Lecture Notes

Advanced Topics in Control

Herbert Werner

TUHH
Hamburg University of Technology

ICS
Institute of Control Systems

ver. 20th November, 2020

Contents

Introduction

The aim of this lecture course *Advanced Topics in Control* is to bridge the gap between a basic knowledge of H_2 and H_∞ optimal control and of the LMI framework used to design and analyse such controllers (which is taught in the course *Optimal and Robust Control*), and some issues arising in currently active research areas in the field of advanced control. This course is intended to provide a thorough understanding of advanced concepts and methods as well as familiarity with associated state-of-the-art software tools, that will enable students to apply techniques such as gain-scheduling control of nonlinear and time-varying systems, cooperative control of multi-agent systems or distributed control of interconnected systems.

This course will build on the material taught in *Optimal and Robust Control*; a firm grasp in particular of the use of the H_∞ norm for synthesis and analysis, and of formulating these problems as LMI Problems will be essential here. We will see how the notion of the H_∞ norm - interpreted as a system norm induced by the signal-2 norm - can be extended to time-varying as well as to multi-agent or distributed systems, leading to analysis and synthesis tools similar to H_∞-norm-based tools for LTI systems.

Three research areas will be covered. The first one, control of *Linear Parameter-Varying Systems* (or LPV systems for short), extends the machinery of H_∞ optimal control from LTI systems to time-varying systems. This is particularly useful when a nonlinear plant is represented as a parameter-dependent linear system, where the time-varying parameters are allowed to be functions of input, output or state variables of the nonlinear plant model. In this case the approach can be used to design gain-scheduled controllers for nonlinear plants in the same way H_∞ controllers are designed for LTI systems (e.g. in the mixed-sensitivity framework); the associated analysis and synthesis problems can be solved as LMI problems. Moreover, as with LTI systems these gain-scheduled controllers come with stability and performance guarantees for the nonlinear closed-loop systems. This part of the course starts with a brief look at the history of gain scheduling, and a review of heuristic gain-scheduling control, and concludes with case studies on the design of gain-scheduled controllers for a robot manipulator, for a control moment gyroscope and of a torque vectoring scheme for an electric car.

The second research area covered in this course is *Cooperative Control of Multi-Agent Systems*. Here one can think of groups of mobile agents such as unmanned aerial or

underwater vehicles, or wheeled mobile robots, equipped with sensors and actuators, which are operating in a coordinated fashion in order to perform a common task. Possible applications include monitoring of disaster areas and search and rescue operations. Rather than employing a centralised control scheme that requires communication links between a central control unit and each single agent, a much more efficient and robust approach (inspired by swarm behaviour observed in nature) is to operate with a group of autonomous agents, where each agent communicates only with its nearest neighbours and acts based on local information. The communication structure in such a group of agents can be represented by a graph, and the design of cooperative control schemes involves a combination of control theory and graph theory. Beginning with the study of consensus protocols, we will focus on a typical problem arising in cooperative control, the *formation control problem*. Agents will initially be modelled as LTI systems, but results will later be extended to agents with LPV dynamics, thus allowing to solve formation control problems for agents subject to non-holonomic constraints.

The third part of this course presents an introduction to *Distributed Control of Spatially Interconnected Systems*. An important feature of these systems is that they do not only depend on time as independent variable, but also on space. Applications we will consider in this context are control of a heat profile over space using an array of temperature sensors and actuators, and vibration control of a flexible structure with the help of an array of piezo sensors and actuators. The underlying systems are governed by partial differential equations, and the use of an array of collocated actuator-sensor pairs induces a spatial discretisation that turns the partial differential equations into partial difference equations. The spatially discretised system can thus be seen as an interconnection of individual segments (each corresponding to a single actuator-sensor pair and interacting with neighbouring segments). The aim is to design a distributed control scheme where a local controller is attached to each actuator-sensor pair and controllers communicate with each other, using the same communication structure as the plant.

The material in each part of the course is accompanied by a number of exercise problems. In some of these problems students are asked to fill in gaps in the derivation of theoretical results; the majority of problems however present illustrative examples, including simulation studies for design problems that involve test rigs available at the Institute of Control Systems, where an experimental validation is possible (e.g. LPV gain scheduling for a control moment gyroscope, which can be carried out as a project in the new Control Lab offered by the Institute).

Much of the material presented in this course is based on results that have been developed only recently and are not yet available in standard text books. Reference is therefore made frequently to the research papers where these results were presented originally. All papers in the literature list are available as pdf documents via StudIP, and students are strongly encouraged to browse through the papers for more details.

Part I

LINEAR PARAMETER-VARYING SYSTEMS

Chapter 1

Nonlinear Systems and Classical Gain Scheduling

1.1 Heuristic Gain Scheduling – A Historic Perspective

We will begin the study of gain scheduling control with a brief look at the history of this particular field of control engineering; the following summary is based on a survey in [1], which contains many references that can be consulted for more details. Heuristic gain scheduling has been in practical use since the 1950s, even though until the 1990s there is little literature on this topic. Gain-scheduled controllers have been widely employed in applications where linear, fixed-gain control was not sufficient; they were designed heuristically and applied without rigorous stability and performance analysis, justified by the fact that they did indeed perform satisfactorily. Gain scheduling is still widely used; in [1] it is stated that

> ” . . . in any case, the market has now spoken: gain scheduling is an effective and economical method for nonlinear control design in practice . . . With little exaggeration we can say, for at least the last few decades, 'Machines that walk, swim, or fly, are gain scheduled'.”

The first applications of gain scheduling were actually ”machines that fly”; in fact for more than two decades the use of gain scheduling control was limited to the military domain. New developments after the second world war - the arrival of jet aircraft and guided missiles - posed challenges to control systems in maintaining stability and performance that could be addressed by gain-scheduled controllers. The reason that at the time there was no commercial interest in this type of control was that it was rather difficult to implement. In flight control for example the dynamic properties of an aircraft depend strongly on

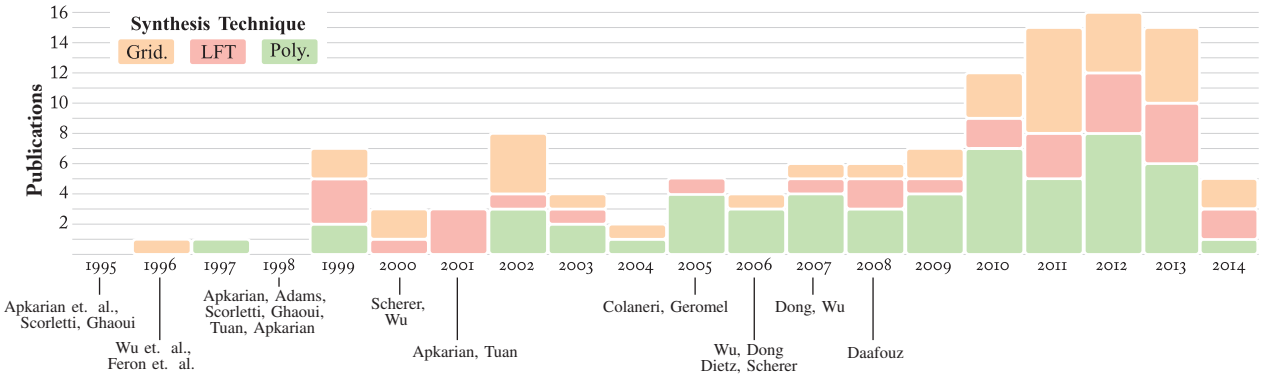


Figure 1.1: Development of number of experimental results in LPV control, together with publication years of key papers in the field

altitude and speed, and in order to schedule controller gains on these quantities one has to calculate e.g. the true air speed based on available measurements. That would be a simple task when digital control is employed, but digital control was not available at the time and everything had to be done using analog circuits. As a result, implementing gain-scheduled control required complex and expensive hardware, which made the approach unattractive for civilian use. The first reported application of gain-scheduled control was the autopilot of the B-52 strategic bomber of the US air force, which was scheduled on air speed and was supposed to maintain a desired closed-loop bandwidth over the whole flight envelope.

Commercial interest in gain scheduling emerged when microprocessors and digital control became available in the 1970s, and one of the first civilian applications of gain scheduling was automotive engine control, in particular control of the air/fuel ratio. The first gain-scheduled engine controller was implemented in serial production on the Ford Pinto, as well as on its rebadged version Mercury Bobcat, in 1978. What was used then was integral feedback scheduled initially on engine speed and later on speed and manifold pressure.

From Heuristic to LPV Gain Scheduling

Gain scheduling as it was used in the above mentioned applications basically meant to design local LTI controllers for local plant models that were linearised about suitable operating points, and to either switch or interpolate between these local controllers, depending on the current value of the scheduling variables. In the early 1990s, following the development of LMI-based solutions to H_∞ analysis and synthesis problems, it was recognised that the H_∞ approach can be extended from LTI systems to nonlinear and time-varying systems if it is possible to represent these as linear parameter-varying models, and that the resulting analysis and synthesis problems can still be solved efficiently as LMI problems.

It is this latter ”modern control approach” to gain scheduling that we will study in in the first part of this course, after taking a brief look back at heuristic gain scheduling as a

way of controlling nonlinear plants. Starting in the mid-90s, there has been an increasing number of reports about the use of LPV gain scheduling in practical applications. Figure 1.1, which is taken from [2], shows how the number of reported experimental validations of LPV gain scheduling control increased over the last 20 years, together with the year of publication of some of the important theoretical results. Indicated is also the synthesis technique used in each case: based on gridding, on LFT models or on polytopic models; these approaches will be studied in detail in Chapters 3 and 4.

1.2 Nonlinear Systems

Consider a nonlinear dynamic system represented by the model

$$\Sigma: \quad \begin{cases} \dot{x}(t) &= f\Big(t, x(t), u(t)\Big), \quad x(0) = x_o \\ y(t) &= h\Big(t, x(t), u(t)\Big) \end{cases} \tag{1.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ the input and $y(t) \in \mathbb{R}^l$ the output signal of the system. The functions $f : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $h : \mathbb{R} + \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^l$ (where $\mathbb{R}^+$ denotes the set of positive reals, here time) are nonlinear, and in the following we will, unless stated otherwise, assume that these functions are continuous in x and y, and that f is continuously differentiable and satisfies a Lipschitz condition, so that existence and uniqueness of a solution to the state equation is guaranteed (see [3]). Note that f and h are allowed to depend on time; the system in (1.1) is referred to as a *nonlinear time-varying system*. When f and h do not depend on time, the system is called *nonlinear time-invariant*. A familiar special case of the latter is the linear time-invariant (LTI) system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

where f and h are independent of time and depend linearly on x and u, respectively.

Definition 1.1 *A value of the state vector $\bar{x}$ is called an* equilibrium *of the unforced system Σ if it satisfies*

$$0 = f(t, \bar{x}, 0) \quad \forall t \geq 0.$$

Unforced LTI systems always have an equilibrium at $\bar{x} = 0$; if an LTI system has a nonsingular system matrix A (no integral behaviour), this is the single unique equilibrium.

Parameter-dependent systems

We will be interested in systems whose dynamic models depend on a set of possibly time-varying parameters $\rho_i(t)$, $i = 1, \ldots, n_\rho$. Collecting the parameters into a parameter

vector $\rho(t)$, we represent such a parameter-dependent system by the model

$$\Sigma_\rho : \quad \left\{ \begin{array}{rcl} \dot{x}(t) & = & f\Big(\rho(t), x(t), u(t)\Big), \quad x(0) = x_o \\ y(t) & = & h\Big(\rho(t), x(t), u(t)\Big) \end{array} \right. \tag{1.2}$$

Note that in this representation f and h do not explicitly depend on time. The system (1.2) is still a nonlinear time-varying system, but the dependence on time is expressed via the time-varying parameter $\rho(t)$.

We will usually associate with a time-varying parameter vector $\rho(t)$ a compact set $\mathcal{P}$, and assume that the parameter vector satisfies

$$\rho(t) \in \mathcal{P} \subset \mathbb{R}^{n_\rho} \quad \forall t \geq 0. \tag{1.3}$$

For the finite dimensional vector space $\mathbb{R}^{n_\rho}$ considered here compact means closed and bounded.

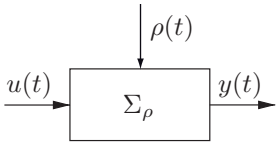


Figure 1.2: Nonlinear system

A block diagram of such a system is shown in Fig. 1.2. For a fixed value of ρ, Σ_ρ in (1.2) is a nonlinear time-invariant system. For such systems, we can consider a forced equilibrium with a constant input $\bar{u}$, determined by

$$0 = f\Big(\rho, \bar{x}(\rho), \bar{u}(\rho)\Big).$$

The equilibrium values $\bar{x}(\rho)$ and $\bar{u}(\rho)$ are functions of the parameter vector ρ, and we refer to $\Big(\bar{x}(\rho), \bar{u}(\rho)\Big)$ as an *equilibrium family* parameterized by ρ. Associated with this equilibrium family is the *equilibrium output family*

$$\bar{y}(\rho) = h\Big(\rho, \bar{x}(\rho), \bar{u}(\rho)\Big).$$

Generalized plant

The parameter-dependent System Σ_ρ can be augmented to include an additional external input channel w and a fictitious output channel z, as indicated in Fig. 1.3. These signals can for example be used to represent a performance channel or model uncertainty; how this is used in the case of LTI systems is discussed in the Lecture Notes *Optimal and Robust Control*. A state space realisation of the generalised plant in Fig. 1.3 is then

$$\begin{array}{rcl} \dot{x}(t) & = & f\Big(\rho(t), x(t), u(t), w(t)\Big), \quad x(0) = x_o \\ z(t) & = & h_z\Big(\rho(t), x(t), u(t), w(t)\Big) \\ v(t) & = & h_v\Big(\rho(t), x(t), u(t), w(t)\Big), \end{array} \tag{1.4}$$

where the feedback signal has been re-labeled $v(t)$.

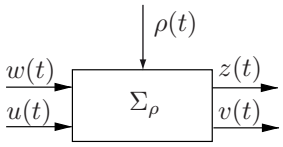


Figure 1.3: Generalized plant

A parameterized equilibrium family for this system is characterised by the condition

$$0 = f\Big(\rho, \bar{x}(\rho), \bar{u}(\rho), \bar{w}(\rho)\Big)$$

with a corresponding output equilibrium family $\Big(\bar{z}(\rho), \bar{v}(\rho)\Big)$.

1.3 Jacobian Linearisation and quasi-LPV Models

For the model (1.4) of a nonlinear generalised plant, we will now consider control strategies based on gain scheduling. In Section 1.1 we introduced the distinction between heuristic, classical gain scheduling based on Jacobian linearization on one hand, and gain scheduling based on a *quasi-LPV model* on the other hand. In this section we focus on the former approach and analyse properties and potential shortcomings of classical gain scheduling.

Given a nonlinear model Σ, the idea of classical gain scheduling is to obtain a linear representation of Σ that involves a set of scheduling parameters, and to use these parameters to schedule a control law. The way of selecting these parameters (which we will collect into the parameter vector ρ) may not be obvious initially but will become so in the process of linearisation, when they are used to parameterise an equilibrium family. This is illustrated by the following example, which is borrowed from [1].

Example 1.1 *Jacobian Linearisation*

Consider the system

$$\begin{aligned} \dot{x}(t) &= -x(t) + u(t) \\ y(t) &= \tanh x(t) \\ z(t) &= r(t) - y(t), \end{aligned} \tag{1.5}$$

where $y(t)$ is the controlled output, $r(t)$ an external reference input and the performance output $z(t)$ is taken to be the control error.

A block diagram of this generalised plant is shown in Fig. 1.4. The objective is to bring the control error z to zero, and we consider equilibrium states associated with this objective:

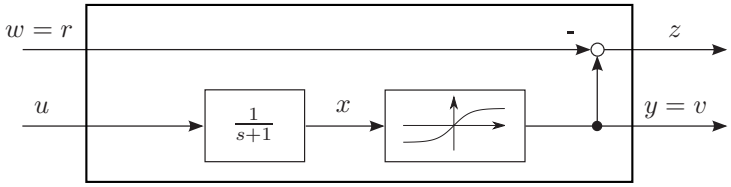


Figure 1.4: Block diagram for Example 1.1

let an equilibrium be determined by the condition

$$\bar{z} = 0 \tag{1.6}$$

which implies

$$\bar{y} = \bar{r}$$

for constant equilibrium values of y and r. Using (1.6), such equilibria can be characterised by a parameter ρ, defined as

$$\rho = \bar{r} = \bar{y} = \tanh \bar{x}. \tag{1.7}$$

With this definition of ρ and noting that the static gain from u to x is 1, we obtain the equilibrium family

$$\bar{x}(\rho) = \bar{u}(\rho) = \tanh^{-1} \rho.$$

It is also clear from the definition that we have

$$\rho \in [-1, 1] = \mathcal{P},$$

which defines the admissible parameter set $\mathcal{P}$.

To linearise this nonlinear model about $\rho = \bar{r} = \bar{y}$, we consider the deviation from the equilibrium

$$x = \bar{x} + \tilde{x}, \quad y = \bar{y} + \tilde{y}, \quad u = \bar{u} + \tilde{u}, \quad z = \tilde{z},$$

where $\tilde{x}$, $\tilde{y}$, $\tilde{u}$ and $\tilde{z}$ represent the deviation variables. The system considered here has the form

$$\begin{aligned} \dot{x} &= f(x, u, \rho) \\ y &= h(x, \rho). \end{aligned}$$

A linearisation about the equilibrium is obtained by calculating

$$\begin{aligned} \dot{\tilde{x}} &= \left.\frac{\partial f}{\partial x}\right|_{\bar{x}(\rho)} \tilde{x} + \left.\frac{\partial f}{\partial u}\right|_{\bar{u}(\rho)} \tilde{u} \\ \tilde{y} &= \left.\frac{\partial h}{\partial x}\right|_{\bar{x}(\rho)} \tilde{x} \end{aligned}$$

Note that the system is linearised about the equilibrium values of x and u, but not about ρ, which is used to parameterise the equilibrium family. With $f = -x + u$ and $y = \tanh x$ we obtain for the example considered here

$$\begin{aligned} \dot{\tilde{x}}(t) &= -\tilde{x}(t) + \tilde{u}(t) \\ \tilde{y}(t) &= c(\rho)\,\tilde{x}(t) \end{aligned} \tag{1.8}$$

where

$$c(\rho) = \left.\frac{\partial}{\partial x}\right|_{\bar{x}(\rho)} \tanh x = 1 - \tanh^2 \bar{x}(\rho)$$

or using (1.7)

$$c(\rho) = 1 - \rho^2.$$

For fixed values of ρ, the linearized system (1.8) can be represented as a transfer function parameterised by ρ

$$G(s,\rho) = \frac{\tilde{Y}(s)}{\tilde{U}(s)} = \frac{1-\rho^2}{s+1}, \tag{1.9}$$

as illustrated in Fig. 1.5. We refer to $G(s,\rho)$ as a *linearisation family* parameterised by ρ.

The static gain of this transfer function is $1-\rho^2$, which is equal to the slope of $y(x)$ at $y = \rho = \bar{y}$, see Fig. 1.6. From the figure we also see that the slope is 1 when $\rho = 0$, whereas it approaches 0 as $\rho \to \pm 1$, which is in agreement with (1.9).

Quasi-LPV model

An alternative way of bringing system (1.5) into the form of a linear model is to represent it as a *quasi-LPV model*. An important feature of quasi-LPV models - in contrast to models obtained via Jacobian linearisation - is that they provide an *exact representation* of the nonlinear system under consideration. This is achieved by hiding nonlinear terms of the system dynamics behind the introduction of new model parameters. Such a representation of a given nonlinear system is usually not unique, as is now illustrated for system (1.5).

Example 1.1 continued

Consider again the nonlinear first order system (1.5). The dynamics from input u to output y can be represented as

$$\begin{aligned} \dot{x} &= -x + u \\ y &= c(\rho)\,x. \end{aligned} \tag{1.10}$$

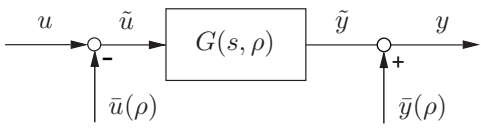


Figure 1.5: Linearized model for fixed ρ

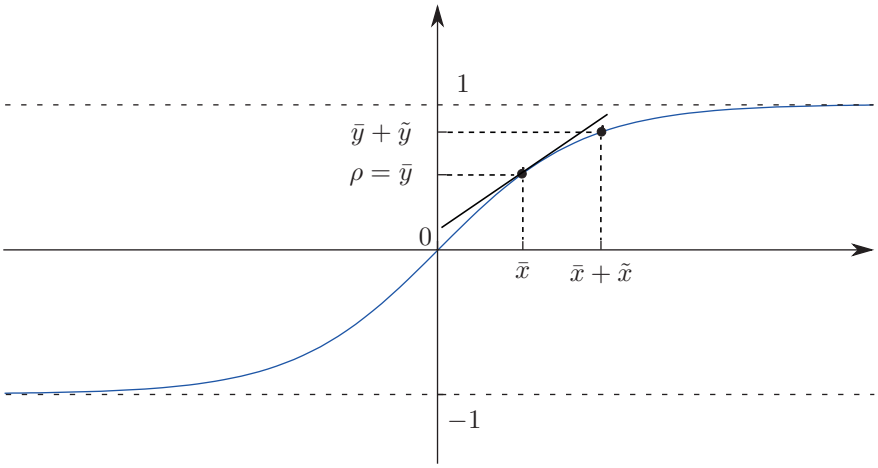


Figure 1.6: Equilibrium about $\rho = \bar{y}$

Note that the output equation has the same form as in (1.8). As before, we are looking for a model that captures the nonlinear dynamics and is parameterised by a suitably selected parameter ρ. Here, however, we are interested in an expression $c(\rho)$ that does not involve an approximation, *i.e.* we need

$$c(\rho)\, x = \tanh x,$$

therefore we choose

$$c(\rho) = \frac{1}{x} \tanh x,$$

(for $x \neq 0$ and taking $c(\rho) = 1$ if $x = 1$), where we still have to define ρ as a function of x. In an LTI model, c in (1.10) would be a constant; the nonlinearity of the system (1.5) is here reflected by the fact that c depends on the state variable x. This dependence is expressed via ρ, and the selection of ρ is not unique: one possible choice is to take

$$\rho = x,$$

which leads to

$$c(\rho) = \frac{1}{\rho} \tanh \rho.$$

On the other hand the choice

$$\rho = \frac{1}{x} \tanh x$$

leads to

$$c(\rho) = \rho.$$

1.4 Classical Gain Scheduling

We now return to the linearized model (1.8), obtained via Jacobian linearisation, and the associated parameterised transfer function $G(s, \rho)$ in (1.9). In this section we consider the

construction of a feedback controller K for this system, and since the plant dynamics G depend on the value of the parameter ρ, we are interested in a controller $K(s, \rho)$ that is also parameterised by ρ. We will refer to time-varying parameters of a plant model that are used to parameterise a controller as *scheduling variables*.

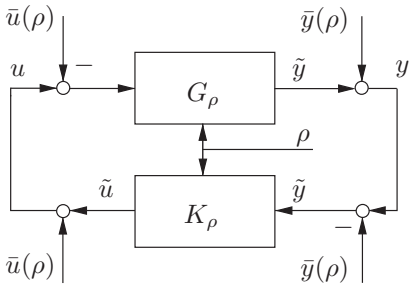


Figure 1.7: Classical gain scheduling: feedback loop

The principle of *classical gain scheduling* is illustrated by the feedback loop shown in Fig. 1.7. Plant and controller are represented as linearisation families, operating on the deviation variables $\tilde{u}$ and $\tilde{y}$. Here G_ρ and K_ρ are used as shorthand notation for $G(s, \rho)$ and $K(s, \rho)$, respectively. Assuming that the equilibrium families $\bar{u}(\rho)$ and $\bar{y}(\rho)$ of plant and controller match, *i.e.* are equal for a given value of ρ, it can be seen from the figure that on each side of the loop the same equilibrium values are added and subtracted, respectively, so that one can equivalently consider a feedback loop that involves only the deviation variables.

When considering this feedback loop, it is assumed that the scheduling variable ρ is known and used to tune the controller to the current operating point of the plant. Note that the linearised plant model and its parameterisation were derived under the assumption that the plant is in equilibrium and ρ is constant. The idea behind classical gain scheduling is to ignore this assumption and treat ρ as a time-varying scheduling variable, that is measured and used to adjust the controller gains on-line.

Tracking

Assuming that the equilibrium families of plant and controller match, we now consider the loop shown in Fig. 1.8, where a reference signal has been introduced that is to be tracked, and we assume that $\tilde{r} = r - \bar{r}$.

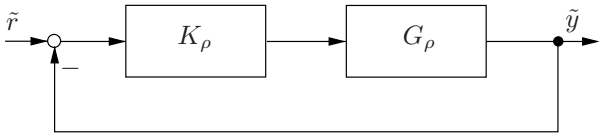


Figure 1.8: Classical gain scheduling: tracking

Since for fixed values of ρ both G_ρ and K_ρ are transfer function models of LTI systems, we can apply linear design techniques to design a family of controllers K_ρ for the family of plants G_ρ. Thus, we consider a PI controller

$$K(s,\rho) = K_P(\rho) + \frac{K_I(\rho)}{s}, \tag{1.11}$$

where we choose

$$K_P(\rho) = \frac{1}{3(1-\rho^2)}, \qquad K_I(\rho) = \frac{1}{1-\rho^2}. \tag{1.12}$$

Note that the controller gains will be large when ρ is close to 1 or -1, *i.e.* when the plant gain is small, and that the controller gains will take their minimum values when $\rho = 0$, *i.e.* when the plant gain takes its maximum value 1. In other words, the controller gains are scheduled to compensate for changes in the plant gain. This is confirmed by calculating the closed-loop transfer function from $\tilde{r}$ to $\tilde{y}$ as

$$G_{cl}(s,\rho) = \frac{G(s,\rho)K(s,\rho)}{1+G(s,\rho)K(s,\rho)} = \frac{\frac{1}{3}s+1}{s^2+\frac{4}{3}s+1}. \tag{1.13}$$

An interesting feature of this closed-loop transfer function, which in general should be itself a linearisation family parameterised by ρ, is that here it is independent of ρ. The choice of scheduled controller gains indeed completely compensates for changes of the plant gain, to the effect that the closed-loop dynamics are independent of the scheduling variable. This is generally the goal of classical gain scheduling: to determine a scheduling policy that adjusts the controller gains on-line such that a desired closed loop behaviour is achieved, independent of the current value of the scheduling variable.

Note that G_{cl} is stable; since it is independent of ρ this implies that it is stable for all $\rho \in \mathcal{P}$.

Implementing Gain-Scheduled Control

The idea behind designing and implementing gain-scheduled controllers for nonlinear plants can be summarised as follows:

- treat ρ as a frozen parameter in the design process, but
- use ρ as a time-varying scheduling signal once the controller is implemented.

