommodation of actuator faults on a reusable rocket engine”. In: *IEEE Transactions on Control Systems Technology* 5.1 (1997), pp. 100–109. DOI: 10.1109/87.553668.

[112] National Transportation Safety Board. *Aircraft Accident Report – United Airlines Flight 585*. Tech. rep. 1991.

[113] National Transportation Safety Board. *Aircraft Accident Report – USAIR Flight 427*. Tech. rep. 1994.

[114] G. Nelson. *A User-Friendly Introduction to Lebesgue Measure and Integration, Providence, Rhode Island*. Ed. by S. L. Devadoss, E. Flapan, J. Stillwell, and S. Tabachnikov. American Mathematical Association, 2015. DOI: 10.1090/stml/078.

[115] M. Nowak, M. Boerlijst, and J. Cooke. “Evolution of genetic redundancy”. In: *Nature* 388 (1997), pp. 167–171.

[116] L. Ntogramatzidis. “Self-Bounded Subspaces for Nonstrictly Proper Systems and Their Application to the Disturbance Decoupling With Direct Feedthrough Matrices”. In: *IEEE Transactions on Automatic Control* 53 (2008), pp. 423–428. DOI: 10.1109/tac.2007.914273.

[117] L. Ntogramatzidis and F. Padula. “A general approach to the eigenstructure assignment for reachability and stabilizability subspaces”. In: *Systems & Control Letters* 106 (2017), pp. 58–67. DOI: 10.1016/j.sysconle.2017.06.003.

[118] L. Ntogramatzidis and R. Schmid. “Robust Eigenstructure Assignment in Geometric Control Theory”. In: *SIAM* 52.2 (2014), pp. 960–986.

[119] L. Ntogramatzidis and R. Schmid. “Robust eigenstructure assignment in the computation of friends of output-nulling subspaces”. In: *52nd IEEE Conference on Decision and Control*. 2013. DOI: 10.1109/cdc.2013.6760086.

[120] M. W. Oppenheimer, D. B. Doman, and M. A. Bolender. “Control Allocation for Over-actuated Systems”. In: *2006 14th Mediterranean Conference on Control and Automation, Ancona, Italy*. 2006. DOI: 10.1109/med.2006.328750.

[121] N. Osmic, A. Tahirovic, and I. Petrovic. “Risk-sensitive motion planning for MAVs based on mission-related fault-tolerant analysis”. In: *Automatika* 61.2 (2020), pp. 295–311. DOI: 10.1080/00051144.2020.1733324.

[122] N. Otsuka. “A necessary and sufficient condition for parameter insensitive disturbance-rejection problem with state feedback”. In: *Automatica* 42 (2006), pp. 447–452. DOI: 10.1016/j.automatica.2005.10.012.

[123] F. Padula, L. Ntogramatzidis, and E. Garone. “MIMO tracking control of LTI systems: A geometric approach”. In: *Systems & Control Letters* 126 (2019), pp. 8–20. DOI: 10.1016/j.sysconle.2019.02.003.

[124] C. Paredis and P. Khosla. “Global trajectory planning for fault tolerant manipulators”. In: Sept. 1995, 428–434 vol.2. ISBN: 0-8186-7108-4. DOI: 10.1109/IROS.1995.526252.

[125] S. Paulraj, C. Chellappan, and T. R. Natesan. “A heuristic approach for identification of redundant constraints in linear programming models”. In: *International Journal of Computer Mathematics* 83 (2006), pp. 675–683. DOI: 10.1080/00207160601014148.

[126] J. A. Paulson, T. A. N. Heirung, and A. Mesbah. “Fault-Tolerant Tube-Based Robust Nonlinear Model Predictive Control”. In: *2019 American Control Conference*. 2019. DOI: 10.23919/acc.2019.8815069.

[127] D. Peaucelle and D. Arzelier. “Robust performance analysis with LMI-based methods for real parametric uncertainty via parameter-dependent Lyapunov functions”. In: *IEEE Transactions on Automatic Control* 46.4 (2001), pp. 624–630. DOI: 10.1109/9.917664.

[128] O. Petrosian, J. Inga, I. Kuchkarov, M. Flad, and S. Hohmann. “Optimal Control and Inverse Optimal Control with Continuous Updating for Human Behavior Modeling”. In: *IFAC–PapersOnLine* 53.2 (2020). 21st IFAC World Congress, pp. 6670–6677. DOI: 10.1016/j.ifacol.2020.12.089.