Applied to the example considered here, this means to replace the choice $\rho = \bar{y}$ of the scheduling variable, on which the derivation of the linearized model (1.8) was based, by

$$\rho(t) = y(t).$$

In other words, the scheduling variable, which was introduced to represent the value $\bar{y}$ of the plant output in equilibrium, is now taken as the current value $y(t)$ of the plant output not only in steady state but also in transient stages of plant operation.

Letting ζ denote the state variable of the PI controller, a state space model of the controller is then

$$\begin{aligned} \dot{\zeta}(t) &= r(t) - y(t) \\ u(t) &= K_I\Big(y(t)\Big)\zeta + K_P\Big(y(t)\Big)\Big(r - y(t)\Big) \\ &= \frac{3\zeta(t) + r - y(t)}{3\Big(1 - y^2(t)\Big)} \end{aligned} \tag{1.14}$$

If the denominator in the controller output equation was constant, this would be a standard PI controller. The dependence of the denominator on y has the effect that the controller gains are large when the plant gain is small, and small when the plant gain is large.

Hidden Coupling

As pointed out above, the closed-loop transfer function obtained with the scheduled PI controller is stable and independent of the scheduling variable ρ, so we might expect the scheduled closed-loop system to be stable for all admissible values of ρ. However, simulation experiments reveal that the closed-loop system displays unstable behaviour in response to input steps as soon as $\rho(t) = y(t)$ exceeds the value 0.7.

This disappointing closed-loop behaviour is obviously related to the fact that we are using ρ, introduced as an equilibrium value, as a scheduling signal in transient operations. To explain the reduced stability range, we represent the state space model (1.14) of the controller in the form of (1.2) as

$$\begin{aligned} \dot{\zeta}(t) &= f_K(\rho(y(t)), \zeta(t), y(t), r(t)\Big) \\ u(t) &= h_K\Big(\rho(y(t)), \zeta(t), y(t), r(t)\Big). \end{aligned} \tag{1.15}$$

To obtain a linearisation family of the controller, we need to calculate

$$\left.\frac{dh_K}{dy}\right|_\rho = \frac{\partial h_K}{\partial y} + \frac{\partial h_K}{\partial \rho}\frac{d\rho}{dy} \tag{1.16}$$

since we need to account for the dependence of ρ on y. For the controller design only the first term of the right hand side was used. The second term is referred to as *hidden coupling* [1], because it represents a coupling between plant and controller that is caused by the fact that both plant and controller are scheduled by the same scheduling signal $\rho(t) = y(t)$; see Fig. 1.7. It is this second term, which has not been taken into account when designing the controller, that is responsible for the reduced stability range; this is further explored in Exercise 1.1.

Stability of Linear Time-Varying Systems

The example discussed in this chapter raises questions about the stability of linear time-varying systems (the closed-loop system in this example is a linearisation family parameterised by a time-varying scheduling variable $\rho(t) = y(t)$). We know what the

condition for an unforced linear system $\dot{x} = Ax$ is to be stable: the eigenvalues of A must all be strictly inside the left half of the complex plane. What can be said about a parameter-dependent system $\dot{x} = A(\rho)x$? As long as the parameter ρ is fixed, the system matrix is constant and we are considering an LTI system whose eigenvalues can be checked. But what if ρ is allowed to be time-varying? Assume we know that $\rho(t) \in \mathcal{P}\ \forall t \geq 0$, and that $A(\rho)$ has all eigenvalues inside the left half plane for all $\rho \in \mathcal{P}$: can we then conclude that

$$\dot{x} = A(\rho(t))x$$

is stable as long as $\rho(t) \in \mathcal{P}$, i.e. as long as the eigenvalues of $A(\rho(t))$ do not leave the right half plane?

Unfortunately the answer is no: even if $A(\rho(t))$ has its eigenvalues strictly inside the left half plane at all times, is it possible that the the free response to a non-zero initial state becomes unbounded. A well-known example that illustrates this fact, a spring-mass-damper system with time-varying stiffness of the spring, is explored in Exercise 1.2.

Exercises — Chapter 1

Problem 1.1 *(Hidden Coupling)*

Learning Goals

- Understand the source of possible instability in ad hoc gain-scheduling design

References *The original example is contained in [1], which has been reworked for the purpose of this exercise. The paper is also an interesting read for a historical and practical perspective on gain-scheduling.*

Task Description Consider the nonlinear plant (1.5) and its linearisation family (1.9), together with the scheduled controller (1.11), (1.12).

a) Check that the equilibrium families $\bar{u}(\rho)$ and $\bar{y}(\rho)$ of plant and controller match, i.e. that

$$\begin{aligned} f_K\Big(\bar{\zeta}(\rho), \bar{y}(\rho), \bar{r}(\rho)\Big) &= 0, \\ h_K\Big(\bar{\zeta}(\rho), \bar{y}(\rho), \bar{r}(\rho)\Big) &= \bar{u}(\rho), \quad \rho \in [-1, 1]. \end{aligned}$$

b) Recalling that the closed-loop transfer function (1.13) is stable independent of the value of ρ, simulate a step response of the closed-loop system as shown in Fig. 1.8 with

$$\tilde{r}(t) = k\sigma(t), \qquad k = 0.5, 0.6, 0.7, 0.8.$$

What do you observe?

c) Linearise the controller about an equilibrium while taking into account the dependence of ρ on y by using (1.16).

Construct the closed-loop system comprised of the linearized controller and linearized plant and determine its stability for different values of ρ.

d*) Try to modify the implementation of the scheduled controller such that hidden couplings are avoided. Specifically, move the scheduled gain such that it appears *before* the integrator, instead of at the controller output. What do you observe in comparison to the previous implementation? Does the closed-loop also turn unstable for large step changes?

Problem 1.2 *(Inferring LTV from LTI Stability)*

Learning Goals

- Observe that (in-)stability of LTV or LPV systems cannot be inferred from the (in-)stability of frozen-parameter LTI snapshots

References *A similar example is contained in [4], a collection of conference papers compiled during a workshop at the Conference on Decision and Control in Kobe, Japan, 1996. The compilation has long been out of print.*

Task Description Consider the mass-spring-damper system shown in Fig. 1.9, with mass $m = 1$, damping coefficient b and variable stiffness k. The system is governed by the following set of differential equations:

$$\dot{x} = \begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k & -b \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = Ax. \tag{1.17}$$

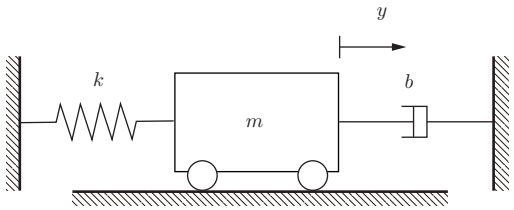


Figure 1.9: Simple mass-spring-damper system

a) Plot the free response to $x(0) = [0\ 1]^\top$ in the phase plane

   i) when $b = 0$, $k = 1/3$,

   ii) when $b = 0$, $k = 3$.

   What can be stated about the stability of the systems?

b) Plot the free response to $x(0) = [0\ 1]^\top$ in the phase plane

   i) when $b = 0.25$, $k = 1/3$,

   ii) when $b = 0.25$, $k = 3$.

   What can be stated about the stability of the systems?

c) Take cases i) and ii) from subtask b) and implement a switching law

$$k = \begin{cases} 3 & \text{, if } y\dot{y} \geq 0 \\ 1/3 & \text{, if } y\dot{y} < 0 \end{cases}$$

Plot the free response to $x(0) = [0\ 1]^\top$ in the phase plane. Also try to turn the switching condition around, such that

$$k = \begin{cases} 1/3 & \text{, if } y\dot{y} \geq 0 \\ 3 & \text{, if } y\dot{y} < 0 \end{cases}$$

What can be observed in terms of stability of the switching system?

d) Explain the observed effect physically. Where does the energy come from?

e) Instead of a switching stiffness k, consider a smoothly time-varying stiffness

$$k(t) = 1 + \frac{1}{2}\cos(2t).$$

Is the system stable? Try to find a smoothly time-varying law that does the opposite (stabilize/destabilize).

Chapter 2

LPV Systems - Stability and Performance

2.1 Lyapunov Stability

The discussion at the end of the last chapter shows that the concept of stability used for LTI systems is not appropriate even for linear time-varying systems, let alone for nonlinear systems, and in this section we will introduce a more general concept of stability. In particular, we will see that - in contrast to LTI systems - it is not meaningful to consider stability as a property of a given system, but as a property of an equilibrium. As an example, consider a pendulum shown in Fig. 2.1.

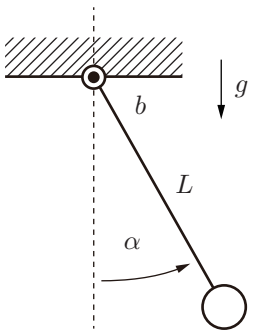


Figure 2.1: Stable and unstable equilibrium of a pendulum

The motion of the pendulum is governed by a nonlinear differential equation; a second order nonlinear state space model

$$\dot{x}(t) = f\Big(x(t)\Big)$$

with angle and angular velocity as state variables, is analysed in Exercise 2.1. It is clear that there are two distinct equilibria (if we ignore periodicity there are in fact infinitely

many): one ($\bar{x}_s$) where the pendulum is at rest when hanging, and the other one ($\bar{x}_u$) where the pendulum is balanced in an upright position. We will refer to the hanging equilibrium as stable and to the upright equilibrium as unstable: when slightly perturbing the pendulum away from its equilibrium state, it will return to it when hanging but move away from it (fall down) when in the upright position.

These notions of stable and unstable equilibria of nonlinear systems are now expressed formally in the following definitions. Consider an unforced nonlinear system

$$\dot{x}(t) = f\Big(x(t)\Big) \tag{2.1}$$

with an equilibrium state $\bar{x}$ such that $f(\bar{x}) = 0$.

Definition 2.1 *$\bar{x}$ is a* stable equilibrium *of system (2.1) if for every $\varepsilon > 0$ there exists a $\delta(\varepsilon) > 0$ such that*

$$\|x(0) - \bar{x}\| < \delta \quad \Rightarrow \quad \|x(t) - \bar{x}\| < \varepsilon \quad \forall t \geq 0.$$

This definition is illustrated in Fig. 2.2. Note that δ is allowed to be a function of ε.

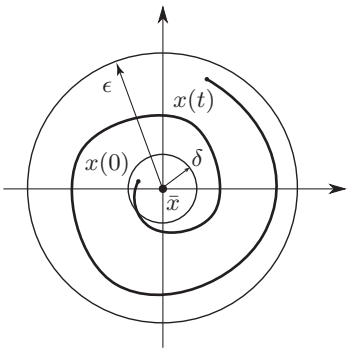


Figure 2.2: Definition of stability

Definition 2.2 *An equilibrium state $\bar{x}$ of system (2.1) is* attractive *if there exists a $\eta > 0$ such that*

$$\|x(0) - \bar{x}\| < \eta \quad \Rightarrow \quad \lim_{t\to\infty} x(t) = \bar{x}$$

This definition is illustrated in Fig. 2.3. Note that a state trajectory is not required to stay within the η-ball at all times.

Definition 2.3 *An equilibrium state of system (2.1) is* asymptotically stable *if it is stable and attractive.*

From now on we will assume without loss of generality $\bar{x} = 0$ for equilibrium states under consideration; if $\bar{x} \neq 0$ it is always possible to apply a coordinate transformation that moves the equilibrium to the origin of the state space.

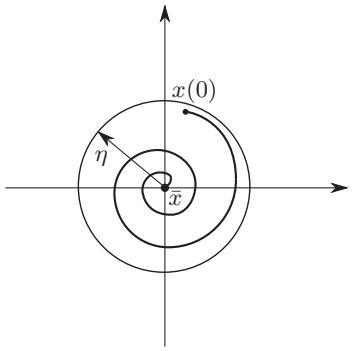


Figure 2.3: Definition of attractiveness

Based on the above definitions of stability and asymptotic stability, we now present a result, due to the Russian Mathematician *A.M. Lyapunov* (1857 - 1918), that provides a sufficient condition for a system to be stable or asymptotically stable, respectively.

For a given $r > 0$, let $\mathcal{B}_r$ denote the ball

$$\mathcal{B}_r = \{x \in \mathbb{R}^n | \ \|x\| < r\}$$

and $\mathcal{B}_r \backslash 0$ the ball $\mathcal{B}_r$ without the origin.

Theorem 2.1 *Consider system (2.1) and assume that $f(0) = 0$. The equilibrium at the origin is stable if there exist a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ and a constant $r > 0$ such that*

$$V(x) > 0 \quad \forall x \neq 0 \tag{2.2}$$

and

$$\dot{V}(x) \leq 0 \quad \forall x \in \mathcal{B}_r. \tag{2.3}$$

The equilibrium is asymptotically stable if

$$\dot{V}(x) < 0 \quad \forall x \in \mathcal{B}_r \backslash 0 \tag{2.4}$$

A proof of this result can be found in [3]. A function V that satisfies the conditions of the theorem is called a *Lyapunov function* for system (2.1). Note that the theorem provides only a sufficient condition; if a Lyapunov function can be found, then stability or asymptotic stability, respectively, is proven. If such a function cannot be found, then no statement is possible. An exception are LTI systems, for which one can show that the existence of a quadratic Lyapunov function $V(x) = x^T P x$ with $P > 0$ is not only sufficient but also necessary for stability [3].

A Lyapunov function of a given system can be interpreted as a generalisation of the total energy of a system; conditions (2.3) and (2.4) then imply that the total system energy decreases or strictly decreases, respectively, along any state trajectory of the system. This is illustrated in Exercise 2.2 for the pendulum example.

Note that Theorem 2.1 presents a local result: required is the existence of a positive constant r such that the conditions of the theorem are satisfied inside a ball of radius r. If the conditions hold for $r \to \infty$, the equilibrium is called *globally stable* or *globally asymptotically stable*, respectively. For LTI systems one can show that if the origin is (asymptotically) stable, it is globally (asymptotically) stable; this is the reason why for LTI systems we consider stability as a property of the system.

2.2 Linear Parameter-Varying Systems

In this section we will formally introduce the concept of a linear parameter-varying (LPV) system. For the system in Example 1.1 it was illustrated in Chapter 1 that it is possible to represent a nonlinear system in the form of a quasi-LPV system; this is the basis for using analysis and synthesis results on LPV systems to analyse and control nonlinear systems.

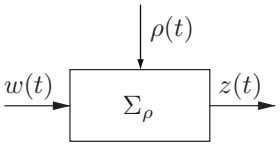


Figure 2.4: LPV system

An LPV system is a dynamic system that can be represented in the form

$$\Sigma_\rho : \quad \left\{ \begin{array}{rcl} \dot{x}(t) & = & A\Big(\rho(t)\Big)\, x(t) + B\Big(\rho(t)\Big)\, w(t) \\ z(t) & = & C\Big(\rho(t)\Big)\, x(t) + D\Big(\rho(t)\Big)\, w(t) \end{array} \right. \tag{2.5}$$

where ρ is a time-varying vector of scheduling variables, and $A(\cdot)$, $B(\cdot)$, $C(\cdot)$ and $D(\cdot)$ are continuous functions of ρ. As before, we assume that

$$\rho(t) \in \mathcal{P} \subset \mathbb{R}^{n_\rho} \quad \forall t \geq 0$$

where $\mathcal{P}$ is a given compact set. Note that the continuity assumption together with compactness of $\mathcal{P}$ implies that A, B, C and D are bounded on $\mathcal{P}$. Fig. 2.4 shows a block diagram of an LPV system.

The LPV system Σ_ρ in (2.5) is linear in x and w, but depends on the time-varying parameter ρ. In Example 1.1, see (1.10), a nonlinear system was represented as a quasi-LPV system in this form; this requires that ρ in turn depends on the state vector x and/or on the input u.

The set $\mathcal{P}$ has been introduced as a compact subset of the Euclidean vector space $\mathbb{R}^{n_\rho}$. It will be convenient to define also a set

$$\mathcal{F}_\mathcal{P} = \Big\{ \rho(t) \in C^1(\mathbb{R}^+, \mathbb{R}^{n_\rho}) \,|\, \rho(t) \in \mathcal{P} \quad \forall t \geq 0 \Big\} \tag{2.6}$$

where the notation $C^1(\mathbb{R}^+, \mathbb{R}^{n_\rho})$ is used to denote the set of continuously differentiable mappings from $\mathbb{R}^+$ to $\mathbb{R}^{n_\rho}$. Note that whereas $\mathcal{P}$ is a subset of the finite-dimensional vector space $\mathbb{R}^{n_\rho}$, $\mathcal{F}_\mathcal{P}$ is a subset of a signal space, *i.e.* the elements of $\mathcal{F}_\mathcal{P}$ are not admissible scheduling *vectors* but admissible scheduling *trajectories*. We will use the notation

$$\Sigma_\mathcal{P} = \{\Sigma_\rho \,|\, \rho \in \mathcal{F}_\mathcal{P}\} \tag{2.7}$$

to represent an LPV system whose dynamic properties are determined by (2.5), and on which in addition the constraint is imposed that the scheduling trajectories $\rho(t)$ are continuously differentiable w.r.t. time and confined to the compact set $\mathcal{P}$.

We will also consider LPV systems with constraints not only on the values of the scheduling vector $\rho(t)$ but also on the rate of change $\dot{\rho}(t)$. In this case we assume that

$$|\dot{\rho}_i| \leq \bar{\nu}_i \quad \forall t \geq 0, \quad i = 1, \ldots, n_\rho. \tag{2.8}$$

Defining a vector

$$\nu = \begin{bmatrix} \nu_1 & \nu_2 & \ldots & \nu_{n_\rho} \end{bmatrix}^T$$

and the set

$$\mathcal{V} = \{\nu \in \mathbb{R}^{n_\rho} \,|\, |\nu_i| \leq \bar{\nu}_i, \quad i = 1, \ldots, n_\rho\}, \tag{2.9}$$

we can modify the set (2.6) and define a set of scheduling trajectories

$$\mathcal{F}_\mathcal{P}^\mathcal{V} = \Big\{\rho(t) \in C^1(\mathbb{R}^+, \mathbb{R}^{n_\rho}) \,|\, \rho(t) \in \mathcal{P},\ \dot{\rho}(t) \in \mathcal{V} \quad \forall t \geq 0\Big\} \tag{2.10}$$

that satisfy constraints on the values and the rate of change of the scheduling variables. We let $\Sigma_\mathcal{P}^\mathcal{V}$ denote an LPV system whose scheduling trajectories are confined to this set. The set $\mathcal{F}_\mathcal{P}$ represents scheduling trajectories without rate limit.

2.3 Classes of Linear Systems

An LPV system as defined in (2.5) is a linear system, in the sense that both state and output equation are linear in the input signal and in the state vector. The simplest class of linear systems consists of LTI systems

$$\dot{x}(t) = Ax(t) + Bu(t), \qquad y(t) = Cx(t) + Du(t),$$

with constant model matrices A, B, C and D. A more general class of linear systems is obtained when the model matrices are allowed to vary over time, resulting in *linear time-varying* (LTV) systems

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \qquad y(t) = C(t)x(t) + D(t)u(t).$$

Note that an LTV system is different from an LPV system

$$\dot{x}(t) = A\Big(\rho(t)\Big)\, x(t) + B\Big(\rho(t)\Big)\, u(t), \qquad y(t) = C\Big(\rho(t)\Big)\, x(t) + D\Big(\rho(t)\Big)\, u(t),$$

where the model matrices do not explicitly depend on time, but on a scheduling vector ρ which in turn is a function of time. One can view both LTI systems and LTV systems as special cases of LPV systems: an LTI system can be interpreted as an LPV system with a constant scheduling trajectory, whereas an LTV system can be seen as an LPV system with a particular choice of a time-varying scheduling trajectory. This is because for a given LTV system $\Big(A(t), B(t), C(t), D(t)\Big)$ it is always possible to construct a scheduling vector $\rho(t)$ that includes all time-varying elements of the model matrices, and select the scheduling trajectory that results in the given LTV model. (For viewing a given LTV system as a special case of an LPV system $\Sigma_{\mathcal{P}}$ pr $\Sigma_{\mathcal{P}}^{\mathcal{V}}$, it is however required that the time-varying model matrices of the LTV system satisfies the constraints imposed by $\mathcal{P}$ and $\mathcal{V}$.)

2.4 Stability of LPV Systems

In this chapter we will extend the notions of stability and performance defined for LTI systems to LPV systems, and we begin with stability. We will do this by applying the definitions and the stability result introduced in Section 2.1 for nonlinear systems to LPV systems. This will also provide a basis for analysing quasi-LPV systems, which are used to represent nonlinear systems.

Consider the state equation of an unforced LPV system

$$\dot{x}(t) = A\Big(\rho(t)\Big)x(t) \tag{2.11}$$

Recalling Definition 1.1, it is obvious that $\bar{x} = 0$ is an equilibrium of this time-varying system. Note that for $\bar{x}$ to be an equilibrium, it is required that $A\Big(\rho(t)\Big)\bar{x}$ is zero for all $t \geq 0$. If there exists any time instant $t_s \geq 0$ at which $A\Big(\rho(t_s)\Big)$ is non-singular, then it follows that $\bar{x} = 0$ is the single unique equilibrium of the system, and from now on we shall, unless stated otherwise, make that assumption.

It is also clear that, for a given set $\mathcal{P}$ of admissible scheduling variables, $\bar{x} = 0$ is an equilibrium for all $\rho \in \mathcal{P}$. In this section we will be looking for a condition under which the equilibrium $\bar{x} = 0$ of system (2.11) is stable for all $\rho(t) \in \mathcal{F}_{\mathcal{P}}$, or - if rate bounds are known - for all $\rho(t) \in \mathcal{F}_{\mathcal{P}}^{\mathcal{V}}$. If we can establish that this condition is satisfied, we will say that system (2.11) is *stable over* $\mathcal{P}$, or *stable over* $(\mathcal{P} \times \mathcal{V})$, respectively.

LTI Systems

Before we derive a condition for system (2.11) to be stable, we briefly review a stability condition for LTI systems based on a quadratic Lyapunov function. To establish stability of the system

$$\dot{x} = Ax,$$

we define the Lyapunov function candidate $V(x) = x^T P x$ with $P > 0$. The condition that

$\dot{V}(x) < 0 \; \forall x \neq 0$ yields

$$\dot{V}(x) = \dot{x}^T P x + x^T P \dot{x} = x^T (A^T P + PA) x < 0 \quad \forall x \neq 0.$$

Using Theorem 2.1, we conclude that V is a Lyapunov function and the system is globally asymptotically stable [1] if there exists a $P > 0$ such that

$$A^T P + PA < 0. \tag{2.12}$$

For LTI systems it can in fact be shown that this condition (a linear matrix inequality in P) is not only sufficient but also necessary for stability, see e.g. [3].

LPV Systems

Now we apply this approach to system (2.11) and define again $V(x) = x^T P x$ with $P > 0$. The condition that $\dot{V}(x) < 0 \; \forall x \neq 0$ then yields the following result.

Theorem 2.2 *System (2.11) is stable over $\mathcal{P}$ if there exists a $P > 0$ such that*

$$A^T(\rho) P + PA(\rho) < 0 \quad \forall \rho \in \mathcal{P}. \tag{2.13}$$

Quasi-LPV Systems

When the model (2.11) is used to represent a nonlinear system as quasi-LPV system, i.e. when ρ depends on x, then stability over $\mathcal{P}$, established by condition (2.13), can be related to the stability condition (2.4) of Theorem 2.1 as follows. Assume that there exists an $r > 0$ such that

$$x \in \mathcal{B}_r \quad \Rightarrow \quad \rho(x) \in \mathcal{P},$$

then (2.13) implies that $\bar{x} = 0$ is a stable equilibrium of (2.11).

In contrast to the above result for LTI systems, the condition of Theorem 2.2 is only sufficient and not necessary. An LPV system that satisfies this condition is said to be *quadratically stable*. An important feature of quadratic stability is that the matrix P in the quadratic Lyapunov function is constant. Demanding that P is constant is however unnecessarily restrictive - if a constant positive definite solution P to (2.13) does not exist, it may still be possible to find a matrix-valued function $P(\rho)$ that proves stability of the system over $(\mathcal{P} \times \mathcal{V})$, with some constraints $\mathcal{V}$ imposed on the rate of parameter variations. A stability condition less conservative than quadratic stability can be derived as follows. Define a quadratic Lyapunov function

$$V(x) = x^T P(\rho) x, \tag{2.14}$$

where $P \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n \times n})$, and $\mathcal{S}^{n \times n}$ is the set of $n \times n$ real symmetric matrices (this means that P is a matrix-valued, symmetric, continuously differentiable function of ρ). Imposing the condition that

$$P(\rho) > 0 \quad \forall \rho \in \mathcal{P} \tag{2.15}$$

[1] From now on we will - unless stated otherwise - use the term stable to mean asymptotically stable

and that $\dot{V}(x) < 0 \; \forall x \neq 0$ yields

$$\begin{aligned} \dot{V}(x) &= \dot{x}^T P(\rho) x + x^T \dot{P}(\rho) x + x^T P(\rho) \dot{x} \\ &= x^T \Big(A(\rho)^T P(\rho) + P(\rho) A(\rho) + \dot{P}(\rho) \Big) x < 0 \quad \forall x \neq 0, \; \forall \rho \in \mathcal{P}. \end{aligned}$$

Because the Lyapunov matrix is now time-varying (via its dependence on $\rho(t)$), the left hand side of the above inequality now includes the term $\dot{P}(\rho)$; with $\rho \in \mathbb{R}^{n_\rho}$ we have

$$\frac{d}{dt} P\Big(\rho(t)\Big) = \sum_{i=1}^{n_\rho} \dot{\rho}_i \frac{\partial P}{\partial \rho_i}.$$

We can summarise the above results as follows.

Theorem 2.3 *System (2.11) is stable over $(\mathcal{P} \times \mathcal{V})$ if there exists a $P(\rho) \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n \times n})$ such that (2.15) holds and*

$$A^T(\rho) P(\rho) + P(\rho) A(\rho) + \sum_{i=1}^{n_\rho} \dot{\rho}_i \frac{\partial P}{\partial \rho_i} < 0 \qquad \forall (\rho, \dot{\rho}) \in (\mathcal{P} \times \mathcal{V}) \tag{2.16}$$

Quasi-LPV Systems

When the model (2.11) is used to represent a nonlinear system as quasi-LPV system, i.e. when ρ depends on x, then stability over $\mathcal{P} \times \mathcal{V}$, established by condition (2.16), can be related to the stability condition (2.4) of Theorem 2.1 as follows. Assume that there exists an $r > 0$ such that

$$x \in \mathcal{B}_r \quad \Rightarrow \quad \rho(x) \in \mathcal{P} \quad \text{and} \quad \dot{\rho}(x) \in \mathcal{V},$$

then (2.16) implies that $\bar{x} = 0$ is a stable equilibrium of (2.11).