[129] B. Polyak. “Convexity of the Reachable Set of Nonlinear Systems Under L 2 Bounded Controls”. In: *Dynamics of Continuous Discrete and Impulsive Systems Series A: Mathematical Analysis* 11 (Apr. 2004), pp. 255–267.

[130] M. C. Priess, R. Conway, J. Choi, J. M. Popovich, and C. Radcliffe. “Solutions to the Inverse LQR Problem With Application to Biological Systems Analysis”. In: *IEEE Transactions on Control Systems Technology* 23 (2015), pp. 770–777. DOI: 10.1109/tcst.2014.2343935.

[131] K. F. Prochazka, T. Ritz, and H. Eduardo. “Over-actuation analysis and fault-tolerant control of a hybrid unmanned aerial vehicle”. In: *5th CEAS Conference on Guidance, Navigation and Control*. 2019.

[132] M. Pusch and D. Ossmann. “$\mathcal{H}_2$-Optimal Blending of Inputs and Outputs for Modal Control”. In: *IEEE Transactions on Control Systems Technology* 28.6 (2020), pp. 2744–2751.

[133] S. V. Raković, E. C. Kerrigan, D. Q. Mayne, and K. I. Kouramas. “Optimized robust control invariance for linear discrete-time systems: Theoretical foundations”. In: *Automatica* 43 (2007), pp. 831–841. DOI: 10.1016/j.automatica.2006.11.006.

[134] S. V. Raković. “Set Theoretic Methods in Model Predictive Control”. In: *Nonlinear Model Predictive Control: Towards New Challenging Applications*. Ed. by L. Magni, D. M. Raimondo, and F. Allgöwer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 41–54. ISBN: 978-3-642-01094-1. DOI: 10.1007/978-3-642-01094-1_3.

[135] S. V. Raković and M. Barić. “Parameterized Robust Control Invariant Sets for Linear Systems: Theoretical Advances and Computational Remarks”. In: *IEEE Transactions on Automatic Control* 55 (2010), pp. 1599–1614. DOI: 10.1109/tac.2010.2042341.

[136] S. V. Raković and W. S. Levine, eds. *Handbook of Model Predictive Control*. 1st ed. Birkhäuser Cham, Sept. 2018. ISBN: 978-3-319-77489-3.

[137] N. Ramdani, N. Meslem, and Y. Candau. “Computing reachable sets for uncertain nonlinear monotone systems”. In: *Nonlinear Analysis: Hybrid Systems* 4.2 (2010), pp. 263–278.

[138] A. Rantzer and B. Bernhardsson. “Control of convex-monotone systems”. In: *53rd IEEE Conference on Decision and Control*. 2014. DOI: 10.1109/cdc.2014.7039751.

[139] J. Rawlings, D. Q. Mayne, and M. Diehl. *Model Predictive Control: Theory, Computation, and Design*. Jan. 2017.

[140] G. Reißig. “Convexity of reachable sets of nonlinear ordinary differential equations”. In: *Automation and Remote Control* 68 (2007), pp. 1527–1543. DOI: 10.1134/s000511790709007x.

[141] M. Rieutord. *Fluid Dynamics – An Introduction*. Springer, Jan. 9, 2015. ISBN: 978-3-319-09350-5.

[142] H. H. Rosenbrock. *State-Space and Multivariable Theory*. Wiley, New York, 1970.

[143] M. Rotkowitz and S. Lall. “A Characterization of Convex Problems in Decentralized Control”. In: *IEEE Transactions on Automatic Control* 51.2 (2006), pp. 274–286.

[144] H. L. Royden. *Real Analysis*. 3rd. Macmillan Publishing Company, 1988.

[145] A. Savchenko. *Efficient Set-based Process Monitoring and Fault Diagnosis*. Vol. 7. Contributions in Systems Theory and Automatic Control, Otto-von-Guericke-Universität Magdeburg. Aachen: Shaker Verlag, 2017. ISBN: 9783844052121.