Bounds on the Rate of Parameter Variation

If the scheduling vector $\rho(t)$ is allowed to change arbitrarily fast, i.e. if $\dot{\rho} \to \infty$, then only constant solutions P to (2.16) are possible, because otherwise for any non-zero value of $\dot{\rho}_i$ the third term on the left hand side of the inequality can take arbitrarily large positive values; only a constant P can accommodate an infinite rate of parameter variation. On the other hand, this discussion shows that quadratic stability guarantees stability in the presence of arbitrarily fast variations. When upper bounds on the rate of parameter variation are known, this turns into a source of conservatism, because stability is guaranteed for stronger conditions than needed. The smaller the bounds on the rate are (meaning a small $\mathcal{V}$), the better are the chances for finding a solution $P(\rho)$ that satisfies (2.16) and (2.15). Later when discussing controller synthesis we will see that Theorem 2.3 offers a way of exploiting known bounds on the rate of parameter variation to achieve improved performance, or of finding a solution at all when the problem is infeasible with constant Lyapunov functions.

Testing the Stability Condition

A practical difficulty when testing conditions (2.16) and (2.15) is that they have to hold for *all admissible values* of ρ and $\dot{\rho}$. A heuristic approach that is used in practice is to check the conditions on a suitably dense grid over $(\mathcal{P} \times \mathcal{V})$. For a given grid on $\mathcal{P}$, the partial derivatives can be approximated using a finite difference scheme. In this case inequalities (2.16) and (2.15) are affine in $P(\rho)$ for fixed values of ρ and $\dot{\rho}$, and can be solved by solving an LMI feasibility problem; this has to be done however in each grid point. A simplification is offered by the fact that (2.16) is affine in $\dot{\rho}_i$. As a consequence, we do not need to grid over the range $-\bar{\nu}_i \leq \dot{\rho}_i \leq \bar{\nu}_i$; it is sufficient to check (2.16) at the boundaries $-\bar{\nu}_i$ and $\bar{\nu}_i$ for each grid point on $\mathcal{P}$. This is illustrated in Fig. 2.5 for the case where ρ is scalar.

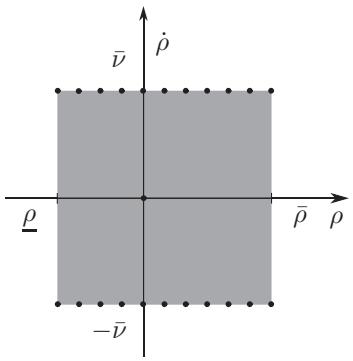


Figure 2.5: Gridding with a scalar scheduling variable ρ

Gain-Scheduled State Feedback

To illustrate how the analysis conditions (2.16) and (2.15) can be used to construct a stabilising gain-scheduled controller for a plant represented as LPV model, we will now consider a gain-scheduled state feedback control law. We first briefly review the state feedback synthesis problem for LTI systems (see Lecture Notes *Optimal and Robust Control*, Chapter 19). Thus, consider the LTI plant

$$\dot{x} = Ax + Bu$$

and LTI state feedback $u = Fx$. We can use (2.12) as an LMI condition for closed-loop stability when we replace A by $A + BF$: the state feedback gain F will be stabilising if and only if there exists a $P > 0$ such that

$$(A + BF)^T P + P(A + BF) < 0.$$

This is however not an LMI in F and P. A linearising change of variables is not possible in this form, but becomes possible when we replace $A + BF$ by its transpose: then we can define a new variable $Y = FP$ to obtain the condition

$$AP + PA^T + BY + Y^T B^T < 0,$$

which can be solved as an LMI problem for the variables P and Y and yields a parameterisation of all stabilising state feedback gains as

$$F = YP^{-1}.$$

Replacing $A + BF$ by its transpose in a stability condition for LTI systems is justified by the fact that this does not change the eigenvalues. This reasoning based on eigenvalues is not valid when we consider gain-scheduled state feedback for an LPV systems; the following example illustrates how a linearising change can be performed.

Example 2.1 *For the system*

$$\dot{x} = (A_0 + \rho A_1)\, x + Bu \tag{2.17}$$

where $\rho \in [0, 1]$ and $|\dot{\rho}| \leq \bar{\nu}$ (this defines the set $\mathcal{P} \times \mathcal{V}$), we consider the problem of finding a stabilising gain-scheduled state feedback law

$$u = F(\rho)x. \tag{2.18}$$

Features of the LPV system (2.17) that simplify the synthesis are (i) the affine dependence of $A(\rho)$ on ρ and (ii) the fact that the input matrix B does not depend on ρ. Substituting (2.18) in (2.17) yields the closed-loop system

$$\dot{x} = \Big(A_0 + \rho A_1 + BF(\rho)\Big)\, x \tag{2.19}$$

We can now apply Theorem 2.3 to obtain a condition for (2.19) to be stable over ($\mathcal{P} \times \mathcal{V}$): a sufficient condition for stability is that there exists a matrix-valued, continuously differentiable function $P(\rho)$ such that

$$\Big(A_0 + \rho A_1 + BF(\rho)\Big)^T P(\rho) + P(\rho)\Big(A_0 + \rho A_1 + BF(\rho)\Big) + \dot{\rho}\, \frac{\partial P(\rho)}{\partial \rho} < 0.$$

for all $(\rho, \dot{\rho}) \in (\mathcal{P} \times \mathcal{V})$. As in the LTI case, this condition is not affine in $P(\rho)$ and $F(\rho)$. For a linearising change of variables we need to rearrange the closed-loop Lyapunov inequality. In contrast to the LTI case, here we cannot simply replace the closed-loop system matrix by its transpose, since stability is not determined by eigenvalues. But we can apply a congruence transformation: multiplying the above inequality from left and right by $P^{-1}(\rho)$ yields

$$P^{-1}(\rho)\Big(A_0 + \rho A_1 + BF(\rho)\Big)^T + \Big(A_0 + \rho A_1 + BF(\rho)\Big)P^{-1}(\rho) + \dot{\rho}\, P^{-1}(\rho)\frac{\partial P}{\partial \rho}P^{-1}(\rho) < 0.$$

Note that this is indeed a congruence transformation because P is symmetric. Using

$$P^{-1}\frac{\partial P}{\partial \rho}P^{-1} = -\frac{\partial P^{-1}}{\partial \rho},$$

renaming the variable P^{-1} *by* P *and defining*

$$Y(\rho) = F(\rho)P(\rho), \tag{2.20}$$

we can conclude that the closed-loop system (2.19) is stable over $\mathcal{P} \times \mathcal{V}$ *if there exist* $Y(\rho)$ *and* $P(\rho)$ *such that*

$$P(\rho) > 0 \tag{2.21}$$

and

$$P(\rho)(A_0 + \rho A_1)^T + (A_0 + \rho A_1)P(\rho) + Y^T(\rho)B^T + BY(\rho) - \dot{\rho}\,\frac{\partial P}{\partial \rho} < 0 \tag{2.22}$$

for all $(\rho, \dot{\rho}) \in \mathcal{P} \times \mathcal{V}$.

We will now discuss two different ways of using this condition for gain-scheduled stabilising state feedback gain: by gridding, and by imposing a structure on the solution.

1. Gridding: Divide the admissible range $[-1, 1]$ *for* ρ *into* N *intervals of width* h *(see Fig. 2.6), and define*

$$\rho_k = kh - 1, \qquad P_k = P(\rho_k), \qquad Y_k = Y(\rho_k).$$

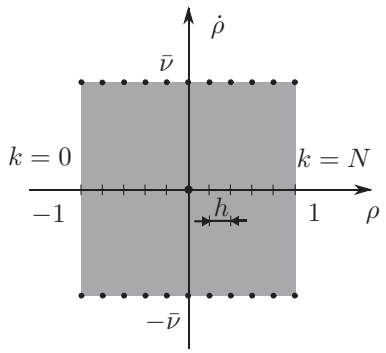


Figure 2.6: Example 2.1, gridding

Using a finite difference approximation for the partial derivative, the infinite family of conditions (2.21) and (2.22), which must hold on $\mathcal{P} \times \mathcal{V}$, *can then be approximated by a finite number of conditions*

$$P_k > 0$$

and

$$(A_0 + \rho_k A_1)P_k + P_k(A_0 + \rho_k A_1)^T + BY_k + Y_k^T B^T \ \pm \bar{\nu}\frac{1}{h}(P_{k+1} - P_k) < 0$$

indexed by $k = 0, 1, \ldots, N-1$. *The* $\pm$ *in front of the last term on the left hand side means that this inequality must be checked at each value of* ρ_k *for* $\bar{\nu}$ *and* $-\bar{\nu}$. *Note that the conditions are linear in the variables* P_k, P_{k+1} *and* Y_k, *which can thus be obtained by solving a finite collection of LMI conditions. Using these values of* P_k *and* Y_k, *we can*

construct a lookup-table for the values of the scheduled state feedback gain $F(\rho_k)$ from (2.20) as $F(\rho_k) = Y_k P_k^{-1}$.

2. Imposing a structure: An alternative way of turning the infinite family of conditions (2.21) and (2.22) into a finite number of LMIs is to impose a structure on the functional dependence of the variables P and Y on ρ. In this approach one typically copies the functional dependence of the model matrices on ρ into the variables; thus here we assume

$$P(\rho) = P_0 + \rho P_1$$

and

$$Y(\rho) = Y_0 + \rho Y_1.$$

Substituting this into (2.22) yields

$$\begin{aligned}(A_0 + \rho A_1)(P_0 + \rho P_1) + (P_0 + \rho P_1)(A_0 + \rho A_1)^T + \\ B(Y_0 + \rho Y_1) + (Y_0 + \rho Y_1)^T B^T - \dot{\rho} P_1 < 0\end{aligned} \tag{2.23}$$

which must hold together with (2.21) for all $(\rho, \dot{\rho}) \in \mathcal{P} \times \mathcal{V}$.

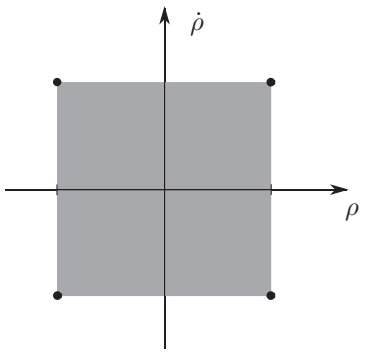


Figure 2.7: Example 2.1, imposing a structure

Note that condition (2.23) is linear in the variables P_0, P_1, Y_0 and Y_1 and can thus be solved as LMI problem. The fact that it is linear in $\dot{\rho}$ means that it is sufficient to check for each value of ρ only the extreme values $\dot{\rho} = \pm\bar{\nu}$. Unfortunatley the condition is not linear in ρ: expanding the brackets shows that the left hand side of (2.23) includes the term

$$\rho^2(A_1 P_1 + P_1 A_1^T).$$

Thus, one still has to grid over ρ. A way of avoiding gridding altogether is to impose the additional constraint

$$A_1 P_1 + P_1 A_1^T \geq 0. \tag{2.24}$$

This renders the quadratic dependence on ρ convex, so that it suffices to check that (2.21) and (2.22) are satisfied at the extreme values of ρ and $\dot{\rho}$ (see Fig. 2.7). Imposing the extra constraint (2.24) introduces however additional conservatism: this condition is not required for stability but only for the convenience of reducing the number of LMIs to be solved.

This example is further explored in Exercise 2.3.

2.5 Performance of LPV Systems

Having established stability conditions for LPV systems, we will now define performance and establish conditions that will later be turned into synthesis tools for LPV controllers. More precisely, we will extend the notion of the $\mathcal{H}_\infty$ norm for LTI systems to LPV systems by taking an induced system norm as performance measure.

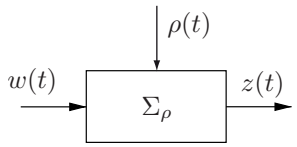


Figure 2.8: LPV system

Consider again an LPV system Σ_ρ as defined in (2.5) and shown in Fig. 2.8. We will use the notation introduced in Section 2.2 and let $\Sigma_{\mathcal{P}}^{\mathcal{V}}$ denote a system with dynamics defined by (2.5), and admissible scheduling trajectories $\rho(t)$ confined to the set $\mathcal{F}_{\mathcal{P}}^{\mathcal{V}}$ defined in (2.6).

The $\mathcal{L}_2$-norm of a vector-valued signal $x : \mathbb{R}^+ \to \mathbb{R}^n$ is defined as

$$\|x(t)\|_2 = \left(\int_0^\infty x(t)^T x(t)\, dt \right)^{\frac{1}{2}}. \tag{2.25}$$

The associated space $\mathcal{L}_2^n$ of n-dimensional signal vectors is

$$\mathcal{L}_2^n = \left\{ x : \mathbb{R}^+ \to \mathbb{R}^N \mid \|x(t)\|_2 < \infty \right\}.$$

Since the $\mathcal{L}_2$-norm is the only signal norm considered here, we will usually drop the subscript and let $\|x(t)\|$ denote the $\mathcal{L}_2$-norm of $x(t)$.

Definition 2.4 *Assume $x(0) = 0$. The induced $\mathcal{L}_2$-norm of the LPV system $\Sigma_{\mathcal{P}}^{\mathcal{V}}$ is*

$$\left\| \Sigma_{\mathcal{P}}^{\mathcal{V}} \right\|_{\mathcal{L}_2} = \sup_{\rho \in \mathcal{F}_{\mathcal{P}}^{\mathcal{V}}} \sup_{0 \neq w \in \mathcal{L}_2} \frac{\|z(t)\|}{\|w(t)\|}. \tag{2.26}$$

In the special case where $\Sigma_{\mathcal{P}}^{\mathcal{V}}$ is an LTI system (and therefore there is no scheduling variable to be constrained to an admissible set), this definition reduces to the familiar $\mathcal{H}_\infty$-norm. For an LPV system with scheduling trajectories confined to a compact set, the definition of the induced $\Sigma_{\mathcal{P}}^{\mathcal{V}}$ involves taking the supremum over all admissible scheduling trajectories, (i.e. considering the worst-case admissible scheduling trajectory).

Review: Induced $\mathcal{L}_2$-Norm of LTI Systems

Before we derive a condition for an upper bound on the $\mathcal{L}_2$-norm of an LPV system, we briefly review the LTI case. Thus, consider the system

$$T: \quad \begin{cases} \dot{x}(t) &= Ax(t) + Bw(t) \\ z(t) &= Cx(t) + Dw(t) \end{cases} \tag{2.27}$$

which is shown in Fig. 2.9.

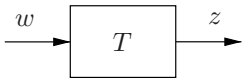


Figure 2.9: LTI system

An upper bound on the H_∞-norm of $T(s)$ is provided by the following result (compare Lecture Notes *Optimal and Robust Control*, Theorem 18.3).

Theorem 2.4 *$T(s)$ is stable and $\|T\|_\infty < \gamma$ if and only if there exists a positive definite matrix P that satisfies*

$$\begin{bmatrix} A^TP + PA & PB & C^T \\ B^TP & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0. \tag{2.28}$$

Note that stability follows immediately from negative definiteness of the (1,1) block on the left hand side of (2.28). The proof of (sufficiency of) this result is based on the idea of "using a Lyapunov function to bound the $\mathcal{L}_2$-gain" of a system: with a function $V(x) = x^TPx$ where $P > 0$, it follows from

$$\frac{dV(x)}{dt} + \frac{1}{\gamma}\, z^Tz - \gamma w^Tw < 0 \tag{2.29}$$

that $\|T\|_\infty < \gamma$. This can be seen by integrating (2.29) from 0 to ∞ and assuming $x(0) = 0$. The LMI condition (2.28) is then obtained by substituting

$$\begin{aligned} \frac{dV(x)}{dt} &= \dot{x}^TPx + x^TP\dot{x} \\ &= (Ax + Bw)^TPx + x^TP(Ax + Bw) \end{aligned} \tag{2.30}$$

and

$$z^Tz = (Cx + Dw)^T(Cx + Dw)$$

in (2.29), rewriting the left hand side as a quadratic form in $[x^T \;\; w^T]$, and using a Schur argument to obtain the 3×3 block matrix in (2.28).

LPV Systems

Now consider the LPV system

$$\Sigma_\rho: \quad \begin{cases} \dot{x}(t) &= A\big(\rho(t)\big)\, x(t) + B\big(\rho(t)\big)\, w(t) \\ z(t) &= C\big(\rho(t)\big)\, x(t) + D\big(\rho(t)\big)\, w(t) \end{cases} \tag{2.31}$$

Defining $V(x) = x^T P(\rho) x$ with $P > 0$, we obtain

$$\frac{dV(x)}{dt} = \dot{x}^T P(\rho) x + x^T \dot{P}(\rho) x + x^T p(\rho) \dot{x}.$$

Compared with (2.30) we have the extra term

$$\dot{P}(\rho) = \sum_{i=1}^{n_\rho} \dot{\rho}_i \frac{\partial P}{\partial \rho_i}.$$

Assuming that the admissible trajectories are confined to a set $\mathcal{F}_{\mathcal{P}}^{\mathcal{V}}$, and following the same reasoning as in the LTI case, we obtain a sufficient condition (necessity is lost for time-varying systems) for γ-performance of an LPV system.

Theorem 2.5 *$\Sigma_{\mathcal{P}}^{\mathcal{V}}$ is stable and $\|\Sigma_{\mathcal{P}}^{\mathcal{V}}\|_{\mathcal{L}_2} < \gamma$ if there exists a continuously differentiable matrix-valued function P that satisfies*

$$\begin{bmatrix} A(\rho)^T P(\rho) + P(\rho) A(\rho) + \sum_{i=1}^{n_\rho} \dot{\rho}_i \frac{\partial P}{\partial \rho_i} & P(\rho) B(\rho) & C^T(\rho) \\ B(\rho)^T P(\rho) & -\gamma I & D(\rho)^T \\ C(\rho) & D(\rho) & -\gamma I \end{bmatrix} < 0 \tag{2.32}$$

and $P(\rho) > 0 \; \forall (\rho, \dot{\rho}) \in \mathcal{P} \times \mathcal{V}$.

Note that by considering frozen values of ρ and $\dot{\rho}$, condition (2.32) can - like the stability condition (2.16) - be turned into a finite collection of LMIs on a grid over $\mathcal{P}$ (when approximating the partial derivatives by finite differences). The statement "$\Sigma_{\mathcal{P}}^{\mathcal{V}}$ is stable" is used here as equivalent to "Σ_ρ is stable on $\mathcal{P} \times \mathcal{V}$".

Quadratic $\mathcal{L}_2$-Performance

A special case of the above result is obtained by imposing the constraint that P is constant. In this case the partial derivative term in the (1,1) block of the above matrix inequality disappears.

Corollary 2.1 *$\Sigma_{\mathcal{P}}$ is stable and $\|\Sigma_{\mathcal{P}}\|_{\mathcal{L}_2} < \gamma$ if there exists a $P > 0$ that satisfies*

$$\begin{bmatrix} A(\rho)^T P + P A(\rho) & P B(\rho) & C^T(\rho) \\ B(\rho)^T P & -\gamma I & D(\rho)^T \\ C(\rho) & D(\rho) & -\gamma I \end{bmatrix} < 0. \tag{2.33}$$

An LPV system that satisfies the condition of Corollary 2.1 with constant P is said to have *quadratic $\mathcal{L}_2$-performance* γ. In this case the performance level γ is achieved even when $\dot{\rho} \to \infty$. For this reason, conditions imposing quadratic performance are more conservative than the conditions of Theorem 2.5, which allow to exploit bounds on the rate of parameter variation to achieve a better performance.

Next we illustrate the conservatism incurred by the constraint that P be constant with an example. In particular, we will consider an unstable LPV system for which a gain-scheduled

state feedback gains exists that makes the closed-loop system *stable*, but for which no gain-scheduled state feedback gain exists that makes the closed-loop system *quadratically stable*. We first define *quadratic stabilisability*.

Definition 2.5 *An LPV system*

$$\dot{x} = A(\rho)x + B(\rho)u$$

is quadratically stabilisable over $\mathcal{P}$ if there exist a continuous matrix-valued function $F(\rho)$ and $P > 0$ such that

$$\Big(A(\rho) + B(\rho)F(\rho)\Big)^T P + P\Big(A(\rho) + B(\rho)F(\rho)\Big) < 0 \quad \forall \rho \in \mathcal{P}.$$

Example 2.2 *This example is taken from [5]. Here we will show that the following system is not quadratically stabilisable; that a stabilising gain-scheduled state feedback law exists nevertheless will be established in Exercise 3.1.*

We consider a fourth-order system with two input and two output channels, see Fig. 2.10. Both input channels include first-order low-pass filters with fixed time constants τ; the filtered signals v_1 and v_2 are then passing through a rotation matrix

$$R(\rho) = \begin{bmatrix} \cos\rho & \sin\rho \\ -\sin\rho & \cos\rho \end{bmatrix}$$

where $\rho \in [-\pi, \pi]$.

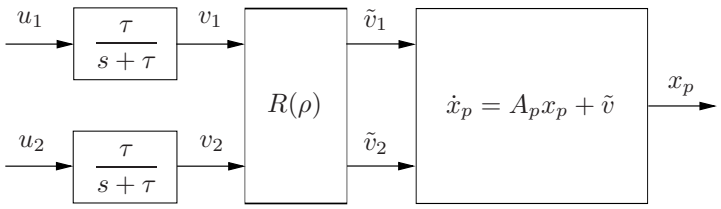


Figure 2.10: System for Example 2.2

The rotated vector $\tilde{v} = R(\rho)v$ is then taken as input to second order LTI dynamics (A_p, I), as shown in the Figure. The overall system can be represented as

$$\begin{bmatrix} \dot{x}_p \\ \dot{v} \end{bmatrix} = \begin{bmatrix} A_p & R(\rho) \\ 0 & -\tau I \end{bmatrix} \begin{bmatrix} x_p \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \tau I \end{bmatrix} u,$$

or in a more compact form as

$$\dot{x} = A(\rho)x + Bu.$$

We will now show that if A_p is unstable, then $\Big(A(\rho), B\Big)$ is not quadratically stabilisable over $[-\pi, \pi]$, whereas Exercise 3.1 will establish that a stabilising state feedback law does exist when the rate of parameter variation is bounded.

We prove the above claim by contradiction. Thus, assume that $\Big(A(\rho), B\Big)$ *is quadratically stabilisable over* $[-\pi, \pi]$*. Then there exist a* $P > 0$ *and a continuous function* $F : [-\pi, \pi] \to \mathbb{R}^{2\times 4}$ *such that*

$$\Big(A(\rho) + BF(\rho)\Big)^T P + P\Big(A(\rho) + BF(\rho)\Big) < 0 \quad \forall \rho \in [-\pi, \pi].$$

In particular, there exist $F(0)$ *and* $F(\pi)$ *such that*

$$\Big(A(0) + BF(0)\Big)^T P + P\Big(A(0) + BF(0)\Big) < 0 \tag{2.34}$$

and

$$\Big(A(\pi) + BF(\pi)\Big)^T P + P\Big(A(\pi) + BF(\pi)\Big) < 0. \tag{2.35}$$

Adding inequalities (2.34) and (2.35) and dividing by 2 yields

$$(\tilde{A} + B\tilde{F})^T P + P(\tilde{A} + B\tilde{F}) < 0$$

where

$$\tilde{A} = \frac{1}{2}\Big(A(0) + A(\pi)\Big) = \begin{bmatrix} A_p & 0 \\ 0 & -\tau I \end{bmatrix}$$

and

$$\tilde{F} = \frac{1}{2}\Big(F(0) + F(\pi)\Big).$$

In other words, there exists a matrix $\tilde{F} \in \mathbb{R}^{2\times 4}$ *that stabilises*

$$(\tilde{A}, B) = \left(\begin{bmatrix} A_p & 0 \\ 0 & -\tau I \end{bmatrix}, \begin{bmatrix} 0 \\ \tau I \end{bmatrix}\right),$$

which is obviously not possible if A_p *is unstable.*

An intuitive explanation of the fact hat the system in Fig. 2.10 is not *quadratically* stabilisable if A_p is unstable, is that quadratic stabilisability allows an infinite rate of parameter variation. In particular, quadratic stability allows that ρ switches between 0 and π, which here means that $R(\rho)$ can switch between I and $-I$. Since the input channels of the system are band-limited, no gain-scheduled state feedback law can cope with the sign of the input channels switching between plus and minus. On the other hand, in Exercise 3.1 a gain-scheduled state-feedback gain is presented that stabilises the system even when A_p is unstable, if a bound on the rate of parameter variation is assumed.

Exercises — Chapter 2

Problem 2.1 *(Multiple Equilibria of Nonlinear Systems)*

Learning Goals

- Appreciate that stability is a property of an equilibrium, of which there may be multiple in a nonlinear system.

References *A resource that treats the analysis of nonlinear equilibria in great detail can be found in [6].*

Task Description Consider the damped pendulum shown in Fig. 2.11, with mass $m = 1$, pendulum length L and damping coefficient b. The system is governed by the following set of differential equations where $x_1 = \alpha$, $x_2 = \dot{\alpha}$:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{g}{L}\sin(x_1) - bx_2 \end{bmatrix} \tag{2.36}$$

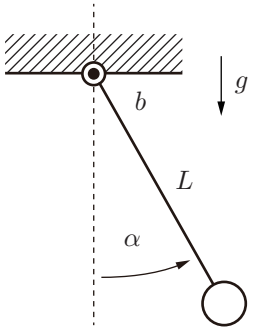


Figure 2.11: Simple damped pendulum

a) For $b = 0$, determine the equilibria of the system.

b) Determine the stability properties of the equilibria.

c) For some positive $b > 0$, determine the equilibria and associated stability properties.

Problem 2.2 *(Prove Stability by Lyapunov Function)*

Learning Goals

- Prove the asymptotic stability of the damped pendulum by means of a Lyapunov function.

References *A resource that treats the direct method of Lyapunov in great detail is [3].*

Task Description Reconsider the damped pendulum shown in Fig. 2.12, with mass $m = 1$, pendulum length L and damping coefficient b. The system is governed by the following set of differential equations where $x_1 = \alpha$, $x_2 = \dot{\alpha}$:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{g}{L}\sin(x_1) - bx_2 \end{bmatrix} \tag{2.37}$$

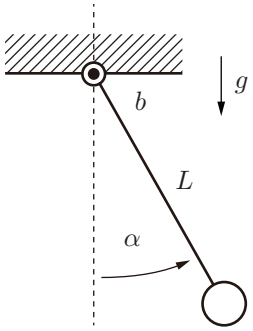


Figure 2.12: Simple damped pendulum

a) Derive the total energy $E(t)$ of the pendulum with respect to the downwards equilibrium position E_0.

b) Prove the (asymptotic) stability of the $(0, 0)$ equilibrium by means of a Lyapunov function using the total energy as a candidate.