[146] P. Schaub and U. Konigorksi. "Structured Linear Quadratic Regulator Design". In: *2023 27th International Conference on System Theory, Control and Computing (ICSTCC)*. 2023, pp. 184–191. DOI: 10.1109/ICSTCC59206.2023.10308439.

[147] P. Schaub and U. Konigorski. "Actuator Redundancy and Safe Operation Abilities of Nonlinear Systems". In: *IFAC-PapersOnLine* 56.2 (2023). 22nd IFAC World Congress, pp. 3598–3603. DOI: 10.1016/j.ifacol.2023.10.1520.

[148] P. Schaub, P. Vogt, and U. Konigorski. "Robust coupling control using pole region assignment". In: *Systems & Control Letters* 158 (Dec. 2021), p. 105067. DOI: 10.1016/j.sysconle.2021.105067.

[149] R. Schmid, A. Pandey, and T. Nguyen. "Robust Pole Placement With Moore's Algorithm". In: *IEEE Transactions on Automatic Control* 59.2 (2014), pp. 500–505.

[150] F. P. Schuller. *Lectures on Quantum Theory*. Tech. rep. Friedrich-Alexander Universität Erlangen-Nürnberg, Apr. 2019.

[151] A. Serrani. "Output regulation for over-actuated linear systems via inverse model allocation". In: *51st IEEE Conference on Decision and Control*. 2012. DOI: 10.1109/cdc.2012.6426209.

[152] P. J. Shaffer and I. M. Ross. "Fault-Tolerant Optimal Trajectory Generation for Reusable Launch Vehicles". In: *Journal of Guidance, Control, and Dynamics* 30.6 (Nov. 2007), pp. 1794–1802.

[153] T. S. Shores. *Applied Linear Algebra and Matrix Analysis*. Ed. by S. Axler and K. Ribet. Springer, 2018. DOI: 10.1007/978-3-319-74748-4.

[154] H. Smith. *Monotone dynamical systems: An Introduction to the Theory of Competitive and Cooperative Systems*. American Mathematical Society, 2008, pp. 1–13.

[155] J. Swigart and S. Lall. "Optimal Controller Synthesis for Decentralized Systems Over Graphs via Spectral Factorization". In: *IEEE Transactions on Automatic Control* 59.9 (2014), pp. 2311–2323.

[156] J. Telgen. "Minimal representation of convex polyhedral sets". In: 38 (1982), pp. 1–24. DOI: 10.1007/bf00934319.

[157] J. Telgen. *On Redundancy In Systems Of Linear Inequalities*. Tech. rep. 272154. Erasmus University Rotterdam, Sept. 1977.

[158] J. Telgen. *Redundancy and linear programs*. Mathematisch Centrum, 1982.

[159] S. S. Tohidi, Y. Yildiz, and I. Kolmanovsky. "Adaptive Control Allocation for Over-Actuated Systems with Actuator Saturation". In: 50 (2017), pp. 5492–5497. DOI: 10.1016/j.ifacol.2017.08.1088.

[160] H. L. Trentelman. *Control theory for linear systems*. London New York: Springer, 2001. ISBN: 978-1-4471-0339-4.

[161] J. Trumpf. “On the Geometry and Parametrization of Almost Invariant Subspaces and Observer Theory”. PhD thesis. Universität Würzburg, 2002.

[162] M. E. Valcher. “Reachability Properties of Continuous-Time Positive Systems”. In: *IEEE Transactions on Automatic Control* 54 (2009), pp. 1586–1590. DOI: 10.1109/tac.2009.2015556.

[163] A. Varga. “Fault detection and isolation of actuator failures for a large transport aircraft”. In: *Proceeding First CEAS European Air and Space Conference, Berlin, Germany*. 2007.

[164] E. Vlahakis, G. Provan, G. Werner, S. Yang, and N. Athanasopoulos. “Quantifying Impact on Safety from Cyber-Attacks on Cyber-Physical Systems”. In: *22nd IFAC World Congress*. 2023.

[165] P. Vogt, E. Lenz, and U. Konigorski. “gammasyn – eine Toolbox zur robusten Polbereichsvorgabe mittels beschränkter Optimierung und Straffunktionen”. In: *at – Automatisierungstechnik* 68.10 (May 2020), pp. 893–904.