Problem 2.3 *(Scheduled Stabilizing State Feedback)*

Learning Goals

- Work out the details of synthesizing a scheduled stabilizing state feedback gain via a finite set of linear matrix inequalities

References *The example is new, the methodology required in this exercise, however, is contained in [1] and has first been proposed in.*

Task Description Consider the parameter varying plant

$$\dot{x} = A(\rho)x + Bu = \begin{bmatrix} -1 & \frac{\rho}{2-\rho^2} \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad y = x. \tag{2.38}$$

Synthesize a scheduled state feedback gain $F(\rho)$, such that $A_{\mathrm{cl}}(\rho, \sigma) = A(\rho) + BF(\rho)$ is stable for all $\rho \in [-1, 1]$ and $\dot{\rho} = \sigma \in [-1, 1]$.

a) Is the plant stabilizable for all admissible values of ρ?

b) Find an affine representation $A(\theta)$ by defining an adequate affine scheduling parameter $\theta(\rho)$. What can be stated about the bounds on both θ and its time derivative $\dot{\theta} = \nu$?

c) Starting with the inequalities

$$P(\theta)A_{\mathrm{cl}}(\theta) + A_{\mathrm{cl}}(\theta)^\top P(\theta) + \sum_{i=1}^{n_\theta} \nu_i \frac{\partial P(\theta)}{\partial \theta_i} < 0, \quad \forall \theta \in P_\theta,\ \forall \nu \in P_\nu, \tag{2.39}$$

$$P(\theta) > 0, \quad \forall \theta \in P_\theta \tag{2.40}$$

and an Ansatz

$$F(\theta)P^{-1}(\theta) = Y(\theta) = Y_0 + \theta Y_1,$$
$$P^{-1}(\theta) = Q(\theta) = Q_0 + \theta Q_1,$$

and the multi-convexity condition to derive convex LMI conditions, in order to find a stabilizing state feedback gain $F(\theta)$.

Chapter 3

Gain-Scheduled Control of LPV Systems

In this chapter we will see how the analysis results on stability and performance of LPV systems presented in Chapter 2 can be be turned into synthesis tools. We will illustrate this in detail for the state feedback case and briefly outline the extension to the output feedback case, referring to the literature for details. One approach to designing a gain-scheduled state feedback law for an LPV system (under simplifying assumptions) was illustrated in Example 2.1, using a linearising change of variables. Here we will introduce an alternative approach, based on the elimination of controller variables from the synthesis conditions.

3.1 The State Feedback Problem

Consider the generalised plant

$$\Sigma_\rho : \quad \left\{ \begin{array}{lll} \dot{x} = & A(\rho)x + B_w(\rho)w & +B_u(\rho)u \\ z = & C_z(\rho)x & +D_{zu}(\rho)u \\ v = & x & \end{array} \right. \tag{3.1}$$

where we assume $\rho \in \mathcal{F}_{\mathcal{P}}^{\mathcal{V}}$ for given compact sets $\mathcal{P}$ and $\mathcal{V}$. This system is shown in Fig. 3.1.

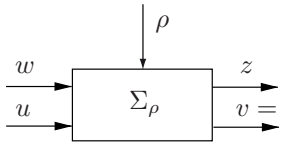


Figure 3.1: Generalised plant for gain-scheduled state feedback

The input-output channel from w to z will be used as performance channel; $u \in \mathbb{R}^m$ is the

control input, and since we consider state feedback the feedback signal v is the full state vector $x \in \mathbb{R}^n$. We will make two assumptions:

- A1: there is no direct feedthrough from w to z, and
- A2: $D_{zu}(\rho)$ has full column rank $\forall \rho \in \mathcal{P}$.

Moreover, in order to facilitate the derivation of the synthesis result presented in this chapter, using assumption A2 we will also assume that a transformation is applied to z such that it can be represented as

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} C_1(\rho) \\ C_2(\rho) \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} u. \tag{3.2}$$

Mixed-Sensitivity S/KS Design

To motivate these simplifying assumptions, we briefly review a mixed-sensitivity S/KS design. A closed-loop system with plant G, controller K and shaping filters W_S and W_K for sensitivity and control sensitivity, respectively, is shown in Fig. 3.2.

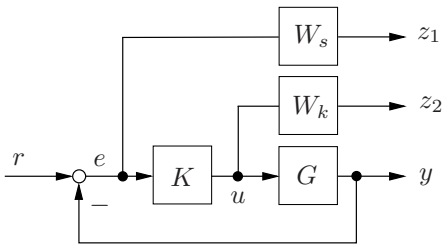


Figure 3.2: Mixed-sensitivity S/KS design

Now assume that (3.1) is used to represent the generalised plant for this design problem. From Fig. 3.2 it is clear that there is no direct feedthrough from u to z_1 if G is strictly proper. As for the path from u to z_2, there is usually a direct feedthrough because W_K is typically chosen to be bi-proper (see Lecture Notes *Optimal and Robust Control*, Chapter 17); in this case the feedthrough matrix is the D-matrix of W_K. This feedthrough matrix will typically have the form βI, where β is a small positive gain. Thus, in order to obtain the representation (3.2) for the performance output in the generalised plant (3.1), only scaling with a scalar factor $1/\beta$ is required.

State Feedback

The problem we now want to solve is this: for system (3.1) and a given set $\mathcal{P} \times \mathcal{V}$, find a continuous function $F : \mathbb{R}^{n_\rho} \to \mathbb{R}^{m \times n}$ such that $\Sigma_{\mathcal{P}}^{\mathcal{V}}$ is stabilised by the state feedback law $u = F(\rho)x$, and

$$\left\| \Sigma_{\mathcal{P}}^{\mathcal{V}} \right\|_{\mathcal{L}_2} < \gamma.$$

This is illustrated in Fig. 3.3. Using the conditions of Theorem 2.5 and assumptions A1 and A2, we formulate the following problem.

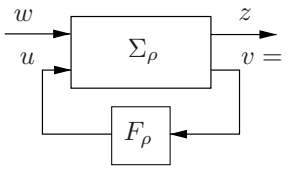


Figure 3.3: Gain scheduled state feedback

Parameter-Dependent State Feedback Problem: Find $F \in C^0(\mathbb{R}^{n_\rho}, \mathbb{R}^{m\times n})$ and $Z \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n\times n})$ such that

$$Z(\rho) > 0 \;\; \forall \rho \in \mathcal{P} \tag{3.3}$$

and

$$\begin{bmatrix} A_F^T(\rho)Z(\rho) + Z(\rho)A_F(\rho) + \sum \nu_i \frac{\partial Z}{\partial \rho_i} & Z(\rho)B_w(\rho) & \frac{1}{\gamma}C_F^T(\rho) \\ * & -I & 0 \\ * & * & -I \end{bmatrix} < 0 \quad \forall (\rho, \nu) \in \mathcal{P} \times \mathcal{V} \tag{3.4}$$

where A_F and C_F represent the closed-loop matrices

$$A_F(\rho) = A(\rho) + B_u(\rho)F(\rho), \qquad C_F(\rho) = C_z(\rho) + D_{zu}F(\rho), \tag{3.5}$$

respectively, Z is a matrix variable representing the Lyapunov matrix, and $(*)$ stands for matrix blocks required to make the left hand side of (3.4) symmetric.

Note that in the partial derivative term in the (1,1) block we let the real vector $\nu \in \mathcal{V} \subset \mathbb{R}^{n_\rho}$ represent the rate vector $\dot{\rho}$. Compared with the condition in Theorem 2.5, a slight change has been introduced concerning the representation of the performance level γ; we leave it as an exercise to verify that (3.4) is indeed equivalent to (2.32).

Since the conditions of Theorem 2.5 are only sufficient and not necessary for stability and $\mathcal{L}_2$-performance, the parameter-dependent state feedback problem, i.e. the problem of finding F and Z that satisfy (3.3) and (3.4) is not equivalent to finding a parameter dependent $F(\rho)$ that guarantees stability and performance on $\mathcal{P} \times \mathcal{V}$: even if (3.3) and (3.4) turn out to be infeasible there may still exist scheduled state feedback gains that solve the problem. The conservatism incurred by this problem formulation is the price for being able to do controller synthesis by solving a convex optimisation problem. In practical applications it turns out that this conservatism is usually at an acceptable level, and that the synthesis tools resulting from this approach provide a simple and efficient way of tuning gain-scheduled controllers for nonlinear plants to a performance level not easily achievable otherwise.

Now turning to the task of finding F and Z that satisfy (3.4), we note that this condition is not linear in the variables because of the product term ZA_F, where A_F depends on F.

One way of dealing with this problem is to introduce a linearising change of variables. For state feedback this was demonstrated in Example 2.1. A linearising change of variables for the output feedback case is also possible (see Lecture Notes *Optimal and Robust Control*, Chapter 19), however more complex than in the state feedback case. Here we will introduce a different technique, referred to as elimination approach, that significantly reduces the complexity of the synthesis problem.

3.2 State Feedback Synthesis - An Elimination Approach

We now present a result, first proposed in [5], that provides a necessary and sufficient condition for the solvability of the parameter-dependent state feedback problem defined in the last section.

Theorem 3.1 *The parameter-dependent state feedback problem is solvable if and only if there exists a* $P \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n\times n})$ *such that* $\forall \rho \in \mathcal{P}$

$$P(\rho) > 0 \quad \text{and} \tag{3.6}$$

$$\begin{bmatrix} P(\rho)\hat{A}(\rho)^T + \hat{A}(\rho)P(\rho) - \sum \pm\bar{\nu}_i \frac{\partial P}{\partial \rho_i} - B_u(\rho)B_u(\rho)^T & P(\rho)C_1(\rho)^T & \frac{1}{\gamma}B_w(\rho) \\ * & -I & 0 \\ * & * & -I \end{bmatrix} < 0 \tag{3.7}$$

where $\hat{A}(\rho) = A(\rho) - B_u(\rho)C_2(\rho)$. *If the problem is solvable, then*

$$F(\rho) = -\Big(B_u^T(\rho)P^{-1}(\rho) + C_2(\rho)\Big) \tag{3.8}$$

In (3.7) we use the notation $\pm\bar{\nu}_i$; since this condition is affine in $\dot{\rho}$ and needs to be checked only for the extreme values, we remove the set $\mathcal{V}$ from the problem formulation. The notation $\pm\bar{\nu}_i$ indicates that for each $\rho \in \mathcal{P}$ one has to check 2^{n_ρ} conditions: one for each combination of the extreme values of $\bar{\nu}_i$, $i = 1, \ldots, n_\rho$.

Before we prove this theorem, we observe that compared with condition (3.4) in the formulation of the parameter-dependent state feedback problem, the state feedback gain matrix $F(\rho)$ has been eliminated from the problem; only $P(\rho)$ is left as the single variable to be solved for. Since the problem is affine in P and also in the rate of parameter variation, (3.6) and (3.7) can be turned via gridding over $\mathcal{P}$ into a finite collection of LMIs. The practical value of this theorem for solving the parameter-dependent state feedback problem is then that instead of having to solve (3.3) and (3.4) for $Z(\rho)$ and $F(\rho)$, we can equivalently solve (3.6) and (3.7) for $P(\rho)$ only, which can be done by solving a collection of LMI problems. If the parameter-dependent state feedback problem is solvable, then

the theorem guarantees that a solution $P(\rho)$ exists, and the state feedback gain can be constructed from (3.8).

Proof: ($\Rightarrow$)

Assume that $F \in C^0(\mathbb{R}^{n_\rho}, \mathbb{R}^{m\times n})$ and $Z \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n\times n})$ exist that satisfy (3.3) and (3.4) on $\mathcal{P}\times\mathcal{V}$. We have to show that this implies the existence of $P(\rho)$ that satisfies (3.6) and (3.7) for all $\rho \in \mathcal{P}$.

Let $M(Z,F)$ denote the right hand side of inequality (3.4) (to simplify notation, from now on we suppress the dependence on ρ). The inequality can be rewritten as

$$M(Z,F) = R(Z) + U(Z)FV^T + VF^TU^T(Z) < 0 \tag{3.9}$$

where

$$M(Z,F) = \left[\begin{array}{cc:c} A_F^TZ + ZA_F + \sum \pm\bar{\nu}_i\frac{\partial Z}{\partial\rho_i} & * & * \\ B_w^TZ & -I & * \\ \hdashline \frac{1}{\gamma}C_F & 0 & -I \end{array}\right],$$

$$R(Z) = \left[\begin{array}{cc:cc} A^TZ + ZA + \sum \pm\bar{\nu}_i\frac{\partial Z}{\partial\rho_i} & * & * & * \\ B_w^TZ & -I & * & * \\ \hdashline \frac{1}{\gamma}C_1 & 0 & -I & * \\ \frac{1}{\gamma}C_2 & 0 & 0 & -I \end{array}\right],$$

$$U(Z) = \left[\begin{array}{c} ZB_u \\ \hdashline 0 \\ \hdashline \frac{1}{\gamma}I \end{array}\right] \qquad \text{and} \qquad V = \left[\begin{array}{c} I \\ \hdashline 0 \\ \hdashline 0 \end{array}\right].$$

(Note that assumptions A1 and A2 as well as (3.2) are used here.) The point of rewriting (3.4) as (3.9) is that the controller variable $F(\rho)$ has been "pulled out" so that it can be eliminated. Introduce

$$P = \frac{1}{\gamma^2}\,Z^{-1}, \tag{3.10}$$

and define bases $U_\perp(P)$ and $V_\perp$ for the null spaces of U and V, respectively, as

$$U_\perp(P) = \begin{bmatrix} \gamma P & 0 & 0 \\ 0 & 0 & I \\ 0 & I & 0 \\ -B_u^T & 0 & 0 \end{bmatrix}, \qquad V_\perp = \begin{bmatrix} 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}.$$

By assumption, $M(Z,F)$ is negative definite over $\mathcal{P}\times\mathcal{V}$. Since both $U_\perp$ and $V_\perp$ have full column rank, we have from (3.9) on $\mathcal{P}\times\mathcal{V}$

$$U_\perp^TMU_\perp < 0 \qquad \text{and} \qquad V_\perp^TMV_\perp < 0,$$

which implies

$$U_\perp^T R U_\perp < 0 \qquad \text{and} \qquad V_\perp^T R V_\perp < 0.$$

Whereas $V_\perp^T R V_\perp < 0$ does not provide any useful information, the condition $U_\perp^T R U_\perp < 0$ is exactly condition (3.7), and that $P(\rho) > 0 \; \forall \rho \in \mathcal{P}$ follows from the assumption that $Z(\rho) > 0 \; \forall \rho \in \mathcal{P}$ and from (3.10).

($\Leftarrow$)

Now assume that there exists $P(\rho)$ that satisfies (3.6) and (3.7) on $\mathcal{P}$. We have to show that this implies the existence of $Z(\rho)$ and $F(\rho)$ that satisfy (3.3) and (3.4) on $\mathcal{P} \times \mathcal{V}$, and that $F(\rho)$ is given by (3.8).

Consider (3.7), here repeated as

$$\left[\begin{array}{c:cc} P\hat{A}^T + \hat{A}P - \Psi_\pm^P - B_u B_u^T & PC_1^T & \frac{1}{\gamma} B_w \\ \hdashline * & -I & 0 \\ * & * & -I \end{array}\right] < 0,$$

where we for simplicity we introduce the notation $\Psi_\pm^P = \sum \pm \bar{\nu}_i \frac{\partial P}{\partial \rho_i}$. Using the Schur complement, we see that (3.7) holds if and only if

$$P\hat{A}^T + \hat{A}P - \Psi_\pm^P - B_u B_u^T + \begin{bmatrix} PC_1^T & \gamma^{-1} B_w \end{bmatrix} \begin{bmatrix} C_1 P \\ \gamma^{-1} B_w^T \end{bmatrix} < 0.$$

Expanding the last term on the left hand side, defining

$$Z = \gamma^{-2} P^{-1} \tag{3.11}$$

and multiplying from left and right by Z, we obtain

$$\hat{A}^T Z + Z\hat{A} + \Psi_\pm^Z - \gamma^2 Z B_u B_u^T Z + \gamma^{-2} C_1^T C_1 + Z B_w B_w^T Z < 0$$

where we introduce the notation $\Psi_\pm^Z = \sum \pm \bar{\nu}_i \frac{\partial Z}{\partial \rho_i}$. This is equivalent to (after adding and subtracting C_2 and some further manipulation)

$$\begin{aligned} &\left[A - B_u(C_2 + \gamma^2 B_u^T Z)\right]^T Z + Z\left[A - B_u(C_2 + \gamma^2 B_u^T Z)\right] + \Psi_\pm^Z \\ &\quad + \gamma^{-2} \begin{bmatrix} C_1^T & C_2^T - (C_2^T + \gamma^2 Z B_u) \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 - (C_2 + \gamma^2 B_u^T Z) \end{bmatrix} + Z B_w B_w^T Z \; < \; 0. \end{aligned} \tag{3.12}$$

Now introducing a new variable

$$F = -\left(B_u^T P^{-1} + C_2\right)$$

and recalling the definitions

$$A_F = A + B_u F, \qquad C_F = C_z + D_{zu} F = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} F$$

where (3.2) is used, we can rewrite (3.12) as

$$A_F^T Z + Z A_F + \Psi_{\pm}^Z + \frac{1}{\gamma^2} C_F^T C_F + Z B_w B_w^T Z < 0.$$

Again employing the Schur complement, we see that this inequality is equivalent to (3.4). Thus we have shown the existence of Z and F that satisfy (3.4) on $\mathcal{P} \times \mathcal{V}$. That $Z(\rho) > 0 \; \forall \rho \in \mathcal{P}$ follows from the assumption that $P(\rho) > 0 \; \forall \rho \in \mathcal{P}$ and from (3.11).

Constructing a Gain-Scheduled State Feedback Controller

If the state feedback problem is solvable, one way of constructing a gain-scheduled state feedback controller is to assume affine dependence of $P(\rho)$ on ρ, i.e.

$$P(\rho) = P_0 + \rho_1 P_1 + \ldots + \rho_{n_\rho} P_{n_\rho}.$$

Conditions (3.6) and (3.7) can then be solved by solving a finite collection of LMIs in $P_0, P_1, \ldots, P_{n_\rho}$ on a grid over $\mathcal{P}$. The state feedback gain $F(\rho)$ is then obtained from (3.8). An application that illustrates this approach on gain-scheduled state feedback for a control moment gyroscope with experimental results, is presented in [7].

3.3 Output Feedback Synthesis

In the previous section the state feedback synthesis problem and its solution via an elimination approach was presented in detail. In this section we will give a brief outline of the extension to output feedback synthesis, and refer to the literature, in particular to [5], for more details.

We consider the generalised plant

$$\Sigma_\rho : \quad \left\{ \begin{array}{lllll} \dot{x} = & A(\rho)x & + \; B_w(\rho)w & + \; B_u(\rho)u \\ z = & C_z(\rho)x & & + \; D_{zu}(\rho)u \\ v = & C_v(\rho)x & + \; D_{vw}(\rho)w & \end{array} \right. \tag{3.13}$$

where the feedback signal v is a linear combination of state variables and external inputs. We make the following assumptions.

- A1: $D_{zw} = 0$ and $D_{vu} = 0$
- A2: $D_{zu}(\rho)$ has full column rank $\; \forall \rho \in \mathcal{P}$
- A3: $D_{vw}(\rho)$ has full row rank $\; \forall \rho \in \mathcal{P}$.

Based on assumptions A2 and A3, we assume that

$$B_w = [B_1 \;\; B_2], \qquad C_z = \left[\begin{array}{c} C_1 \\ C_2 \end{array} \right], \qquad D_{vw} = [0 \;\; I], \qquad D_{zu} = \left[\begin{array}{c} 0 \\ I \end{array} \right]$$

so that the generalised plant (3.13) becomes

$$\Sigma_\rho : \left\{ \begin{array}{rcl} \dot{x} &=& Ax + [B_1 \;\; B_2] \left[\begin{array}{c} w_1 \\ w_2 \end{array} \right] + B_u u \\ \left[\begin{array}{c} z_1 \\ z_2 \end{array} \right] &=& \left[\begin{array}{c} C_1 \\ C_2 \end{array} \right] x + \left[\begin{array}{c} 0 \\ I \end{array} \right] u \\ v &=& C_v x + [0 \;\; I] \left[\begin{array}{c} w_1 \\ w_2 \end{array} \right] \end{array} \right. \tag{3.14}$$

It was pointed out in Section 3.1 that the assumption on D_{zu}, that results in partitioning of z, is easily satisfied e.g. in mixed-sensitivity S/KS design problems. Similarly, the dual assumption on D_{vw}, can be interpreted in terms of partitioning the external input w into a reference and a disturbance signal, or into process and measurement noise.

The controller that closes the feedback loop as shown in Fig. 3.4 is

$$K_\rho : \left\{ \begin{array}{rcl} \dot{\zeta} = & A_K(\rho,\dot{\rho})\zeta & + \; B_K(\rho,\dot{\rho})v \\ u = & C_K(\rho,\dot{\rho})\zeta & + \; D_K(\rho,\dot{\rho})v \end{array} \right. . \tag{3.15}$$

Note that in contrast to the parameter-dependent state feedback control law derived in the previous section, this controller depends not only on ρ but also on $\dot{\rho}$. This is a particular feature of the solution to the parameter-dependent output feedback control problem formulated below, that makes it difficult to implement - it requires the on-line availability not only of the scheduling signals but also of their derivatives. In practice one often tries to avoid control laws depending on $\dot{\rho}$, usually at the expense of achievable performance.

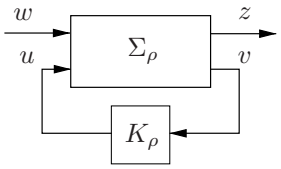


Figure 3.4: Gain scheduled output feedback

Defining

$$x_c = \left[\begin{array}{c} x \\ \zeta \end{array} \right], \quad z = \left[\begin{array}{c} z_1 \\ z_2 \end{array} \right], \quad w = \left[\begin{array}{c} w_1 \\ w_2 \end{array} \right]$$

we obtain the closed-loop system

$$\begin{array}{rcl} \dot{x}_c &=& A_c(\rho,\dot{\rho})x_c + B_c(\rho,\dot{\rho})w \\ z &=& C_c(\rho,\dot{\rho})x_c. \end{array} \tag{3.16}$$

where

$$A_c(\rho,\dot{\rho}) = \left[\begin{array}{cc} A + B_u D_K C_v & B_u C_K \\ B_K C_v & A_K \end{array} \right], \qquad B_c(\rho,\dot{\rho}) = \left[\begin{array}{cc} B_1 & B_2 + B_u D_K \\ 0 & B_K \end{array} \right],$$

$$C_c(\rho,\dot\rho) = \begin{bmatrix} C_1 & 0 \\ C_2 + D_K C_v & C_K \end{bmatrix}. \tag{3.17}$$

Because of the assumed partitioning of the external input and output channels in (3.14), the closed-loop matrices B_c and C_c are 2×2 block matrices.

We are now ready to formulate the synthesis problem.

Parameter-Dependent Output Feedback Problem: Find continuous, matrix-valued functions (A_K, B_K, C_K, D_K) and $Z \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n\times n})$ such that

$$Z(\rho) > 0 \;\; \forall \rho \in \mathcal{P},$$

and

$$\begin{bmatrix} A_c^T(\rho,\dot\rho)Z(\rho) + Z(\rho)A_c(\rho,\dot\rho) + \sum \nu_i \frac{\partial Z}{\partial \rho_i} & Z(\rho)B_c(\rho,\dot\rho) & \frac{1}{\gamma}C_c^T(\rho,\dot\rho) \\ * & -I & 0 \\ * & * & -I \end{bmatrix} < 0$$

$$\forall (\rho,\nu) \in \mathcal{P}\times\mathcal{V}.$$

where A_c, B_c and C_c are defined in (3.17). The product terms ZA_c and ZB_c that involve products between the Lyapunov matrix Z and the controller matrices, prevent turning this condition directly into a collection of LMIs via gridding. One way of dealing with this is a linearising change of variables as in the LTI case; here we present a simpler solution based on an extension of the elimination approach that was used for the state feedback problem in the previous section.

The following result is due to [5].

Theorem 3.2 *The parameter-dependent output feedback problem is solvable if and only if there exist $X, Y \in C^1(\mathbb{R}^{n_\rho}, \mathcal{S}^{n\times n})$ such that $\forall \rho \in \mathcal{P}$ the following conditions are satisfied:*

$$X(\rho) > 0, \qquad Y(\rho) > 0 \tag{3.18}$$

and

$$\begin{bmatrix} X\hat A^T + \hat A X - \sum \left(\pm \bar\nu_i \frac{\partial X}{\partial \rho_i}\right) - B_u B_u^T & XC_1^T & \frac{1}{\gamma}B_w \\ * & -I & 0 \\ * & * & -I \end{bmatrix} < 0, \tag{3.19}$$

$$\begin{bmatrix} \tilde A^T Y + Y\tilde A - \sum \left(\pm \bar\nu_i \frac{\partial Y}{\partial \rho_i}\right) - C_v^T C_v & YB_1 & \frac{1}{\gamma}C_z^T \\ * & -I & 0 \\ * & * & -I \end{bmatrix} < 0, \tag{3.20}$$

$$\begin{bmatrix} X & \frac{1}{\gamma}I \\ \frac{1}{\gamma}I & Y \end{bmatrix} \geq 0 \tag{3.21}$$

where $\hat A = A - B_u C_2$ and $\tilde A = A - B_2 C_v$.

Comparing this theorem with the state feedback result of the last section, we see that the conditions involving X have the same form as the state feedback conditions. Here we have in addition two dual conditions in Y, and a coupling condition. The conditions of this theorem can be turned via gridding into a finite collection of LMIs; the two conditions involving $\pm\bar{\nu}_i$ have to be solved at each grid value of ρ for the 2^{n_ρ} extreme combinations of $\bar{\nu}_i$.

Constructing a Gain-Scheduled Output Feedback Controller

If the output feedback problem is solvable, an output feedback controller (3.15) can be constructed as follows. Assume affine dependence of $X(\rho)$ and $Y(\rho)$ on ρ, i.e.

$$X(\rho) = X_0 + \rho_1 X_1 + \ldots + \rho_{n_\rho} X_{n_\rho}, \qquad Y(\rho) = Y_0 + \rho_1 Y_1 + \ldots + \rho_{n_\rho} Y_{n_\rho}.$$

Solve (3.18), (3.19), (3.20) and (3.21) on a grid over $\mathcal{P}$ for X_i and Y_i, $i = 1, 2, \ldots, n_\rho$. Define

$$F(\rho) = -\Big(B_u^T(\rho)X^{-1}(\rho) + C_2(\rho)\Big)$$
$$L(\rho) = -\Big(Y^{-1}(\rho)C_v(\rho) + B_2(\rho)\Big)$$

and

$$H(\rho,\nu) = \\ -\left(X^{-1}A_F + A_F^T X^{-1} + \sum_{i=1}^{n_\rho}\left(\nu_i \frac{\partial X^{-1}}{\partial \rho_{i.k}}\right) + C_F^T C_F + \gamma^{-2} X^{-1} B_w B_w^T X^{-1}\right),$$

where

$$A_F(\rho) = A(\rho) + B_u(\rho)F(\rho), \qquad C_F(\rho) = C_z(\rho) + D_{zu}F(\rho).$$

Note that H depends not only on ρ but also on ν, which represents $\dot{\rho}$. Furthermore, define

$$Q(\rho) = Y(\rho) - \gamma^{-2}X^{-1}(\rho) > 0$$

and

$$M(\rho,\nu) = H(\rho,\nu) - \gamma^2\Big(YLD_{vw} + QB_w\Big)B_w^T X^{-1}$$

Then the GS output feedback problem is solved by

$$\begin{aligned} A_K(\rho,\nu) &= A + B_u F + Q^{-1}YLC_v - \gamma^{-2}Q^{-1}M(\rho,\nu) \\ B_K(\rho) &= -Q^{-1}(\rho)Y(\rho)L(\rho) \\ C_K(\rho) &= F(\rho) \\ D_K(\rho) &= 0. \end{aligned}$$

Exercises — Chapter 3

Problem 3.1 *(Parameter-Dependent Lyapunov Function Based Analysis)*

Learning Goals

- Implement gridding based analysis conditions via LMIs and parameter-dependent Lyapunov functions

References *The original example is contained in the PhD thesis of Fen Wu, [5].*

Task Description Consider the nonlinear plant

$$\dot{x} = A(\rho)x + Bu, \quad y = x. \tag{3.22}$$

with

$$A(\rho) = \begin{bmatrix} A_{11} & R(\rho) \\ 0_{2\times 2} & -\tau I_{2\times 2} \end{bmatrix}, \quad R(\rho) = \begin{bmatrix} \cos(\rho) & \sin(\rho) \\ -\sin(\rho) & \cos(\rho) \end{bmatrix},$$
$$A_{11} = \begin{bmatrix} 0.75 & 2.00 \\ 0.00 & 0.50 \end{bmatrix}, \quad B = \begin{bmatrix} 0_{2\times 2} \\ \tau I_{2\times 2} \end{bmatrix}, \quad \tau = 3.75,$$

and a state feedback control law

$$u = F(\rho)x.$$

Consider the parameter range and range of the rate of change

$$\rho \in P_\rho = [-\pi, \pi], \quad \dot{\rho} = \sigma \in P_\sigma = [-1, 1].$$

a) Suppose a continuous state-feedback gain $F(\rho)$ exists. Is the system quadratically stable, i.e. does there exist a symmetric matrix $P = P^T > 0$, such that

$$\left(A(\rho) + BF(\rho)\right)^T P + P\left(A(\rho) + BF(\rho)\right) < 0, \quad \forall \rho \in P_\rho \, ?$$

c) Assume the state-feedback gain

$$F(\rho) = -\left[R^\top(\rho)\left(\gamma I_{2\times 2} + A_{11}\right) \quad 0_{2\times 2}\right], \quad \gamma = 0.5.$$

has been derived.

Analyze the closed-loop system's stability in the parameter range

$$\rho \in P_\rho = [-\pi, \pi], \quad \dot{\rho} = \sigma \in P_\sigma = [-1, 1]$$

by means of solving the linear matrix inequalities

$$P(\rho)A_{\mathrm{cl}}(\rho) + A_{\mathrm{cl}}(\rho)^\top P(\rho) + \sigma\frac{\partial P(\rho)}{\partial \rho} < 0, \quad \forall \rho \in P_\rho,\ \forall \sigma \in P_\sigma, \tag{3.23}$$

$$P(\rho) = P_0 + \cos(\rho)P_1 + \sin(\rho)P_2 > 0, \quad \forall \rho \in P_\rho \tag{3.24}$$

for P_0, P_1 and P_2 on a grid.

Check the feasibility with the resulting solutions for P_0, P_1 and P_2 on a much denser grid.

Problem 3.2 *(State Feedback Synthesis with Parameter-Dependent Lyapunov Functions)*

Learning Goals

- Synthesize an LPV state feedback controller for a control moment gyroscope with Wu's method using Matlab Tools
- Investigate the use of parameter-dependent Lyapunov functions
- Implement the LPV controller in Simulink

References *The example is adapted from [7].*

Task Description A model of a control moment gyroscope is described by the nonlinear differential equation

$$M(q)\,\ddot{q} + k(q,\dot{q}) = f(\dot{q}) + \begin{bmatrix} I \\ 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

where q_i, $i = \{1,\ldots,4\}$ are angles and T_i, $i = \{1,2\}$ are torques.

Linearization around a moving operating point $\rho(t) = \begin{bmatrix} \dot{q}_1, q_2, q_3 \end{bmatrix}$ yields a linear parameter-varying plant model

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

$\rho\colon \mathbb{R} \mapsto \mathcal{P}, \dot{\rho}\colon \mathbb{R} \mapsto \mathcal{V}$

The model has

- controlled outputs $y = \begin{bmatrix} q_3 & q_4 \end{bmatrix}^\mathsf{T}$
- control inputs $u = \begin{bmatrix} T_1 & T_2 \end{bmatrix}^\mathsf{T}$
- state vector $x = \begin{bmatrix} q_3 & q_4 & \dot{q}_2 & \dot{q}_3 & \dot{q}_4 \end{bmatrix}^\mathsf{T}$

The model matrices can be obtained by using the provided Matlab function `linearize_gyro(q1dot, q2, q3)`.

The Matlab script `exercise_design_state_feedback.m` contains all necessary commands to design an LPV controller using Matlab's Robust Control Toolbox and the LPVTools.

a) Design an integral augmented LPV state feedback controller for the Gyroscope using Matlab. As weights, you can use

$$W_S = \begin{bmatrix} \frac{1}{s} & 0 \\ 0 & \frac{1}{s} \end{bmatrix}$$
$$W_K = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Compare the results to an LTI state feedback controller.

b) Experiment with different choices of basis functions for the Lyapunov matrix. The LPVTools' `basis(F,NAME1,PARTIAL1,NAME2,PARTIAL2,...)` command can be used to specify a basis function `F` and its partial derivative `PARTIAL` with respect to the variable specified by `NAME`. Does the performance index depend on the choice of basis functions? Does the computational effort depend on the choice of basis functions?

c) Implement the controller in Simulink by reconstructing the state feedback gain in every time step from 1) the model matrices and Lyapunov matrix according to Wu's formula; 2) a lookup table representation. Run Simulations. Are there noticable differences?

Chapter 4

Gain-Scheduled Control of LFT and Polytopic Systems

In the last chapter a rather general result was presented that can be used to synthesise gain-scheduled controllers for nonlinear plants modelled as LPV systems, with no restrictions on the type of functional dependence of the model on the scheduling signals. The synthesis problem is based on a quadratic Lyapunov function which is allowed to depend on the scheduling parameters in order to reduce conservatism. The price to be paid for the generality of this approach is its complexity. This includes complexity of the synthesis problem, which involves gridding the admissible parameter range and solving a collection of LMIs on the grid points. As for controller implementation, it requires - at least in the output feedback case - that not only the scheduling signals but also their derivatives are available online.

There are thus strong incentives to look for simplifications when LPV gain scheduling is considered in practical applications. Simplifications are possible when the LPV model of the plant displays special types of functional dependence on the scheduling signals, in particular when the dependence is *affine* or *rational*. LPV models with affine parameter dependence can be represented as *polytopic systems* or as *LFT systems*, and LPV models with rational dependence can be represented as LFT systems. Both model structures will be introduced in this chapter, and synthesis techniques applicable to each will be presented.

4.1 LFT Representations

LFT stands for *linear fractional transformation*; many control problems can be represented in the form of LFTs. Before we discuss the representation of LPV systems in LFT form, we first provide a definition and an illustrative example.

Definition 4.1 *Let M be a complex matrix partitioned as*

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \in \mathbb{C}^{(l_1+l_2)\times(m_1+m_2)},$$

and let $\Delta \in \mathbb{C}^{l_1 \times m_1}$ be another complex matrix. Then the upper LFT with respect to Δ is defined as

$$\mathcal{F}_u(P, \Delta) = P_{22} + P_{21}\Delta(I - P_{11}\Delta)^{-1}P_{12}. \tag{4.1}$$

This definition is illustrated by the block diagram in Fig. 4.1, where P and Δ are assumed to be complex matrices with dimensions given in the definition. Then Δ can be seen as mapping $z \to w$, whereas (4.1) represents the mapping $u \to v$.

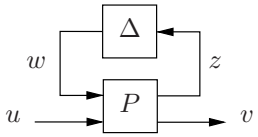


Figure 4.1: Linear fractional transformation

We will often assume that the matrix Δ is unknown, but confined to a compact set $\mathbf{\Delta} \subset \mathbb{C}^{l_1 \times m_1}$.

Definition 4.2 *The mapping in (4.1) is said to be well-posed if*

$$\det(I - P_{11}\Delta) \neq 0 \quad \forall \quad \Delta \in \mathbf{\Delta}$$

A *lower LFT* can be defined similarly, where Δ is connected to v and z in a lower loop, resulting in a new map $w \to z$

$$\mathcal{F}_l(P, \Delta) = P_{11} + P_{12}\Delta(I - P_{22}\Delta)^{-1}P_{21},$$

where well-posedness of the mapping is defined accordingly. Representations as shown in Fig. 4.1 are frequently used in the context of robust control, where P is assumed to be an LTI system with input channels partitioned into w and v, and output channels partitioned into z and v. A typical usage of this structure is to let P represent a generalised plant, and Δ a matrix containing uncertain plant parameters; see e.g. Lecture Notes *Optimal and Robust Control*, Chapter 21. When constructing such a representation of an uncertain system, care must be taken to ensure the well-posedness of the model.

Here we will present an example that illustrates the technique of "pulling out parameters", i.e. isolating uncertain or time-varying parameters of a physical plant model and expressing them as uncertainty block Δ in an LFT representation. This technique is also used for the design of gain-scheduled controllers; the parameters to be pulled out are then time-varying scheduling parameters.

Example 4.1 *Consider the coupled-mass spring system shown in Fig. 4.2. Masses m_1 and m_2 are connected by a spring with stiffness k. The position of mass m_2 is to be controlled by applying force u at mass m_1. Available for feedback is only the measured position of m_2. Since sensor and actuator are non-collocated, this becomes a challenging control problem when the model parameters m_1, m_2 and k are uncertain or time-varying.*

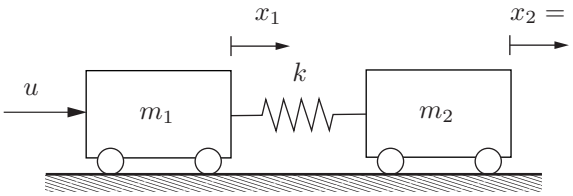


Figure 4.2: Coupled-mass spring system

The equations of motion are

$$m_1\ddot{x}_1 = u - f \qquad \text{and} \qquad m_2\ddot{x}_2 = f$$

where f is the spring force

$$f = k(x_1 - x_2).$$

The measured output is

$$y = x_2.$$

A state space model of this system is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k}{m_1} & \frac{k}{m_1} & 0 & 0 \\ \frac{k}{m_2} & -\frac{k}{m_2} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix}$$

$$y = [0 \;\; 1 \;\; 0 \;\; 0] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Now assume that the model parameters m_1, m_2 and k are uncertain or time-varying and are to be pulled out of the model. For example, assume that we only know about m_1 that $0.5 \le m_1 \le 1.5$. Define a nominal mass $m_{10} = 1$, and express m_1 as

$$m_1 = m_{10} + \alpha_1\delta_1,$$

where $\alpha_1 = 0.5$ and δ_1 is an unknown parameter satisfying $|\delta_1| < 1$.

In this manner, the other uncertain model parameters can be represented as

$$\begin{aligned} m_2 &= m_{20} + \alpha_2\,\delta_2 \\ k &= k_0 + \alpha_3\,\delta_3. \end{aligned}$$

where $|\delta_i| < 1,\ i = 1, 2, 3,$ *and* α_i *represents the admissible parameter range. Substituting these expressions in the equations of motion yields for example for* m_1

$$(m_{10} + \alpha_1\,\delta_1)\ddot{x}_1 = m_{10}\ddot{x}_1 + \alpha_1\delta_1\ddot{x}_1 = u - f.$$

Defining

$$w_1 = \delta_1 z_1 \qquad \textit{and} \qquad z_1 = \ddot{x}_1$$

we can rewrite this as

$$m_{10}\ddot{x}_1 + \alpha_1 w_1 = u - f. \tag{4.2}$$

The equations for m_2 *and* k *can be rewritten in the same manner as*

$$m_{20}\ddot{x}_2 + \alpha_2 w_2 = f \tag{4.3}$$

and

$$f = k_0(x_1 - x_2) + \alpha_3 w_3, \tag{4.4}$$

where

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} \delta_1 & & \\ & \delta_2 & \\ & & \delta_3 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} \delta_1 & & \\ & \delta_2 & \\ & & \delta_3 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ x_1 - x_2 \end{bmatrix}. \tag{4.5}$$

Using $\dot{x}_1 = x_3$, $\dot{x}_2 = x_4$ *and (4.2) - (4.5), we can construct a state space model for the fixed, LTI part of the uncertain system, with state equation*

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_0}{m_{10}} & \frac{k_0}{m_{10}} & 0 & 0 \\ \frac{k_0}{m_{20}} & -\frac{k_0}{m_{20}} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\frac{\alpha_1}{m_{10}} & 0 & -\frac{\alpha_3}{m_{10}} \\ 0 & -\frac{\alpha_2}{m_{20}} & \frac{\alpha_3}{m_{20}} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_{10}} \\ 0 \end{bmatrix} u,$$

output equation

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} -\frac{k_0}{m_{10}} & \frac{k_0}{m_{10}} & 0 & 0 \\ \frac{k_0}{m_{20}} & -\frac{k_0}{m_{20}} & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} +$$

$$\begin{bmatrix} -\frac{\alpha_1}{m_{10}} & 0 & -\frac{\alpha_3}{m_{10}} \\ 0 & -\frac{\alpha_2}{m_{20}} & \frac{\alpha_3}{m_{20}} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{m_{10}} \\ 0 \\ 0 \end{bmatrix} u.$$

and $w = \Delta z$*, where* $\Delta = \mathrm{diag}(\delta_1, \delta_2, \delta_3)$*. In a more compact form, this reads*

$$\begin{aligned} \dot{x} &= A\,x + B_w w + B_u u \\ z &= C_z x + D_{zw} w + D_{zu} u \\ v &= C_v x, \end{aligned}$$

A block diagram of this system is shown in Fig. 4.3.

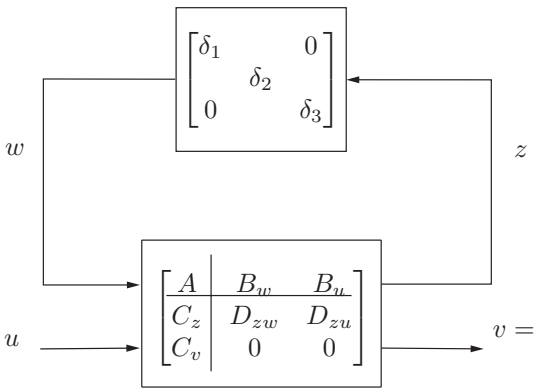


Figure 4.3: LFT representation of uncertain coupled-mass spring system

4.2 Gain-Scheduled Control of LFT Systems

The approach presented in the previous section can be applied to pull out uncertain parameters, so that the small gain theorem can be used for robust controller design. This approach can also be used to model parameters that are *known but time-varying*. Assume e.g. that a plant to be controlled has model parameters $\rho_1(t)$, $\rho_2(t)$, $\ldots$, $\rho_{n_\rho}(t)$ which vary over time and whose values can be measured online. Assume further that we want to design a gain-scheduled controller for this plant with the ρ_is as scheduling signals, as shown in Fig. 4.4.

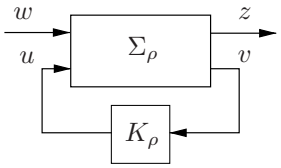


Figure 4.4: Gain scheduled control

In the same way as in Example 4.1 we can pull out the time-varying parameters by using the representation

$$\rho_i(t) = \rho_{0i} + \alpha_i \theta_i(t) \tag{4.6}$$

and arranging the new parameters θ_i in a diagonal matrix

$$\Theta(t) = \begin{bmatrix} \theta_1(t) I_{r_1} & & \\ & \ddots & \\ & & \theta_{n_\rho}(t) I_{r_{n_\rho}} \end{bmatrix}. \tag{4.7}$$

If the plant model depends rationally on $\rho_i,\ i = 1, \ldots, n_\rho$, then it is possible to construct an LFT representation $\mathcal{F}_u(P, \Theta)$ with Θ as in (4.7). Even though in Example 4.1 the parameters appear only once in the diagonal block Δ, in general it may be required to have parameter θ_i repeated r_i times along the diagonal, see Exercise 4.3.

Utilising the LFT representation $\mathcal{F}_u(P, \Theta)$ of the parameter-dependent plant Σ_ρ in Fig. 4.4, a gain-scheduled controller K_ρ can now be designed that has itself a (lower) LFT form $\mathcal{F}_l(K, \Theta)$, as shown in Fig. 4.5. Since the scheduling signals $\rho_i(t)$ and therefore from (4.6) the scheduling parameters $\theta_i(t)$ are assumed to be available online, they can be used to schedule the controller. A feature of the closed-loop structure shown in Fig. 4.5 is that it allows to design an LTI controller $K(s)$ for an LTI plant $P(s)$, while taking the time-varying scheduling blocks into account.

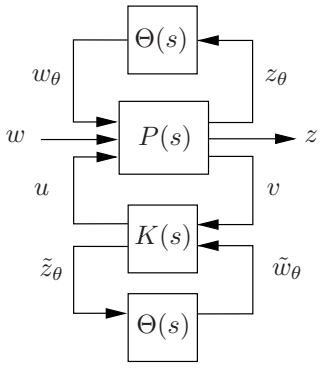


Figure 4.5: Gain-scheduled control of LFT system

Note that the external channel from w to z in Fig. 4.4 has been split in Fig. 4.5 into an LFT channel from w_θ to z_θ, and a performance channel from w to z. The scheduling

channel of the controller has been labelled $\tilde{z}_\theta$ and $\tilde{w}_\theta$, to mirror the scheduling channel of the plant. A state space realisation of the LTI part of the plant is (note that the plant now has three input/output channels)

$$P: \quad \left\{ \begin{array}{rcl} \dot{x} &=& A\,x + B_\theta\,w_\theta + B_w\,w + B_u\,u \\ z_\theta &=& C_\theta\,x + D_{\theta\theta}\,w_\theta + D_{\theta w}\,w + D_{\theta u}\,u \\ z &=& C_z\,x + D_{z\theta}\,w_\theta + D_{zw}\,w + D_{zu}\,u \\ v &=& C_v\,x + D_{v\theta}\,w_\theta + D_{vw}\,w \end{array} \right. \tag{4.8}$$

with

$$w_\theta = \Theta\, z_\theta. \tag{4.9}$$

When constructing the LFT representation, we assume that the LTI part of the plant model has been scaled such that

$$\Theta^T(t)\Theta(t) \leq I \quad \forall t \geq 0. \tag{4.10}$$

The LTI part of the controller (which has now two input/output channels) can be represented as

$$K: \quad \left\{ \begin{array}{rcl} \dot{\zeta} &=& A_K\,\zeta + B_{Kv}\,v + B_{K\theta}\,\tilde{w}_\theta \\ u &=& C_{Ku}\,\zeta + D_{Kuv}\,v + D_{Ku\theta}\,\tilde{w}_\theta \\ \tilde{z}_\theta &=& C_{K\theta}\,\zeta + D_{K\theta v}\,v + D_{K\theta}\,\tilde{w}_\theta \end{array} \right. \tag{4.11}$$

with

$$\tilde{w}_\theta = \Theta\, \tilde{z}_\theta. \tag{4.12}$$

Equivalent Robust Control Problem

One approach to designing a gain-scheduled feedback controller K_ρ was proposed in [8]; it is based on transforming the problem of synthesising a scheduled controller for a parameter-dependent plant into the problem of synthesising an LTI controller for an uncertain plant. Consider Fig. 4.6. It represents exactly the same feedback configuration as Fig. 4.5, displayed in a rearranged form (the Θ-block of the controller has been "flipped up").

The point of redrawing the feedback configuration in this form is that one can now define a new, augmented generalised plant $P_a(s)$ as shown in Fig. 4.7. This augmented generalised plant has the LTI controller $K(s)$ and an uncertainty block

$$\bar{\Theta} = \left[\begin{array}{cc} \Theta & 0 \\ 0 & \Theta \end{array} \right] \tag{4.13}$$

with norm less than one attached to it, in the same form as e.g. the representation of an uncertain plant in *Optimal and Robust Control*, Chapter 21 (compare Fig. 21.6). If we want to find a controller that guarantees robust stability of the system in Fig. 4.7 for all

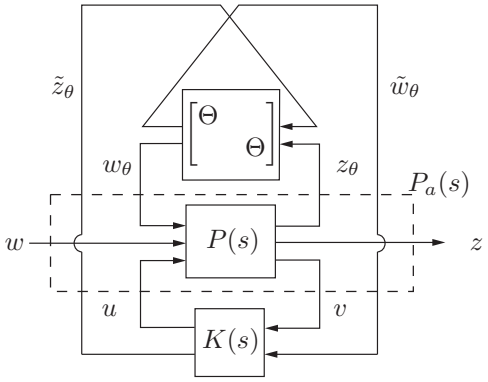


Figure 4.6: Fig. 4.5 rearranged

admissible $\bar{\Theta}$, one can thus employ the small gain theorem, and design an LTI controller $K(s)$ that achieves a H_∞-norm across the uncertainty channel

$$\begin{bmatrix} w_\theta \\ \tilde{w}_\theta \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} z_\theta \\ \tilde{z}_\theta \end{bmatrix} \tag{4.14}$$

of less than one. This problem can be formulated as that of finding K such that

$$\left\| \mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \bar{\Theta}\Big) \right\|_\infty < 1.$$

Now consider the implementation of $K(s)$. From the interconnection structure of $P_a(s)$ it is clear that $\tilde{u} = \tilde{z}_\theta$ and $\tilde{w} = \tilde{z}_\theta$. Thus, the LTI controller obtained by solving the robust control problem in Fig. 4.7 is the same as the gain-scheduled controller of Fig. 4.6 and Fig. 4.5 (since these figures all show exactly the same feedback configuration).

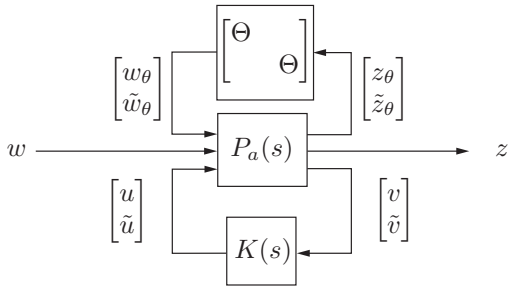


Figure 4.7: Equivalent robust control problem with augmented generalised plant

Structured Uncertainty and Conservatism

So a straightforward way of obtaining a robustly stabilising gain-scheduled controller for a parameter-dependent plant Σ_ρ that can be represented in LFT form (4.8) and (4.9), is to find a controller for the generalised LTI plant $P_a(s)$ that achieves a H_∞ norm less than one across the uncertainty channel, if such a controller exists. That is a standard H_∞

synthesis problem. The resulting controller can then be implemented as the gain-scheduled controller (4.11) and (4.12). This approach is however highly conservative: it is based on the small gain theorem, and the small gain theorem guarantees stability for any real (and in fact complex) uncertainty matrix, whereas the uncertainty block in Fig. 4.7 has the repeated block diagonal structure of (4.13). Clearly, the set of all controllers that robustly stabilise the configuration in Fig. 4.7 for all $\|\bar{\Theta}\| \leq 1$ is larger than the set of all controllers that robustly stabilise the same configuration when $\bar{\Theta}$ is replaced by an arbitrary matrix Δ where $\|\Delta\| \leq 1$. In fact the latter set of controllers is a strict subset of the former, because the highly structured matrix $\bar{\Theta}$ is a special case of an arbitrary matrix Δ. Thus, when we search for a controller that robustly stabilises $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \bar{\Theta}\Big)$ for all $\|\bar{\Theta}\| \leq 1$, and we do it by using the small gain problem, then we will be searching only over the small subset of controllers that solve the more difficult problem of robustly stabilising $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \Delta\Big)$ for all $\|\Delta\| \leq 1$.

In order to reduce this conservatism, define a set of symmetric, positive definite matrices

$$\mathcal{L} = \Big\{ L > 0 \,\Big|\, L\bar{\Theta} = \bar{\Theta}L \Big\}. \tag{4.15}$$

Then one can modify the block diagram in Fig. 4.7 as shown in Fig. 4.8 by introducing scaling blocks L and L^{-1} in the upper feedback loop. If $L \in \mathcal{L}$, then Fig. 4.8 is equivalent to Fig. 4.7 because the scaling blocks commute with the uncertainty block.

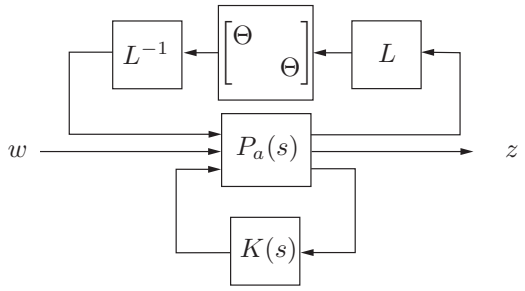


Figure 4.8: Equivalent robust control problem with scaling

The introduction of these scaling blocks reduces the conservatism of the synthesis approach outlined above, because the set of all controllers that robustly stabilise the closed-loop system $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), L^{-1}\bar{\Theta}L\Big)$ for all $\|\bar{\Theta}\| \leq 1$ and all $L \in \mathcal{L}$ contains the set of all controllers that robustly stabilise $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \bar{\Theta}\Big)$ for all $\|\bar{\Theta}\| \leq 1$ as a subset (i.e. the special case $L = I$). The scaling blocks can be absorbed into the generalised plant, as indicated in Fig. 4.9, so that L can be turned into a decision variable of the synthesis problem.