[166] A. Wahrburg and J. Adamy. “Entkopplungsregelungen für lineare überaktuierte Systeme”. In: *at – Automatisierungstechnik* 61 (2013), pp. 28–38. DOI: 10.1524/auto.2013.0006.

[167] S. Weiland and J. C. Willems. “Almost disturbance decoupling with internal stability”. In: *IEEE Transactions on Automatic Control* 34 (1989), pp. 277–286. DOI: 10.1109/9.16417.

[168] J. Willems. “Almost invariant subspaces: An approach to high gain feedback design – Part I: Almost controlled invariant subspaces”. In: *IEEE Transactions on Automatic Control* 26 (1981), pp. 235–252. DOI: 10.1109/tac.1981.1102551.

[169] J. Willems. “Almost invariant subspaces: An approach to high gain feedback design – Part II: Almost conditionally invariant subspaces”. In: *IEEE Transactions on Automatic Control* 27 (1982), pp. 1071–1085. DOI: 10.1109/tac.1982.1103074.

[170] J. C. Willems. “Almost A(mod β)-invariant subspaces”. In: *Analyse des systèmes, Astérisque* 75-76 (1980), pp. 239–248.

[171] J. C. Willems. “Approximate disturbance decoupling by measurement feedback”. In: ed. by D. Hinrichsen and A. Isidori. Vol. 39. Springer, 1982. Chap. Feedback Control of Linear and Nonlinear Systems. Lecture Notes in Control and Information Sciences, pp. 268–284. DOI: 10.1007/bfb0006835.

[172] W. M. Wonham and A. S. Morse. “Decoupling and Pole Assignment in Linear Multivariable Systems: A Geometric Approach”. In: *SIAM Journal on Control* 8 (1970), pp. 1–18. DOI: 10.1137/0308001.

[173] W. M. Wonham. *Linear Multivariable Control: a Geometric Approach*. 2nd ed. Springer, 1979. ISBN: 3-540-90354-2.

[174] J. Wu and Y. Lu. “Decoupling and Optimal Control of Multilevel Buck DC-DC Converters With Inverse System Theory”. In: *IEEE Transactions on Industrial Electronics* 67.9 (2020), pp. 7861–7870.

[175] G.-H. Yang, H. Wang, and L. Xie. “Fault detection for output feedback control systems with actuator stuck faults: A steady-state-based approach”. In: *International Journal of Robust and Nonlinear Control* 20.15 (2010), pp. 1739–1757. DOI: 10.1002/rnc.1546.

[176] L. Yang and N. Ozay. “Tight decomposition functions for mixed monotonicity”. In: *58th IEEE Conference on Decision and Control*. 2019. DOI: 10.1109/cdc40024.2019.9030065.

[177] T. Yoshikawa. “Analysis and control of robot manipulators with redundancy”. In: *Preprints of 1st Int. Symp. of Robotics Research, Boston, MA, 1983*. 1983, pp. 735–747.

[178] T. Yoshikawa. “Manipulability of robotic mechanisms”. In: *The international journal of Robotics Research* 4.2 (1985), pp. 3–9.

[179] L. Zaccarian. “Dynamic allocation for input redundant control systems”. In: *Automatica* 45 (2009), pp. 1431–1438. DOI: 10.1016/j.automatica.2009.01.013.

[180] P. Zhou, F.-Y. Wang, W. Chen, and P. Lever. “Optimal construction and control of flexible manipulators: a case study based on LQR output feedback”. In: *Mechatronics* 11.1 (2001), pp. 59–77.

Own Publications

[O1] P. Schaub and U. Konigorksi. “Structured Linear Quadratic Regulator Design”. In: *2023 27th International Conference on System Theory, Control and Computing (ICSTCC)*. 2023, pp. 184–191. DOI: 10.1109/ICSTCC59206.2023.10308439.

[O2] P. Schaub and U. Konigorski. “Actuator Redundancy and Safe Operation Abilities of Nonlinear Systems”. In: *IFAC-PapersOnLine* 56.2 (2023). 22nd IFAC World Congress, pp. 3598–3603. DOI: 10.1016/j.ifacol.2023.10.1520.

[O3] P. Schaub, P. Vogt, and U. Konigorski. “Robust coupling control using pole region assignment”. In: *Systems & Control Letters* 158 (Dec. 2021), p. 105067. DOI: 10.1016/j.sysconle.2021.105067.