Robust Performance

So far only robust stability of the closed-loop system $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \bar{\Theta}\Big)$ for all $\|\bar{\Theta}\| \leq 1$ was considered. Performance can be included in the problem formulation by augmenting

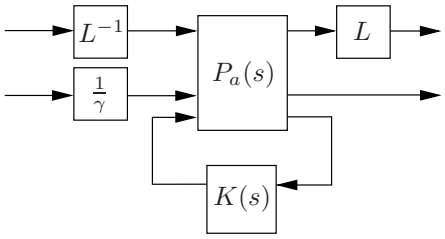


Figure 4.9: Generalised plant with scaling and performance measure

the uncertainty block (4.13) with a performance block and closing the upper feedback loop in Fig. 4.7 not only between z_θ, $\tilde{z}_\theta$ and w_θ, $\tilde{w}_\theta$, but also between w and z according to

$$\begin{bmatrix} z_\theta \\ \tilde{z}_\theta \\ z \end{bmatrix} = \begin{bmatrix} \Theta(t) & 0 & 0 \\ 0 & \Theta(t) & 0 \\ 0 & 0 & \Delta(t) \end{bmatrix} \begin{bmatrix} w_\theta \\ \tilde{w}_\theta \\ w \end{bmatrix}, \tag{4.16}$$

where $\Delta(t)$ is a fictitious, time-varying uncertainty block. Robust stability of the system in Fig. 4.9 with an upper feedback loop closed by (4.16), for $\Theta^T(t)\Theta(t) < I$ and $\|\Delta(t)\| \leq \gamma \ \forall t \geq 0$, is then guaranteed by the small gain theorem if the H_∞ norm across the uncertainty channel (4.14) is less than 1. We can now pose the following problem.

Robust Controller Synthesis Problem: Given $P_a(s)$ and $\gamma > 0$, find an LTI controller $K(s)$ and a scaling matrix $L \in \mathcal{L}$ such that the H_∞ norm across the uncertainty channel (4.14) is less than one.

It is shown in [8] that robust stability of $\mathcal{F}_u\Big(\mathcal{F}_l(P_a, K), \bar{\Theta}\Big)$ for all Θ and Δ such that $\Theta^T(t)\Theta(t) < I$ and $\Delta(t)\| \leq \gamma \ \forall t \geq 0$, implies that the gain-scheduled controller $\mathcal{F}_l(K, \Theta)$ quadratically stabilises the LPV system $\mathcal{F}_u(P, \Theta)$, and achieves quadratic $\mathcal{L}_2$ performance γ for all admissible trajectories of $\theta(t)$. Thus, from an LTI controller that solves the robust controller synthesis problem defined above, we can construct a gain-scheduled controller (4.11) that achieves quadratic stability and performance for the LFT-LPV system (4.8). Moreover, it is also shown in [8] how the problem of constructing an LTI controller that solves the robust controller synthesis problem can be obtained by solving an LMI problem; this is achieved via an elimination approach.

Since the method proposed in [8] was published, considerable research effort has been devoted to reducing the conservatism incurred by this approach. In [9] the use of parameter-dependent Lyapunov functions was proposed. A method based on the *full-block S-procedure* was presented in [10], which uses full block rather than structured multipliers, and in [11] an extension of this approach using parameter-dependent Lyapunov functions was proposed.

4.3 Gain-Scheduled Control of Polytopic Systems

The second "simplified" model structure for LPV systems, that allows to avoid gridding the parameter space, is that of *polytopic systems*. Consider the LPV system

$$\begin{aligned} \dot{x} &= A(\theta)x + B(\theta)w \\ z &= C(\theta)x + D(\theta)w. \end{aligned} \tag{4.17}$$

We assume that $\theta(t) \in \mathcal{P}\ \forall t \geq 0$ for a given compact set $\mathcal{P} \subset \mathbb{R}^{n_\theta}$. Such a system is called *polytopic* if it satisfies two conditions:

1. The set $\mathcal{P}$ is a *polytope*, i.e. it can be expressed as a convex hull

$$\mathcal{P} = \mathrm{Co}\{\theta_{v1}, \theta_{v2}, \ldots, \theta_{vs}\} \tag{4.18}$$

where the $\theta_{vi} \in \mathbb{R}^{n_\theta}$ are the *vertices* of the polytope and s is the number of vertices. The representation (4.18) implies that

$$\mathcal{P} = \{\theta \in \mathbb{R}^{n_\theta} |\ \theta = \sum_{i=1}^{s} \alpha_i \theta_{vi},\ \sum_{i=1}^{s} \alpha_i = 1,\ \alpha_i \geq 0, i = 1, \ldots, s\}.$$

The coefficients α_i are referred to as *convex coordinates*. A typical occurrence of a polytopic set of parameters is a hyperrectangle, which arises when the individual parameters θ_i are known to be confined to intervals $\underline{\theta}_i \leq \theta_i \leq \bar{\theta}_i$, $i = 1, \ldots, s$. Other shapes are possible; Fig. 4.10 shows two examples.

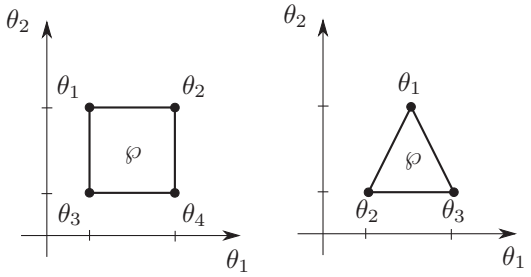


Figure 4.10: Polytopic admissible parameter ranges

2. The model (4.17) depends *affinely* on the parameter vector θ. In this case, the set of admissible LTI systems that is generated when θ in (4.17) ranges over the polytope $\mathcal{P}$ is itself a polytope, i.e. it can be represented by

$$\begin{bmatrix} A(\theta) & B(\theta) \\ C(\theta) & D(\theta) \end{bmatrix} = \mathrm{Co}\left\{ \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix},\ i = 1, \ldots, s \right\} \quad \forall \theta \in \mathcal{P},$$

where

$$(A_i, B_i, C_i, D_i) = \Big(A(\theta_{v_i}), B(\theta_{v_i}), C(\theta_{vi})C(\theta_{v_i})\Big),\ i = 1, \ldots, s \tag{4.19}$$

are the *vertex plants*. Assume for example that $n_\theta = 1$ and $\theta_1 \le \theta \le \theta_2$, then affine dependence on ρ implies that

$$A\Big(\alpha\theta_1 + (1-\alpha)\theta_2\Big) = \alpha A_1 + (1-\alpha)A_2, \quad 0 \le \alpha \le 1.$$

More generally, if

$$\theta(t) = \sum_{i=1}^{s} \alpha_i \theta_{v_i},$$

then we have

$$\begin{bmatrix} A(\theta) & B(\theta) \\ C(\theta) & D(\theta) \end{bmatrix} = \sum_{i=1}^{s} \alpha_i \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix}, \tag{4.20}$$

where the α_i are convex coordinates.

Quadratic $\mathcal{L}_2$ Performance of Polytopic Systems

Recall from Section 2.5 that an LPV system

$$\begin{aligned} \dot{x} &= A(\theta)x + B(\theta)w \\ z &= C(\theta)x + D(\theta)w \end{aligned} \tag{4.21}$$

is quadratically stable and has quadratic $\mathcal{L}_2$ performance γ on a compact parameter set $\mathcal{P}$ if there exists a $P > 0$ such that

$$\begin{bmatrix} A^T(\theta)P + PA(\theta) & PB(\theta) & C^T(\theta) \\ B^T(\theta)P & -\gamma I & D^T(\theta) \\ C(\theta) & D(\theta) & -\gamma I \end{bmatrix} < 0 \tag{4.22}$$

for all $\theta \in \mathcal{P}$.

If (4.17) is a polytopic LPV system, i.e. if $A(\cdot), B(\cdot), C(\cdot), D(\cdot)$ depend affinely on θ and

$$\mathcal{P} = \text{Co}\,\{\theta_{v_1}, \theta_{v_2}, \ldots, \theta_{v_s}\},$$

then substituting from (4.20) in (4.22) yields the following result.

Theorem 4.1 *System (4.17) is quadratically stable and has quadratic performance γ, if there exists a $P > 0$ such that*

$$\begin{bmatrix} A_i^T P + PA_i & PB_i & C_i^T \\ B_i^T P & -\gamma I & D_i^T \\ C_i & D_i & -\gamma I \end{bmatrix} < 0$$

for $i = 1, 2, \ldots, s$, where (A_i, B_i, C_i, D_i) are the vertex plants as in (4.19).

In this case gridding is not required; it suffices to check the condition in the vertices of the polytope $\mathcal{P}$.

$\mathcal{L}_2$-Optimal GS Control of Polytopic Systems

This last result can be turned into a synthesis procedure for the design of gain-scheduled controllers for polytopic LPV systems. Consider the generalised plant

$$\Sigma_\theta : \quad \left\{ \begin{array}{rcl} \dot{x} = & A(\theta)x & + B_w(\theta)w + B_u u \\ z = & C_z(\theta)x & + D_{zw}(\theta)w + D_{zu}u \\ v = & C_v x & + D_{vw}w \end{array} \right. \tag{4.23}$$

and assume that $\theta \in \mathcal{P}\ \forall t \geq 0$, that $\mathcal{P}$ is a polytope with vertices $\{\theta_{v_1}, \ldots, \theta_{v_s}\}$, and that A, B_w, C_z and D_{zw} are affine in θ. The synthesis approach we will present next requires the following assumptions.

- A1: $B_u,\ C_v,\ D_{zu}$ and D_{vw} are constant
- A2: $\Big(A(\theta), B_u\Big)$ is quadratically stabilizable on $\mathcal{P}$
- A3: $\Big(A(\theta), C_v\Big)$ is quadratically detectable on $\mathcal{P}$

Whereas it is clear that assumptions A2 and A3 are necessary for the existence of a solution to the output feedback synthesis problem, assumption A1 looks like a serious restriction that may not be satisfied in many applications. It is however possible by pre- and post-filtering the plant with low-pass filters of sufficiently large bandwidth, to satisfy this assumption, see [12].

For the generalised LPV plant (4.23) we want to design a gain-scheduled controller

$$K_\theta : \quad \left\{ \begin{array}{rcl} \dot{\zeta} = & A_K(\theta)\zeta & + B_K(\theta)v \\ u = & C_K(\theta)\zeta & + D_K(\theta)v \end{array} \right. \tag{4.24}$$

that is affine in θ. When this controller is used to close the feedback loop from v to u in (4.23), we obtain the closed-loop system

$$\begin{array}{rcl} \dot{x}_c & = & A_c(\theta)x_c + B_c(\theta)w \\ z & = & C_c(\theta)x_c + D_c(\theta)w. \end{array} \tag{4.25}$$

The closed-loop matrices in (4.25) are given by

$$\left[\begin{array}{cc} A_c(\theta) & B_c(\theta) \\ C_c(\theta) & D_c(\theta) \end{array} \right] = \left[\begin{array}{cc} \mathcal{A}(\theta) + \mathcal{B}_2\Omega(\theta)\mathcal{C}_2 & \mathcal{B}_1(\theta) + \mathcal{B}_2\Omega(\theta)\mathcal{D}_{21} \\ \mathcal{C}_1(\theta) + \mathcal{D}_{12}\Omega(\theta)\mathcal{C}_2 & \mathcal{D}_{11}(\theta) + \mathcal{D}_{12}\Omega(\theta)\mathcal{D}_{21} \end{array} \right], \tag{4.26}$$

where the controller matrices are collected in

$$\Omega(\theta) = \left[\begin{array}{cc} A_K(\theta) & B_K(\theta) \\ C_K(\theta) & D_K(\theta) \end{array} \right]$$

and augmented model matrices are defined as

$$\left[\begin{array}{ccc} \mathcal{A}(\theta) & \mathcal{B}_1(\theta) & \mathcal{B}_2 \\ \mathcal{C}_1(\theta) & \mathcal{D}_{11}(\theta) & \mathcal{D}_{12} \\ \mathcal{C}_2 & \mathcal{D}_{21} & 0 \end{array}\right] = \left[\begin{array}{cc:cc:c} A(\theta) & 0 & B_w(\theta) & 0 & B_u \\ 0 & 0 & 0 & I & 0 \\ \hdashline C_z(\theta) & 0 & D_{zw}(\theta) & 0 & D_{zu} \\ \hdashline 0 & I & 0 & 0 & 0 \\ C_v & 0 & D_{vw} & 0 & 0 \end{array}\right].$$

From (4.26) it is clear that the closed-loop system (4.25) is affine in the controller Ω, and also in the parameter vector θ. Here it becomes clear why assumption A1 is necessary: without it, affinity in θ of the closed-loop system would be lost.

Now we are in a position to formulate the synthesis problem.

Quadratic $\mathcal{L}_2$-gain Performance Problem: Find $P > 0$ and vertex-controllers

$$\Omega_i = \left[\begin{array}{cc} A_{Ki} & B_{Ki} \\ C_{Ki} & D_{Ki} \end{array}\right]$$

such that for $i = 1, 2, \ldots, s$

$$\left[\begin{array}{ccc} A_c^T(\theta_{v_i})P + PA_c(\theta_{v_i}) & PB_c(\theta_{v_i}) & C_c^T(\theta_{v_i}) \\ B_c^T(\theta_{v_i})P & -\gamma I & D_c^T(\theta_{v_i}) \\ C_c(\theta_{v_i}) & D_c(\theta_{v_i}) & -\gamma I \end{array}\right] < 0 \tag{4.27}$$

From these vertex controllers, if they exist, on can construct a polytopic, gain-scheduled controller (4.24), that guarantees quadratic stability and $\mathcal{L}_2$ performance γ over $\mathcal{P}$ for the polytopic plant (4.23). Note however that even though the closed-loop matrices (A_c, B_c, C_c, D_c) are affine in the controller variables, (4.27) is not an LMI in the controller variables and P because of the product terms PA_c and PB_c. The following result, proposed in [12], can be used to solve the quadratic $\mathcal{L}_2$-gain performance problem via an elimination approach.

Theorem 4.2 *Define $\mathcal{N}_X$ and $\mathcal{N}_Y$ as bases for the null spaces*

$$\mathcal{R}(\mathcal{N}_X) = \mathcal{N}\left([C_v \;\; D_{vw} \;\; 0]\right)$$

and

$$\mathcal{R}(\mathcal{N}_Y) = \mathcal{N}\left(\left[B_u^T \;\; D_{zu}^T \;\; 0\right]\right),$$

respectively. The quadratic $\mathcal{L}_2$-gain performance problem is solvable if and only if there exist $X = X^T$ and $Y = Y^T$ such that for $i = 1, 2, \ldots, s$

$$\mathcal{N}_X^T \left[\begin{array}{ccc} A_i^T X + XA_i & XB_{wi} & C_{zi}^T \\ B_{wi}^T X & -\gamma I & D_{zwi}^T \\ C_{zi} & D_{zwi} & -\gamma I \end{array}\right] \mathcal{N}_X < 0 \tag{4.28}$$

$$\mathcal{N}_Y^T \begin{bmatrix} A_iY + YA_i^T & YC_{zi}^T & B_{wi} \\ C_{zi}Y & -\gamma I & D_{zwi} \\ B_{wi}^T & D_{zwi}^T & -\gamma I \end{bmatrix} \mathcal{N}_Y < 0 \tag{4.29}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} > 0 \tag{4.30}$$

The controller variables have been eliminated from (4.28) and (4.29). The above conditions can be solved as LMIs in the vertices for the variables X and Y. From these variables one can construct the closed-loop Lyapunov matrix P. Substituting P in (4.27) turns (4.27) into an LMI that can be solved for the controller variables. For more details see [12].

Exercises — Chapter 4

Problem 4.1 *(LFTs - Pulling out the Deltas (Simple Example))*

Learning Goals

- Learn how to derive linear fractional representations.

References *The core methodology and many other useful lemmas and formulas can be found in [13].*

Task Description The mass-spring-damper system shown in Fig. 4.11 has an uncertain mass $m(t) = m_0 + \delta_1$, damping coefficient b and a variable stiffness coefficient $k(t) = k_0 + \delta_2$. The system is governed by the following set of differential equations:

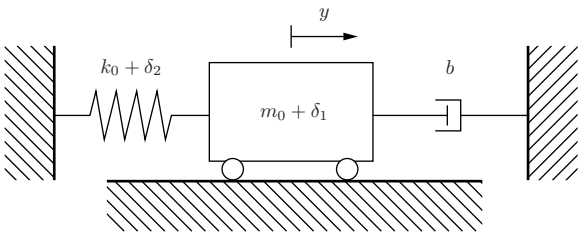


Figure 4.11: Simple mass-spring-damper system with uncertain mass

$$\dot{x} = \begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k(t)/m(t) & -b/m(t) \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} = Ax. \tag{4.31}$$

a) Derive an LFT representation of the state-space model with only the mass being uncertain and $k(t) = k_0$.

b) Derive an LFT representation of the state-space model with all uncertainties.

Problem 4.2 *(LFT of system with repeated parameters)*

Learning Goals

- Derive Linear Fractional Transformations of a system with repeated parameters.

References *The original example is taken from [1].*

Task Description Consider again the system given in example 1.1, and a corresponding quasi-LPV representation:

$$\begin{aligned} \dot{x}(t) &= -x(t) + u(t) \\ y(t) &= (1-\rho^2)x = C(\rho)x \end{aligned} \tag{4.32}$$

a) Derive an LFT representation of the output equation, such that $y = \mathcal{F}_u(M, \Theta)$.

b) Construct a generalized plant of the system using the robust control approach as in Figure 4.6.

Problem 4.3 *(LFTs - Pulling out the Deltas)*

Learning Goals

- Learn how to construct and manipulate linear fractional representations manually and programmatically.

References *The model is taken from actual research conducted at the Institute of Control Systems, which has been published, e.g. in [14].*

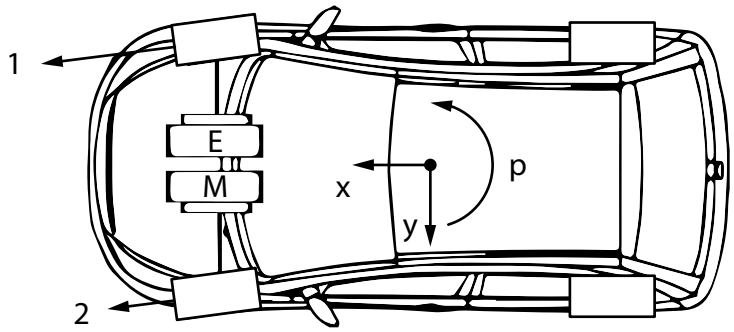


Figure 4.12: A simple single-track vehicle model

Task Description The following nonlinear plant represents a simple single-track vehicle model as shown in Fig. 4.12, with v_x, v_y as the longitudinal and transversal direction, respectively, as well as $\dot{\psi}$ as the yaw rate.

$$\begin{array}{rcccccccc} M\dot{v}_x & = M\dot{\psi}v_y & & & & + & u_1 & + & u_2 \\ M\dot{v}_y & = & -M\dot{\psi}v_x & -c_1\frac{v_y}{v_x} & +c_2\frac{\dot{\psi}}{v_x} & & & +c_4 w & \\ I\ddot{\psi} & = & & c_1\frac{v_y}{v_x} & -c_3\frac{\dot{\psi}}{v_x} & -b_1u_1 & +b_1u_2 & +c_5 w & \end{array} \tag{4.33}$$

The disturbance input w denotes the steering angle, whereas controlled inputs u_1 and u_2 are the left and right wheel forces, respectively. The plant outputs are $v_y = \begin{bmatrix} v_x & \dot{\psi} \end{bmatrix}^\top$. Constants M and I denote mass and rotational inertia. The physical meaning of the constants $c_1, c_2, ..., c_5$ and b_1 is omitted for brevity.

Table 4.1: Values of the physical parameters.

	M	I	c_1	c_2	c_3	c_4	c_5	b_1
Value	1772	1800	154,000	14,000	234,430	70,000	86,800	0.72

a) Manually derive a linear fractional state-space representation of the model. Use $\delta_1 = v_x$ and $\delta_2 = \dot{\psi}$ as LFT parameters, as well as $x = \begin{bmatrix} v_x & v_y & \dot{\psi} \end{bmatrix}^\top$ as the state vector. Draw a block diagram with the "Deltas pulled out".

b) Use MATLAB to build a linear fractional representation and compare the size of the resulting Δ-blocks from the manually and the MATLAB derived models.

Problem 4.4 *(Torque Vectoring by LPV Control)*

Learning Goals

- Learn how to synthesize, tune and implement LPV controllers.

References *The polytopic LPV synthesis method is described in [12], whereas the LFT LPV synthesis method is described in [15], [16].*

Task Description Reconsider the vehicle model from Problem 4.3.

a) Derive an affine LPV parameterization of the nonlinear plant and synthesize an affine LPV controller.

b) Synthesize an LFT LPV controller for the torque vectoring problem based on the LFT model and using the LFT LPV synthesis method described in [15], [16]. *The MATLAB files implementing the synthesis method are provided.*

Problem 4.5 *(Parameter Set Mapping vs. LFT (Simple Example))*

Learning Goals

- Learn how to find a tighter convex hull or to reduce a set of affine parameters.

References *The core methodology can be found in [17].*

Task Description The crane-trolley system shown in Fig. 4.13 has the trolley mass m_T, trolley position x, damping coefficient b and a variable rope length L. The trolley is actuated by a force u_T, whereas the rope length can be directly controlled by $u_L = \dot{L}$. The load mass is denoted m_L and its oscillation is damped by a force parameterized by the coefficient d. For small angles $-10° < \varphi < 10°$, the system is governed by the following

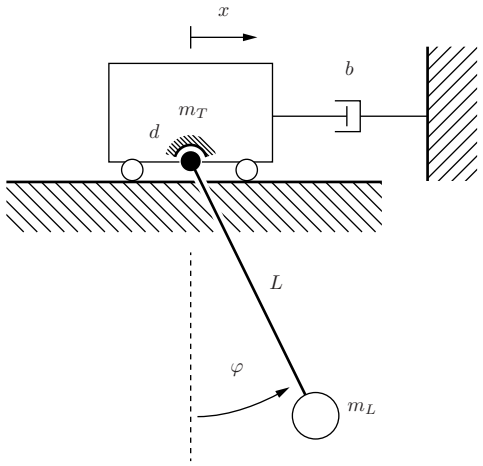


Figure 4.13: Simple crane-trolley system with variable rope length

set of differential equations:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\varphi} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{m_T} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{b}{m_T L} & -\frac{d}{m_L L^2} & -\frac{g}{L} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{m_T} & 0 \\ 0 & 0 \\ -\frac{1}{m_T L} & -2\frac{\dot{\varphi}}{L} \end{bmatrix} \begin{bmatrix} u_T \\ u_L \end{bmatrix} \tag{4.34}$$

a) Use an affine LPV parameterization with $\theta_1 = \frac{1}{L}$, $\theta_2 = \frac{1}{L^2}$ and $\theta_3 = \frac{\dot{\varphi}}{L}$ to derive an affine LPV model.

b) Investigate the relation between $\theta_1 = \frac{1}{L}$ and $\theta_2 = \frac{1}{L^2}$. Therefore assume the limits $1 < L < 10$ and compare the physically admissible parameter range with the box shaped convex hull spanned by the maximum and minimum parameter values.

c) Generate a data set by gridding the parameter range in terms of L, resulting in values $L_i, i = 1, \ldots, N$. Stack the resulting data samples in a fat matrix

$$\Theta = \begin{bmatrix} \theta_1(L_1) & \theta_1(L_2) & \ldots & \theta_1(L_N) \\ \theta_2(L_1) & \theta_2(L_2) & \ldots & \theta_2(L_N) \end{bmatrix} \in \mathbb{R}^{2\times N}.$$

Then perform a singular value decomposition on the data matrix Θ

$$\Theta = U\Sigma V^*$$

and recall the meaning of the matrices U and V.

d) Consider the data set $\Phi = \Sigma V^*$. What is the relation to Θ and how can a new parameter set ϕ_1, ϕ_2 be derived from the set θ_1, θ_2 that more tightly represents the physically admissible trajectories?

e) Compare the singular values found in Σ. Divide them into significant and less significant ones and use the same approach on U to project the parameters θ_1, θ_2 onto a reduced parameter set with only a single parameter ϕ.

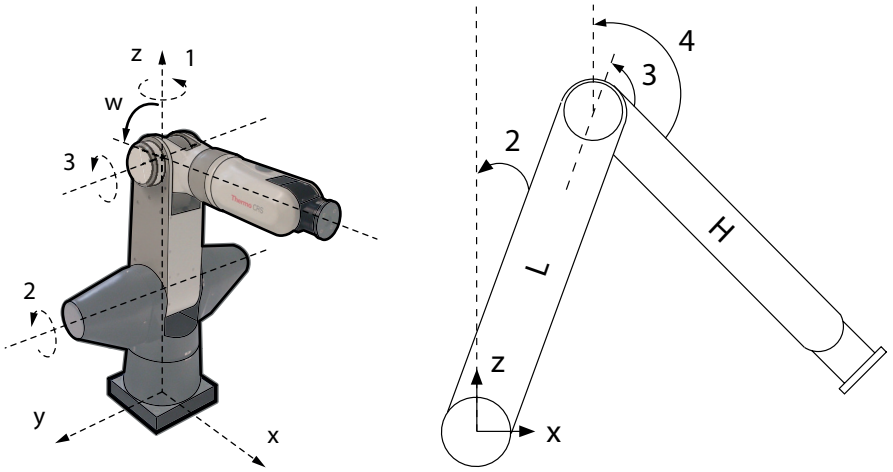
Problem 4.6 *(Parameter Set Mapping and Control — 2-DOF Robotic Manipulator)*

Learning Goals

- Learn how to make polytopic LPV controller synthesis tractable for plants with a large number of affine scheduling parameters.

References *An extensive description of both plant and methodology can be found in [18] using a slightly different approach and nomenclature.*

Task Description Two degrees-of-freedom of a robotic manipulator are to be controlled. The model has been transformed, such that $\tilde{q}_3 = q_2 + q_3$. Consider the following state-space



(a) Isometric view of the robot (b) Side view of the 2-DOF robot model

Figure 4.14: The CRS Thermo A465 robotic manipulator

model of the 2-DOF robotic manipulator.

$$\begin{bmatrix} \ddot{q}_2 \\ \ddot{\tilde{q}}_3 \\ \dot{q}_2 \\ \dot{\tilde{q}}_3 \end{bmatrix} = \begin{bmatrix} A_{11}(\theta) & A_{12}(\theta) & A_{13}(\theta) & A_{14}(\theta) \\ A_{21}(\theta) & A_{22}(\theta) & A_{23}(\theta) & A_{24}(\theta) \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_2 \\ \dot{\tilde{q}}_3 \\ q_2 \\ \tilde{q}_3 \end{bmatrix} + \begin{bmatrix} B_{11}(\theta) & B_{12}(\theta) \\ B_{21}(\theta) & B_{22}(\theta) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \end{bmatrix}, \tag{4.35}$$

$$\begin{bmatrix} q_2 \\ \tilde{q}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_2 \\ \dot{\tilde{q}}_3 \\ q_2 \\ \tilde{q}_3 \end{bmatrix} \tag{4.36}$$

with

$$
\begin{aligned}
A_{11}(\theta) &= -\theta_2(b_3b_7 - b_3b_2) - \theta_3b_3b_9 - \theta_1(b_6b_7 + b_2b_9) + \theta_4b_3^2,\\
A_{12}(\theta) &= \theta_1b_2b_9 + \theta_3b_3b_9 + \theta_6b_7b_3,\\
A_{13}(\theta) &= \theta_5b_5b_7,\\
A_{14}(\theta) &= \theta_7(b_7b_4 - b_2b_4) - \theta_8b_3b_4,\\
A_{21}(\theta) &= -\theta_2(b_3b_1 + b_3b_8) + \theta_3(b_3b_9 + b_3b_6) + \theta_1(b_1b_9 - b_6b_8),\\
A_{22}(\theta) &= -\theta_1b_1b_9 - \theta_3b_3b_9 + \theta_6b_3b_8 - \theta_{10}b_3^2,\\
A_{23}(\theta) &= \theta_5b_5b_8 - \theta_9b_5b_3,\\
A_{24}(\theta) &= \theta_7(b_8b_4 + b_1b_4),
\end{aligned}
$$

$$
\begin{aligned}
B_{11}(\theta) &= \theta_1b_7, & B_{12}(\theta) &= -\theta_1b_2 - \theta_3b_3,\\
B_{21}(\theta) &= \theta_1b_8 - \theta_3b_3, & B_{22}(\theta) &= -\theta_1b_1 + \theta_3b_3
\end{aligned}
$$

and

$$
\begin{aligned}
\delta_1 &= \cos(-q_2 + q_3), & \delta_4 &= \dot{q}_3,\\
\delta_2 &= \sin(-q_2 + q_3), & \delta_5 &= \operatorname{sinc}(q_2),\\
\delta_3 &= \dot{q}_2, & \delta_6 &= \operatorname{sinc}(q_3),
\end{aligned}
$$

$$
\begin{aligned}
v &= b_7b_1 + b_7\delta_1b_3 - b_2\delta_1b_3 + b_2b_8 - \delta_1^2b_3^2 + \delta_1b_3b_8,\\
\theta_1 &= 1/v, & \theta_6 &= \delta_4\delta_2/v,\\
\theta_2 &= \delta_3\delta_2/v, & \theta_7 &= \delta_6/v,\\
\theta_3 &= \delta_1/v, & \theta_8 &= \delta_6\delta_1/v,\\
\theta_4 &= \delta_3\delta_1\delta_2/v, & \theta_9 &= \delta_5\delta_1/v,\\
\theta_5 &= \delta_5/v, & \theta_{10} &= \delta_4\delta_1\delta_2/v.
\end{aligned}
$$

The parameters have been identified based on experimental data as described in [18]. The resulting values are

Table 4.2: Values of the identified condensed parameters.

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9
Value	0.0715	0.0058	0.0114	0.3264	0.3957	0.6253	0.0749	0.0705	1.1261

a) By the technique of parameter set mapping, [17], find a parameter set that approximates $\theta_1, \ldots, \theta_{10}$ by two parameters ϕ_1, ϕ_2. Therefore, generate data by gridding the parameter range in terms of the actual degrees-of-freedom and their time derivatives. Alternatively, use an experimental trajectory provided with the MATLAB files.

b) Synthesize a polytopic LPV gain-scheduling controller based on the approximate model.

Problem 4.7 *(LPV Control of a 2-DOF Robotic Manipulator)*

Learning Goals

- Learn how to synthesize, tune and implement LPV controllers.

References *An extensive description of both plant and methodology can be found in [18]. The polytopic LPV synthesis method is described in [12], whereas the LFT LPV synthesis method is described in [15], [16].*

Task Description Two degrees-of-freedom of a robotic manipulator are to be controlled. Use the model given in Problem 4.6, [18].

a) Synthesize an affine LPV controller for the reduced parameter set-based model. What issue arises when you are trying to synthesize a controller for the plant with full scheduling order?

b) Synthesize an affine LPV controller for the robot based on the model with full scheduling order using the LFT LPV synthesis method described in [15], [16]. *T*he MATLAB files implementing the synthesis method are provided.

Part II

COOPERATIVE CONTROL OF MULTIAGENT SYSTEMS

Chapter 5

Consensus Protocol and Formation Control

In this chapter we will introduce the concept of a *consensus protocol*, and we will see how consensus protocols can be used as a basis for the design of *formation control* schemes for a network of dynamic agents. A consensus protocol can be viewed as a local control law, that each agent in a group is equipped with, in combination with a structure of communication between individual agents. The formation control problem addresses the design of local controllers for mobile agents that are required to move in space while maintaining specified relative positions with respect to each other (a formation); this is to be achieved by each agent using only local information (i.e. each agent uses its own measurements and data received from its "neighbours").

5.1 Consensus Protocols, Illustrative Example

We consider a group of dynamic agents who are exchanging information through a given communication network. To describe such a group and its communication, we need a dynamic model of each agent, and we need a representation of the *communication topology* (who is exchanging information with whom?). In this section we will illustrate these concepts with a simple example.

Consider a group of three agents (agent 1, agent 2 and agent 3), with the communication topology shown in Fig. 5.1. The communication topology is represented by a *communication graph*: agent 2 is sending information to agents 1 and 3, and receives information from those agents, whereas there is no exchange of information between agents 1 and 3.

In this example we assume that each communication link is bidirectional, i.e. whenever there is a communication link between two agents then information is exchanged in both directions; this is indicated by the arrow heads in Fig. 5.1. A communication graph that has

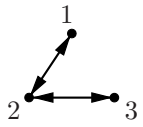


Figure 5.1: Communication Graph

only bidirectional links between agents is called *undirected*, otherwise it is called *directed*. In this section we will consider multi-agent systems with undirected communication graphs; multi agent systems with directed communication graphs will be briefly discussed in Section 5.2.

Note that the graph shown in Fig. 5.1 does not tell us anything about the *spatial location* of agents (e.g. it does not imply that agent 3 is located to the right of agent 2). It provides only information about the existence or non-existence of communication links between agents. We nevertheless assume that the agents we consider do move in space; of practical interest are groups of agents moving in 2-dimensional or 3-dimensional space. For simplicity we will first restrict our discussion to a group of agents that move only in one spatial dimension. Fig. 5.2 shows possible spatial locations for each agent, the position of agent i is denoted by x_i.

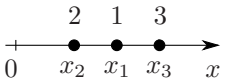


Figure 5.2: Spatial location of agents

Here we assume that we want the agents to move such that they converge towards a common location x_{cons} (this is referred to as the agents *reaching consensus* on a common position), and we want this consensus position to be the arithmetic mean of their initial individual positions, i.e. our goal is

$$\lim_{t\to\infty} x_1(t) = \lim_{t\to\infty} x_2(t) = \lim_{t\to\infty} x_3(t) = x_{\text{cons}} = \frac{1}{3}(x_{10} + x_{20} + x_{30}), \tag{5.1}$$

assuming that the agents are located at initial positions x_{10}, x_{20} and x_{30}, respectively, at time 0. If this is achieved, we will say that the agents reach *average consensus*.

So far we have discussed the communication topology; we have yet to specify dynamic models of the agents. In the following, we assume that all agents have identical dynamic properties; here we assume that

$$\dot{x}_i(t) = u_i(t), \qquad x_i(0) = x_{i0}, \qquad i = 1, 2, 3. \tag{5.2}$$

Thus each agent is modelled as a single integrator and can be seen as a point on which a velocity u_i is imposed. In order for the agents to reach average consensus, each agent is

equipped with a local control law. Let $\mathcal{J}_i$ denote the *set of neighbours* of agent i, i.e. the set of agents that exchange information with agent i. In our example we have

$$\mathcal{J}_1 = \{2\}, \quad \mathcal{J}_2 = \{1, 3\}, \quad \mathcal{J}_3 = \{2\}.$$

We now assume that each agent i is equipped with a local control law

$$u_i(t) = \sum_{j \in \mathcal{J}_i} \Big(x_j(t) - x_i(t) \Big). \tag{5.3}$$

This is called a *consensus protocol*; in addition to local feedback (using an agent's internal measurements) it requires information from neighbouring agents. Substituting (5.3) into (5.2) yields

$$\begin{aligned} \dot{x}_1 &= x_2 - x_1 \\ \dot{x}_2 &= x_1 - x_2 + x_3 - x_2 \\ &= x_1 - 2x_2 + x_3 \\ \dot{x}_3 &= x_2 - x_3. \end{aligned}$$

This can be represented in a more compact form as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \qquad x(0) = \begin{bmatrix} x_{10} \\ x_{20} \\ x_{30} \end{bmatrix}. \tag{5.4}$$

This vector differential equation governs the motion of the three agents when they start from given initial positions x_{10}, x_{20} and x_{30}. It has the form of the state equation of an unforced LTI system

$$\dot{x}(t) = Fx(t), \quad x(0) = x_0,$$

and we know that if F has all eigenvalues strictly inside the left half plane, $x(t)$ will converge to zero for any initial state x_0 as $t \to \infty$. A closer inspection of the matrix F reveals that all row sums are zero. Define

$$\mathbf{1} = [1 \;\; 1 \ldots 1]^T \in \mathbb{R}^N$$

as an N-dimensional column vector with all elements equal to 1, where N is the number of agents, then we have

$$F\mathbf{1} = 0 \cdot \mathbf{1},$$

thus F has a zero eigenvalue with corresponding eigenvector $\mathbf{1}$. The matrix F was obtained from the consensus protocol (5.3) - which represents the information contained in the communication graph - substituted into the agent models (5.2). In Section 5.2 we will formally define the *unscaled graph Laplacian* L^0 of a communication graph as the negative of this matrix, i.e.

$$L^0 = -F.$$

Thus, if λ is an eigenvalue of F, then $-\lambda$ is an eigenvalue of L^0. In our example, the eigenvalues of F are $\{0, -1, -3\}$, therefore the eigenvalues of L^0 are $\{0, 1, 3\}$. Moreover, we have

$$L^0\mathbf{1} = 0 \cdot \mathbf{1}. \tag{5.5}$$

The graph Laplacian L^0 introduced here is called unscaled in order to distinguish it from the scaled graph Laplacian discussed in Section 5.3. Since in this and the next section we will consider only unscaled Laplacians, we drop the attribute "unscaled". We now return to the question of whether the agents (5.2), equipped with the local control law (5.3) and starting from initial conditions x_{10}, x_{20} and x_{30}, will reach average consensus as formulated in (5.1). In terms of the graph Laplacian, (5.4) can be expressed as

$$\dot{x}(t) = -L^0 x(t). \tag{5.6}$$

Thus, possible equilibrium states are all vectors x that lie in the null space of L^0, which, as we saw above, includes $\mathcal{R}(\mathbf{1})$, the space spanned by $\mathbf{1}$. The solution to this vector differential equation is

$$x(t) = e^{-L^0 t}x(0). \tag{5.7}$$

In order to further investigate this solution, we first observe that the graph Laplacian in this example is symmetric. Symmetry of the graph Laplacian reflects the fact that the underlying communication graph is undirected. A consequence of this symmetry is that there exists a nonsingular matrix U such that

$$L^0 = U\Lambda U^T, \qquad \text{where} \qquad \Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix},$$

and U can be chosen to be orthogonal, i.e. such that $UU^T = I$. Substituting in (5.7) yields

$$x(t) = e^{-L^0 t}x(0) = e^{-U\Lambda U^T t}x(0) = Ue^{-\Lambda t}U^T x(0),$$

where

$$e^{-\Lambda t} = \begin{bmatrix} e^{-\lambda_1 t} & 0 & 0 \\ 0 & e^{-\lambda_2 t} & 0 \\ 0 & 0 & e^{-\lambda_3 t} \end{bmatrix},$$

(compare Exercise 1.4b, Lecture Notes *Control Systems Theory and Design*). Letting u_i denote the i^{th} column of U, we can then express the matrix exponential $e^{-L^0 t}$ as

$$e^{-L^0 t} = Ue^{-\Lambda t}U^T = e^{-\lambda_1 t}u_1 u_1^T + e^{-\lambda_2 t}u_2 u_2^T + e^{-\lambda_3 t}u_3 u_3^T.$$

The states of the agents then evolve as

$$x(t) = Ue^{-\Lambda t}U^T x(0) = \left(e^{-\lambda_1 t}u_1 u_1^T + e^{-\lambda_2 t}u_2 u_2^T + e^{-\lambda_3 t}u_3 u_3^T\right)x(0). \tag{5.8}$$

Recalling that $\lambda_1 = 0$ with eigenvector u_1, whereas λ_2 and λ_3 are strictly positive, we see that the second and third term on the right hand side of (5.8) will disappear as $t \to \infty$. With

$$u_1 = \frac{1}{\sqrt{3}}\mathbf{1}$$

where the eigenvector has been normalised such that $\|u_1\| = 1$, we obtain

$$\lim_{t\to\infty} x(t) = \frac{1}{3}\mathbf{1}\mathbf{1}^T x(0) = x_{\text{cons}}\mathbf{1} = \begin{bmatrix} x_{\text{cons}} \\ x_{\text{cons}} \\ x_{\text{cons}} \end{bmatrix}, \tag{5.9}$$

where x_{cons} is the average consensus position defined in (5.1). Thus average consensus is achieved.

Agreement Space

Let us summarise and try to generalise the observations obtained from the example discussed in this section, that involves three single-integrator agents, a communication graph as shown in Fig. 5.1 and the corresponding consensus protocol (5.3). We start with a definition.

Definition 5.1 *The* agreement space *associated with a group of N agents is*

$$\mathcal{A} = \mathcal{R}(\mathbf{1}).$$

The question we tried to answer was whether the three agents, starting from arbitrary initial positions, would reach average consensus, and we saw that this is indeed the case. Conditions required to derive this result were the fact that the graph Laplacian has a single eigenvalue at 0 with corresponding normalised eigenvector $1/\sqrt{N}\cdot\mathbf{1}$, and that all other eigenvalues are strictly positive - this is what leads from (5.8) to (5.9). It is thus straightforward to extend the reasoning leading to the above result to a group of N agents and derive the following result.

Theorem 5.1 *Given a group of N agents with single integrator dynamics*

$$\dot{x}_i(t) = u_i(t), \quad i = 1, 2, \ldots, N,$$

an undirected communication graph with graph Laplacian $L^0 = (L^0)^T \in \mathbb{R}^{N\times N}$ and a corresponding consensus protocol (5.3), the state vector $x = [x_1 \ \ldots x_N]^T$ governed by

$$\dot{x}(t) = -L^0 x(t), \quad x(0) = x_0$$

will converge to the agreement space $\mathcal{A}$ if and only if

$$\lambda_1(L^0) = 0 \quad \text{and} \quad \lambda_i(L^0) > 0, \quad i = 2, \ldots, N.$$

In this case we have

$$\lim_{t\to\infty} x(t) = \frac{1}{N}\mathbf{1}^T x_0.$$

In Section 5.2 we will present conditions that a communication graph must satisfy for the eigenvalue conditions of the theorem to hold. If we assume that the eigenvalues are ordered such that

$$0 = \lambda_1 < \lambda_2 \leq \lambda_3 \leq \ldots \leq \lambda_N,$$

then it follows from (5.8) that the speed of convergence is determined by the smallest non-zero eigenvalue λ_2 (sometimes referred to as the *Fiedler eigenvalue* or as the *algebraic connectivity* of the communication graph), i.e. the dominant time constant is $1/\lambda_2$.

Constant of Motion

The observations about consensus protocols made so far can be used to gain some further insight into the way a group of agents reaches the agreement space. From the vector differential equation

$$\dot{x}(t) = -L^0 x(t)$$

governing the motion of the individual agents, we obtain by left-multiplying with $\mathbf{1}^T$ (and noting that $L^0 = L^{0^T}$)

$$\mathbf{1}^T \dot{x}(t) = -\mathbf{1}^T L^0 x(t) = 0$$

or

$$\frac{d}{dt}\Big(\mathbf{1}^T x(t)\Big) = 0 \qquad \forall t \geq 0.$$

This simply says that the sum of the states of all agents $\mathbf{1}^T x(t)$ is constant over time; it is referred to as a *constant of motion* for the agreement dynamics. Since this is true for all $t \geq 0$, we have in particular

$$\mathbf{1}^T x(t) = \mathbf{1}^T x(0)$$

or

$$\mathbf{1}^T \Big(x(t) - x(0)\Big) = 0 \qquad \forall t \geq 0.$$

Thus, the state vector $x(t)$ will move along a straight line orthogonal to the agreement space, starting from $x(0)$. This is illustrated for two agents in Fig. 5.3. One can easily check that the intersection of this orthogonal line passing through $x(0)$ with the agreement space is indeed the average of all initial state variables.

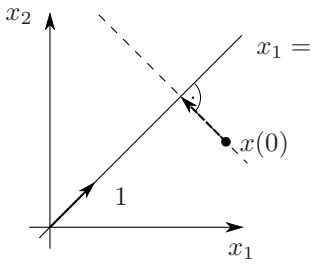


Figure 5.3: Consensus trajectory

5.2 Some Facts from Graph Theory

In this section we introduce some concepts and results from graph theory that are required for a more formal treatment of the consensus problem, and also for the analysis and synthesis of formation control schemes. First we formally define a graph. A graph is built upon a set with a finite number of elements, referred to as *vertex set*

$$V = \{v_1, v_2, \ldots, v_N\}.$$

In our context of multi-agent systems, each vertex v_i will represent an agent; N denotes the cardinality of the set V, written as $N = |V|$. We also consider the product space $V^2 = V \times V$ that contains all ordered pairs of vertices

$$V^2 = \{(v_i, v_j) |\, v_i, v_j \in V,\ i \neq j\}. \tag{5.10}$$

Ordered pairs means $(v_i, v_j) \neq (v_j, v_i)$. For example, if a vertex set V has three vertices, i.e. $V = \{v_1, v_2, v_3\}$, then V^2 contains six ordered pairs, referred to as *edges*. With a given vertex set V we can associate an *edge set*

$$E \subseteq V^2;$$

for example we can associate with the vertex set above the edge set

$$E = \{(v_1, v_2), (v_2, v_1), (v_2, v_3), (v_3, v_2)\}.$$

Now we are in a position to formally define a *graph*: a graph is a pair of two sets, a vertex set V and an associated edge set E; we let $\mathcal{G}(V, E)$ denote such a graph. The vertex set and edge set used in the above example together define the graph shown in Fig. 5.4.

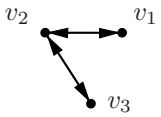


Figure 5.4: Graph with three vertices and four edges

Note that since we defined an edge as an ordered pair, this graph has four edges. In the context of multi-agent systems where vertices represent agents, an edge represents the flow of information from one agent to another. An edge (v_i, v_j) represents information flowing from v_j to v_i; we refer to v_i as the *head* and to v_j as the *tail* of the edge (when a graph is drawn, the head of an edge is indicated by the arrow head). We can now formalise the notions of directed and undirected graphs introduced in the previous section.

Definition 5.2 *A graph is undirected if*

$$(v_i, v_j) \in E \quad \Leftrightarrow \quad (v_j, v_i) \in E,$$

otherwise it is directed.

Undirected Graphs

For an undirected graph, a path is defined as a sequence of edges

$$\{v_{i_0}, v_{i_1}, v_{i_2}, \ldots, v_{i_m}\}$$

such that

$$(v_{i_k}, v_{i_{k+1}}) \in E \quad \forall k : \ 0 \leq k < m.$$

We say that an undirected graph is *connected* if there is a path between any $v_i, v_j \in V,\ i \neq j$. Thus, the graph shown in Fig. 5.4 is connected.

In an undirected graph, we define the degree of a vertex v_i as the cardinality of its neighbour set

$$d(v_i) = |\mathcal{J}_i|,$$

where the neighbour set that was introduced informally in the previous section can now be formally defined as

$$\mathcal{J}_i = \{v_j \in V | \, (v_i, v_j) \in E\},$$

and $d(v_i)$ is the number of neighbours of vertex v_i. For a given undirected graph $\mathcal{G}$, we next define three associated matrices. We first define the *degree matrix* as the diagonal matrix

$$\Delta(\mathcal{G}) = \begin{bmatrix} d(v_1) & & 0 \\ & \ddots & \\ 0 & & d(v_N) \end{bmatrix}.$$

The *unscaled adjacency matrix* of an undirected graph is defined element-wise as

$$\left[A^0(\mathcal{G})\right]_{ij} = \begin{cases} 1 & \text{if } \ (v_j, v_i) \in E, \\ 0 & \text{else.} \end{cases}$$

And finally we define the *Laplacian* of an undirected graph as

$$L^0(\mathcal{G}) = \Delta(\mathcal{G}) - A^0(\mathcal{G}).$$

For the graph shown in Fig. 5.4, we have

$$\Delta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{and} \qquad A^0 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Note that - following definition (5.10) of the product set V^2 where we excluded edges between a vertex and itself - the adjacency matrix has always zero diagonal entries. The Laplacian for this example is therefore

$$L^0 = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}.$$

This is exactly the Laplacian of the example in the previous section, which reflects the fact that the graph in Fig. 5.4 is the same as the communication graph considered in that section.

A useful result about the locations of the eigenvalues of L^0 and A^0 can be derived with the help of *Gersgorin's disk theorem*, which is stated next.

Theorem 5.2 *All eigenvalues of a matrix $M \in \mathbb{R}^{n\times n}$ are located in the union of disks*

$$\bigcup_{i=1}^{n} \left\{ z \in \mathbb{C} : \ |z - m_{ii}| \leq \sum_{j=1, j\neq i}^{n} |m_{ij}| \right\},$$

where m_{ij} denotes the (i, j) entry of M.

Using this theorem, it is straightforward to check that all eigenvalues of A^0 are located inside a disk centred at the origin with radius

$$\bar{d} = \max_i d(v_i),$$

whereas the eigenvalues of L^0 are located inside a disk centred at $\bar{d}$ with radius $\bar{d}$. Since A^0 and L^0 are symmetric for undirected graphs, the eigenvalues in this case are real, and an important consequence is that all eigenvalues of L^0 are nonnegative, and all eigenvalues of A^0 have a magnitude less than or equal to one.

We complete the discussion of undirected graphs with the following result.

Theorem 5.3 *Given an undirected graph $\mathcal{G}$, the following statements are equivalent:*

i) $\mathcal{G}$ *is connected*

ii) $\mathcal{N}(L^0) = \mathcal{R}(\mathbf{1})$*, i.e. the null space of the graph Laplacian is the agreement space*

iii) $\lambda_1 = 0$ *and* $\lambda_i > 0, \ i = 2, \ldots, N.$

The third statement is the necessary and sufficient condition of Theorem 5.1 for reaching average consensus. Note that because an eigenvalue of L^0 is either zero or positive (from Gersgorin's disk theorem), statement (iii) is equivalent to saying that L^0 has rank $N - 1$. In the previous section we saw that because of the special form of the graph Laplacian we have $(iii) \Rightarrow (ii)$; a proof of all equivalences can be found in [19].

Directed Graphs

According to Definition 5.2 a graph is directed if there is at least one edge $(v_i, v_j) \in E$ such that $(v_j, v_i) \notin E$. We let $\mathcal{D}(V, E)$ denote a directed graph, also referred to as *digraph*. Figure 5.5 shows an example of a directed graph: there is an edge (v_2, v_1) but no edge (v_1, v_2).

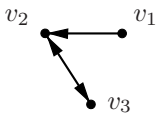


Figure 5.5: Directed graph

We can extend the definitions of matrices associated with undirected graphs to directed graphs as follows. First we need now to distinguish between the *in-degree* and the *out-degree* of a vertex: the in-degree $d_{\text{in}}(v_i)$ of vertex v_i counts the number of edges of which v_i is the head, whereas the out-degree $d_{\text{out}}(v_i)$ counts the number of edges of which v_i is the tail. We then define the *in-degree matrix* of a directed graph as

$$\Delta(\mathcal{D}) = \begin{bmatrix} d_{\text{in}}(v_1) & & \\ & \ddots & \\ & & d_{\text{in}}(v_N) \end{bmatrix}.$$

The unscaled adjacency matrix $A^0(\mathcal{D})$ of a directed graph is defined exactly as for an undirected graph, and the graph Laplacian of a directed graph is

$$L^0(\mathcal{D}) = \Delta(\mathcal{D}) - A^0(\mathcal{D}).$$

For the digraph in Fig. 5.5 we have

$$\Delta(\mathcal{D}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad A^0(\mathcal{D}) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

and therefore

$$L^0(\mathcal{D}) = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}.$$

As in the case of undirected graphs, it is clear from the way the graph Laplacian L^0 is constructed that the row sums are still zero, and that we have in this case as well

$$L^0\mathbf{1} = 0,$$

i.e. there is at least one zero eigenvalue with corresponding eigenvector $\mathbf{1}$.

Some further properties of directed graphs are explored in an Exercise.

5.3 Formation Control

In Section 5.1 we considered a consensus problem involving a group of N first order agents modelled as (5.2), each equipped with local control law (5.3) that involves the use of

information received from neighbouring agents. Combining individual agent states x_i into an overall state vector x, we saw that the states are governed by

$$\dot{x}(t) = -L^0 x(t), \tag{5.11}$$

where L^0 is the unscaled Laplacian of the underlying undirected communication graph. We also found that because the Laplacian satisfies $L^0\mathbf{1} = 0$, all agents converge to the average of their initial values as $t \to \infty$.

We will now extend this approach to a group of N agents that can be modelled as LTI systems

$$P(s): \quad \left\{ \begin{array}{rcl} \dot{x}_i(t) & = & A_P\, x_i(t) + B_P\, u_i(t) \\ y_i(t) & = & C_P\, x_i(t). \end{array} \right. \tag{5.12}$$

Here we assume $x_i \in \mathbb{R}^n$ for the agent states: whereas in Section 5.1 the agents where assumed to be single integrators (points with a velocity imposed on them), they are now allowed to be general dynamic LTI systems representing e.g. linearised models of mobile robots, unmanned aerial vehicles etc. The output y_i does not need to be scalar; here we assume it to be the position of agent i that is being transmitted to its neighbours. Thus we assume $y_i \in \mathbb{R}^p$ and $u_i \in \mathbb{R}^m$, where p is typically 2 or 3 (the dimension of the space in which agents are moving).

Each agent is equipped with a local controller

$$K(s): \quad \left\{ \begin{array}{rcl} \dot{\zeta}_i(t) & = & A_K\, \zeta_i(t) + B_K\, e_i(t) \\ u_i(t) & = & C_K\, \zeta_i(t) \end{array} \right. \tag{5.13}$$

with $\zeta_i \in \mathbb{R}^{n_K}$. We can define the position error $e_i(t) \in \mathbb{R}^p$ that forms the input to the controller of agent i as

$$e_i(t) = \sum_{j \in \mathcal{J}_i} \Big(y_i(t) - y_j(t) \Big). \tag{5.14}$$

As before, we introduce stacked column vectors representing states, inputs, outputs and position errors of the whole multi-agent system as

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}, \quad u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \quad \text{and} \quad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix},$$

respectively, where $x \in \mathbb{R}^{Nn}$, $u \in \mathbb{R}^{Nm}$ and $y, e \in \mathbb{R}^{Np}$. We assume again that the underlying communication graph is undirected. In order to represent the relationship (5.14) between position error and agents' positions in terms of the unscaled Laplacian L^0 of the communication graph, we need to modify the approach we took in Section 5.1 because the individual positions y_i and position errors e_i are now p-dimensional vectors. Assuming for a moment that y_i and e_i are scalar, it is easily verified that we have

$$e(t) = L^0 y(t).$$

Note that in contrast to (5.4) there is no minus sign attached to L^0 because in (5.14) the order of the terms in the bracket has been reversed compared with (5.3), i.e. the minus sign has been absorbed into the controllers $K(s)$.

To handle p-dimensional output and error vectors, we can use the *Kronecker product* and write

$$e(t) = (L^0 \otimes I_p)y(t),$$

where I_p denotes the $p \times p$ identity matrix; thus on the right hand side an element $[L^0]_{ij}$ multiplies position y_j of agent j to generate a term that contributes to the error e_i of agent i.

We will now slightly modify the input to the controller: we replace the error signal $e_i(t)$ in (5.14) by the weighted error

$$e_i(t) = \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} \Big(y_i(t) - y_j(t) \Big). \tag{5.15}$$

With this redefined error we obtain

$$e(t) = (L \otimes I_p)y(t), \tag{5.16}$$

where L is the *scaled graph Laplacian*, which is related to the unscaled Laplacian by

$$L = \Delta^{-1} L^0 = I - A, \tag{5.17}$$

and where $A = \Delta^{-1}A^0$ is the *scaled adjacency matrix* (note that the cardinality $|\mathcal{J}_i|$ of the neighbourhood of agent i equals $d(v_i)$). Returning to the undirected communication graph shown in Fig. 5.1, we have

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \end{bmatrix} \qquad \text{and} \qquad L = \begin{bmatrix} 1 & -1 & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & -1 & 1 \end{bmatrix}.$$

In this section we will find it convenient to use the scaled rather than the unscaled Laplacian to derive a formation control scheme (and thus follow the approach in [20]). One disadvantage of the scaled Laplacian is however evident from the above example: even for undirected graphs the scaled Laplacian is in general not symmetric. An important property that is preserved however is that

$$L\mathbf{1} = 0.$$

This property is the basis for reaching average consensus in the example studied in Section 5.1. That it will work also when agents transmit p-dimensional position vectors to their neighbours can be seen as follows: assume that all agents have reached consensus on a common position $y_{\text{cons}} \in \mathbb{R}^p$, i.e. that

$$y_{\text{cons}} = y_1 = y_2 = \ldots = y_N. \tag{5.18}$$

Then the position vector of the multi-agent system can be written as

$$y = \mathbf{1} \otimes y_{\text{cons}}.$$

Substituting this in (5.16) yields

$$e = (L \otimes I_p)(\mathbf{1} \otimes y_{\text{cons}}). \tag{5.19}$$

To evaluate this expression, we use the *mixed-product property* of the Kronecker product:

Lemma 5.1 *Let $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{r\times s}$, $C \in \mathbb{R}^{n\times p}$ and $D \in \mathbb{R}^{s\times t}$. Then*

$$(A \otimes B)(C \otimes D) = AC \otimes BD \quad \in \mathbb{R}^{mr\times pt}. \tag{5.20}$$

A proof of this fact and more properties of the Kronecker product can be found in **La05**, Chapter 13 (the proof consists of representing the matrices A and C in (5.20) in terms of matrix elements a_{ij} and c_{kl}, respectively, and using the rules of matrix multiplication). In the following treatment of formation control schemes we will frequently use the mixed-product property of the Kronecker product.

If we apply (5.20) to (5.19), we obtain

$$e = L\mathbf{1} \otimes y_{\text{cons}} = 0,$$

which shows that we can extend the notion of agreement space to vector-valued agent outputs: when the agents reach consensus on a position $y_{\text{cons}} \in \mathbb{R}^P$ in the sense of (5.18), the resulting output vector $y \in \mathbb{R}^{Np}$ will be in the null space of $L \otimes I_p \in \mathbb{R}^{Np\times Np}$. For brevity we will frequently use the notation $L_{(p)} = L \otimes I_p$.

Returning to the agent and controller models introduced above, we now consider a scenario where a group of N agents with dynamic models (5.12), equipped with local controllers (5.13) that realise an underlying undirected communication topology with scaled graph Laplacian L, are required to reach consensus with respect to their positions. In practical applications, where the agents may represent mobile robots, UAVs etc., it would usually be undesirable for them to converge all to the same point in space; one would rather have them attain a *formation*, i.e. a specified geometric pattern where each agent's position is specified relative to the positions of its neighbours. Introducing a reference vector

$$r^T = [r_1^T \;\; r_2^T \;\; \dots \;\; r_N^T],$$

that specifies the shape of the desired formation, we can represent the information flow between agents as shown in Fig. 5.6, where the $P(s)$-blocks represent agents and the $K(s)$-blocks the local controllers. Note that an absolute reference position r_i for agent i has only an effect on the relative position of that agent with respect to its neighbours. To see this, assume that two reference position vectors $r \in \mathbb{R}^{Np}$ and $\tilde{r} \in \mathbb{R}^{Np}$ differ only by a constant offset $\Delta r \in \mathbb{R}^p$, i.e.

$$r_i - \tilde{r}_i = \Delta r, \qquad i = 1, 2, \dots, N,$$

or

$$r - \tilde{r} = \mathbf{1} \otimes \Delta r.$$

Then the error vector e will be independent of that offset because

$$e = L_{(p)}(r + y) = L_{(p)}(\tilde{r} + \mathbf{1} \otimes \Delta r + y) = L_{(p)}(\tilde{r} + y).$$

For the same reason it does not matter whether the feedback signal y is added to or subtracted from the reference r; here we consider positive feedback.

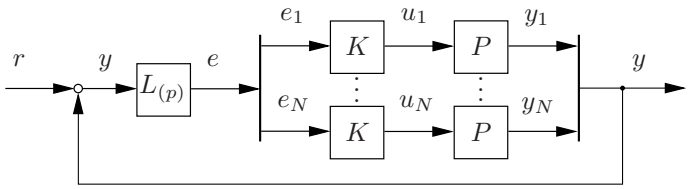


Figure 5.6: Group of N agents $P(s)$ with formation controllers $K(S)$ and formation reference r

The feedback loop shown in Fig. 5.6 represents a group of N LTI agents $P(s)$, each transmitting its output y_i to which a reference r_i is added. The block labelled $L_{(n)}$ determines the distribution of information between agents; the formation errors e_i are taken as inputs to the local controllers $K(s)$. To analyse the dynamic properties of this feedback system, we represent the dynamics of the whole group of N agents as

$$\begin{aligned} \dot{x}(t) &= (I_N \otimes A_P)x(t) + (I_N \otimes B_P)u(t) \\ y(t) &= (I_N \otimes C_P). \end{aligned}$$

The matrices in this state space model are block diagonal, with each of the matrices of the agent model (A_P, B_P, C_P) repeated N times along the diagonal. For brevity we introduce the notation

$$\hat{M} = I_N \otimes M$$

for such block diagonal matrices. Thus, the overall agent dynamics can be represented as

$$\begin{aligned} \dot{x}(t) &= \hat{A}_P x(t) + \hat{B}_P u(t) \\ y(t) &= \hat{C}_P x(t). \end{aligned} \tag{5.21}$$

Similary, the controller dynamics are

$$\begin{aligned} \dot{\zeta}(t) &= \hat{A}_K \zeta(t) + \hat{B}_K e(t) \\ u(t) &= \hat{C}_K \zeta(t). \end{aligned} \tag{5.22}$$

We let

$$\hat{P}(s) = I_N \otimes P(s) \qquad \text{and} \qquad \hat{K}(s) = I_N \otimes K(s)$$

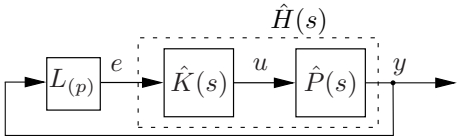


Figure 5.7: Group of N agents in formation control loop

denote the systems with state space realisations (5.21) and (5.22), respectively. A more compact form of the feedback loop shown in Fig. 5.6 is shown in Fig. 5.7, where we assume for simplicity that $r = 0$.

We let

$$\hat{H}(s) = \hat{P}(s)\hat{K}(s)$$

denote the series connection of agents and plants (note that $\hat{H}(s)$ is block-diagonal), and consider a state space realisation of this transfer function

$$\hat{H}(s) = \left[\begin{array}{c|c} A_H & B_H \\ \hline C_H & 0 \end{array}\right] = \left[\begin{array}{c|c} \hat{A}_P & \hat{B}_P \\ \hline \hat{C}_P & 0 \end{array}\right] \left[\begin{array}{c|c} \hat{A}_K & \hat{B}_K \\ \hline \hat{C}_K & 0 \end{array}\right].$$

It is straightforward to verify that

$$A_H = \begin{bmatrix} \hat{A}_P & \hat{B}_P\hat{C}_K \\ 0 & \hat{A}_K \end{bmatrix}, \quad B_H = \begin{bmatrix} 0 \\ \hat{B}_K \end{bmatrix}, \quad C_H = [\hat{C}_P \;\; 0]. \tag{5.23}$$

With $\hat{H}(s)$ representing $\hat{P}(s)\hat{K}(s)$, the loop shown in Fig. 5.7 is redrawn as in Fig. 5.8.

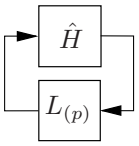


Figure 5.8: Group of N agents in formation control loop

Using the state space realisation (A_H, B_H, C_H) of $\hat{H}(s)$, this is equivalent to the loop shown in Fig. 5.9.

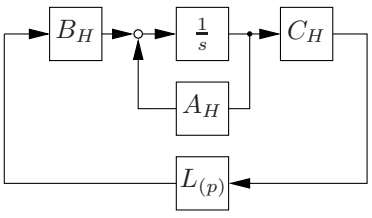


Figure 5.9: Group of N agents in formation control loop

a) formation feedback loop b) single integrator consensus loop

Figure 5.10: Comparison of formation feedback loop (a) and single integrator consensus loop (b)

It is now instructive to compare this feedback loop that represents a group of N agents in a formation control loop, shown slightly rearranged in Fig. 5.10.a, with the feedback loop that represents the dynamics of N single integrators with a consensus protocol governed by (5.6), shown in Fig. 5.10.b. We saw in Section 5.1 that the loop in Fig. 5.10.b results in position consensus as equilibrium state because $\dot{x} = L\mathbf{1} = 0$. The same is true for the formation control loop, as we will now illustrate with an example.

Example 5.1 *Consider a group of N agents, each modelled as*

$$P(s): \quad \begin{cases} \begin{bmatrix} \dot{y}_i \\ \dot{v}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_i \\ v_i \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i \\ y_i(t) = [1 \;\; 0] \begin{bmatrix} y_i \\ v_i \end{bmatrix}, \end{cases}$$

where u_i and y_i are scalar, i.e. agents are moving only in one spatial dimension. Each agent is equipped with a formation controller $K(s)$, and the goal is to have all agents converge into a single point y_{cons} as $t \to \infty$ (in the case where $r = 0$, or to attain a specified formation if $r \neq 0$). A condition for this to be possible is that $y = \mathbf{1} \otimes y_{\text{cons}}$ is compatible with an equilibrium state of the closed-loop system, i.e. with $[\dot{x}^T \;\; \dot{\zeta}^T]^T = 0$.

For $y = \mathbf{1} \otimes y_{\text{cons}}$ to hold in an equilibrium, we need the velocities v_i and controller states ζ_i to satisfy $v_i = 0$ and $\zeta_i = 0$ for $i = 1, \ldots, N$, respectively, or

$$\bar{x} = [y_{\text{cons}} \;\; 0 \;\; y_{\text{cons}} \;\; 0 \;\; \ldots \;\; y_{\text{cons}} \;\; 0]^T, \qquad \text{and} \qquad \bar{\zeta} = 0,$$

with a closed-loop equilibrium state

$$\begin{bmatrix} \bar{x} \\ \bar{\zeta} \end{bmatrix} = \left[\begin{array}{c} y_{\text{cons}} \\ 0 \\ \vdots \\ y_{\text{cons}} \\ 0 \\ \hdashline 0 \\ \vdots \\ 0 \end{array}\right]. \tag{5.24}$$

Referring to Fig. 5.11, we find that

$$C_H = [\hat{C}_P \;\; 0] = \left[\begin{array}{cccccc:ccc} 1 & 0 & & & & & 0 & \dots & 0 \\ & & 1 & 0 & & & & & \\ & & & & \ddots & & & \ddots & \\ & & & & & 1 \;\; 0 & 0 & \dots & 0 \end{array}\right]$$

and thus

$$\xi = C_H \begin{bmatrix} \bar{x} \\ \bar{\zeta} \end{bmatrix} = \mathbf{1} \otimes y_{\text{cons}}. \tag{5.25}$$

The input to the block $L_{(p)}$, the signal vector $\xi = \mathbf{1} \otimes y_{\text{cons}}$, is therefore in the null space of the Laplacian L, and the contribution to the summing junction at the integrator input in Fig. 5.11 coming from the lower branch is zero. We also have

$$A_H = \begin{bmatrix} \hat{A}_P & \hat{B}_P\hat{C}_K \\ 0 & \hat{A}_K \end{bmatrix} = \left[\begin{array}{cccccc:c} 0 & 1 & & & & & \\ 0 & 0 & & & & & \\ & & 0 & 1 & & & * \\ & & 0 & 0 & & & \\ & & & & & \ddots & \\ \hdashline 0 & & \dots & & & 0 & * \end{array}\right]$$

where $(*)$ denotes matrix blocks we are not interested in (since $\bar{\zeta} = 0$), and therefore

$$A_H \begin{bmatrix} \bar{x} \\ \bar{\zeta} \end{bmatrix} = 0. \tag{5.26}$$

Thus both contributions to the summing junction at the integrator input in Fig. 5.11 are zero, and we have

$$\begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} = 0,$$

which shows that $y = \mathbf{1} \otimes y_{\text{cons}}$ is compatible with an equilibrium state.

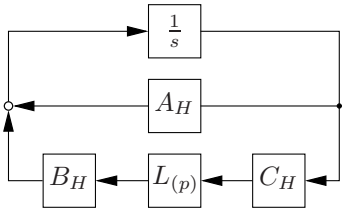


Figure 5.11: Formation feedback loop, Example 5.1

This example illustrates that position consensus $\bar{y} = \mathbf{1} \otimes y_{\text{cons}}$ is compatible with an equilibrium of the feedback loop shown in Fig. 5.6 and in Fig. 5.7. In order to establish that the agents do indeed reach this consensus, we must show that it is a *stable* equilibrium. We will do this by showing that the closed loop system in Fig. 5.11 is *marginally* stable. The state equation of this closed-loop system is

$$\begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} = (A_H + B_H L_{(p)} C_H) \begin{bmatrix} x \\ \zeta \end{bmatrix} \tag{5.27}$$

Note that this system has an eigenvalue at zero, because it follows from (5.25) and (5.26) that for the closed-loop equilibrium state (5.24) we have

$$(A_H + B_H L_{(p)} C_H) \begin{bmatrix} \bar{x} \\ \bar{\zeta} \end{bmatrix} = 0;$$

this is a necessary condition for the existence of a non-zero equilibrium. What is also required for reaching consensus is that the other eigenvalues of the closed-loop system (5.27) are strictly inside the left half of the complex plane.

Substituting from (5.23) in (5.27) yields

$$\begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} \hat{A}_P & \hat{B}_P \hat{C}_K \\ \hat{B}_K L_{(p)} \hat{C}_P & \hat{A}_K \end{bmatrix} \begin{bmatrix} x \\ \zeta \end{bmatrix}. \tag{5.28}$$

For the group of N agents $P(s)$ to reach consensus on a position y_{cons}, the block-diagonal controller

$$\hat{K}(s) = \begin{bmatrix} K(s) & & \\ & \ddots & \\ & & K(s) \end{bmatrix} = \left[\begin{array}{c|c} \hat{A}_K & \hat{B}_K \\ \hline \hat{C}_K & 0 \end{array}\right]$$

must stabilise (5.28) in the sense that all eigenvalues except for a single zero eigenvalue are inside the left half plane. We now present a well-known result that states a necessary and sufficient condition for $\hat{K}(s)$ to stabilise the closed-loop system.

Theorem 5.4 *The block diagonal controller $\hat{K}(s)$ stabilises (5.28) if and only if $K(s)$ stabilises the loop shown in Fig. 5.12 for all eigenvalues $\lambda_i(L)$, $i = 1, \ldots, N$ of the graph Laplacian.*

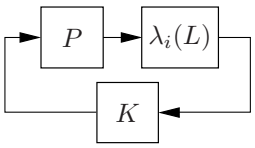


Figure 5.12: Single-agent loop for Theorem 5.4

This result was first proposed in [20]; it reduces the problem of assessing whether a formation controller $K(s)$ stabilises the feedback loop (5.28), the order of which depends on N and can be very large, to assessing whether it stabilises the small (size of a single agent) loop in Fig. 5.12 for all eigenvalues of L. Thus, it turns a possibly very large stability analysis problem into a small robust stability analysis problem.

We will present a proof of Theorem 5.4 that is based on diagonalising the graph Laplacian L. This requires the restrictive assumption that L is diagonalisable; recall that we are working with the *scaled* Laplacian which is in general not symmetric even for undirected graphs. Since the theorem is only concerned with stability and not with performance, it would be enough to apply a Schur decomposition and upper-triangularise L, which is always possible and enough for proving stability (this is in fact what is done in [20]). Here we nevertheless assume that we can diagonalise L because we want to take this discussion further to study also the performance of a group of agents forming a formation control loop. If we are considering undirected communication graphs and want to exploit the symmetry of the unscaled Laplacian, the Theorem can be modified to be applicable to the unscaled Laplacian.

To outline the idea of the following proof, consider the system matrix of the closed-loop system (5.28), which is a 2×2-block matrix. Three of the blocks are block diagonal, the only one that is not is the (2,1)-block $\hat{B}_K L_{(p)} \hat{C}_P$.That this block is not block-diagonal is due to $L_{(p)}$; the non-zero off-diagonal entries in this block represent communication between agents. If we assume for a moment that there is no communication between agents (i.e. $L = I$), all four blocks of the system matrix in (5.28) are block diagonal, and stability of the overall feedback system is equivalent to stability a single agent $P(s)$ in feedback with its local controller $K(s)$ (we are assuming that all agents and controllers, respectively, are identical). This fact is however not immediately obvious from the system matrix in (5.28); it seems not clear how block diagonality simplifies the stability analysis, since even though each of the four blocks of this block matrix is block diagonal, the system matrix itself is not.

To clarify this point, we can bring in a permutation matrix, as illustrated by a small example.

Example 5.2 *(Block-diagonalising permutation)*

Consider a group of two agents, assume $L = I$ (i.e. no communication) and write (5.28)

as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{\zeta}_1 \\ \dot{\zeta}_2 \end{bmatrix} = \left[\begin{array}{cc:cc} A_P & 0 & \Phi & 0 \\ 0 & A_P & 0 & \Phi \\ \hdashline \Psi_1 & 0 & A_K & 0 \\ 0 & \Psi_2 & 0 & A_K \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ \zeta_1 \\ \zeta_2 \end{bmatrix}, \tag{5.29}$$

where Φ and Ψ_i, $i = 1, 2$, denote the off-diagonal coupling blocks $\hat{B}_P\hat{C}_K$ and $\hat{B}_K\hat{C}_P$, respectively, between x and ζ (for later reference we allow the Ψ_i-blocks to be different). The system matrix is not block diagonal, but can be made so by swapping rows and columns. Define

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

and note that $\Pi^T\Pi = I$. Applying the similarity transformation

$$\begin{bmatrix} x_1 \\ \zeta_1 \\ x_2 \\ \zeta_2 \end{bmatrix} = \Pi \begin{bmatrix} x_1 \\ x_2 \\ \zeta_1 \\ \zeta_2 \end{bmatrix}$$

then yields

$$\begin{bmatrix} \dot{x}_1 \\ \dot{\zeta}_1 \\ \dot{x}_2 \\ \dot{\zeta}_2 \end{bmatrix} = \left[\begin{array}{cc:cc} A_P & \Phi & 0 & 0 \\ \Psi_1 & A_K & 0 & 0 \\ \hdashline 0 & 0 & A_P & \Phi \\ 0 & 0 & \Psi_2 & A_K \end{array}\right] \begin{bmatrix} x_1 \\ \zeta_1 \\ x_2 \\ \zeta_2 \end{bmatrix}. \tag{5.30}$$

In this transformed block-diagonal system matrix each block represents the closed-loop dynamics of a single agent.

Such a block diagonalising permutation is possible for any number of agents N when all four blocks of the system matrix in (5.28) are block diagonal, i.e. when there is no communication between agents. This provides the motivation for block-diagonalising the (2,1)-block $\hat{B}_K L_{(p)} \hat{C}_P$ in (5.28): as a result, the system matrix can be transformed into block-diagonal form, and stability of the overall system can be assessed by checking stability of the diagonal blocks (this is also true when the system matrix is brought into block triangular form).

Before we now turn to diagonalising the term $\hat{B}_K L_{(p)} \hat{C}_P$, we present two useful facts about Kronecker products involving matrices of the type $\hat{M} = I_N \otimes M$ and $Z_{(n)} = Z \otimes I_n$.

Lemma 5.2 *Given $M \in \mathbb{R}^{n\times m}$ and $Z \in \mathbb{R}^{N\times N}$, we have*

$$Z_{(n)}\hat{M} = \hat{M}Z_{(m)}.$$

Proof: Apply the mixed-product property to obtain

$$(Z \otimes I_n)(I_N \otimes M) = Z \otimes M = (I_N \otimes M)(Z \otimes I_m).$$

Note that when Z is nonsingular, we have as a direct consequence of Lemma 5.2 that

$$Z_{(n)}^{-1} \hat{M} Z_{(m)} = \hat{M}. \tag{5.31}$$

Lemma 5.3 *Given a diagonalisable matrix $L \in \mathbb{R}^{N\times N}$ and a nonsingular matrix $Z \in \mathbb{R}^{N\times N}$ such that $Z^{-1}LZ = \Lambda$, where Λ is diagonal, we have*

$$Z_{(n)}^{-1} L_{(n)} Z_{(n)} = \Lambda_{(n)}.$$

Proof: Follows from

$$Z_{(n)}^{-1} L_{(n)} Z_{(n)} = (Z \otimes I_n)^{-1}(L \otimes I_n)(Z \otimes I_n) = (Z^{-1} \otimes I_n)(LZ \otimes I_n) = \Lambda \otimes I_n,$$

where we used the fact that

$$(Z \otimes I_n)^{-1} = Z^{-1} \otimes I_n.$$

Now we are ready to bring the closed-loop system matrix

$$\begin{bmatrix} \hat{A}_P & \hat{B}_P \hat{C}_K \\ \hat{B}_K L_{(p)} \hat{C}_P & \hat{A}_K \end{bmatrix}$$

into a form that can be made block-diagonal by permutation. We do this by applying the similarity transformation

$$\begin{bmatrix} Z_{(n)} & 0 \\ 0 & Z_{(n_K)} \end{bmatrix}.$$

Introducing new closed-loop state vectors $\tilde{x}$ and $\tilde{\zeta}$ of agents and controllers, respectively, defined by

$$x = Z_{(n)}\tilde{x}, \qquad \text{and} \qquad \zeta = Z_{(n_K)}\tilde{\zeta},$$

the above transformation results in an equivalent closed-loop system

$$\begin{bmatrix} \dot{\tilde{x}} \\ \dot{\tilde{\zeta}} \end{bmatrix} = \begin{bmatrix} \hat{A}_P & \hat{B}_P \hat{C}_K \\ \hat{B}_K \hat{C}_P \Lambda_{(n)} & \hat{A}_K \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{\zeta} \end{bmatrix}. \tag{5.32}$$

That the (1,1), (1,2) and (2,2)-blocks of the system matrix remain unchanged under this transformation follows directly from Lemma 5.2. To see why we obtain the (2,1)-block shown in (5.32), note first that from Lemma 5.2 we have

$$\hat{B}_K L_{(p)} \hat{C}_P = \hat{B}_K \hat{C}_P L_{(n)}.$$

Then again applying Lemma 5.2 together with Lemma 5.3 yields

$$Z_{(n_K)}^{-1} \hat{B}_K \hat{C}_P L_{(n)} Z_{(n)} = \hat{B}_K \hat{C}_P Z_{(n)}^{-1} L_{(n)} Z_{(n)} = \hat{B}_K \hat{C}_P \Lambda_{(n)}.$$

The transformed closed-loop system is now in the same form as the two-agent system (5.29) in Example 5.2, therefore a permutation can be used to bring it into the form of (5.30). The (2,1)-block

$$\begin{bmatrix} \Psi_1 & 0 \\ 0 & \Psi_2 \end{bmatrix}$$

in (5.29) corresponds to the block $\hat{B}_K\hat{C}_P\Lambda_{(n)}$ here. Note that whereas $\hat{B}_K\hat{C}_P$ has identical blocks repeated along the diagonal, right multiplication with $\Lambda_{(n)}$ results in each of these blocks to be multiplied by a different eigenvalue of L; for this reason we allowed the Ψ_i-blocks in the example to be different. For the transformed formation control loop (5.32) considered here, we conclude that this system is stable if and only if all the diagonal sub-systems

$$\begin{aligned} \dot{\tilde{x}}_i &= A_P\tilde{x}_i + B_P C_K\tilde{\zeta}_i \\ \dot{\tilde{\zeta}}_i &= \lambda_i B_K C_P\tilde{x}_i + A_K\tilde{\zeta}_i, \qquad i = 1,\ldots,N \end{aligned} \tag{5.33}$$

are stable. Note that these subsystems represent the closed-loop dynamics of a single agent, however with the output scaled by an eigenvalue λ_i of L, so that (5.33) describes exactly the feedback loop shown in Fig. 5.12. Thus, Theorem 5.4 has been proved.

5.4 Formation Control as Robust Control Problem

Theorem 5.4 provides a necessary and sufficient condition for stability of the formation control loop shown in Fig. 5.7. The loop consists of a group of N agents each modelled as $P(s)$, and each equipped with a local controller $K(s)$. A controller $K(s)$ stabilises the whole formation if and only if it stabilises the single-agent loop shown in Fig. 5.12. Thus, a problem of potentially huge size (if N is large) has been reduced to a small (single-agent size) *robust control problem*. For single-input single-output agents, this result can be given an interesting interpretation in terms of the *Nyquist stability criterion*; this is explored in Exercise 6.2.

In this section we will present a slightly different approach to the problem of analysing and synthesising formation control loops within a robust control framework; the results presented here are based on [21]. Instead of using the normalised Laplacian L to represent the communication structure we use the normalised adjacency matrix A. From Gersgorin's theorem (Theorem 5.2) it follows that the eigenvalues of A are all located inside the unit disk $\mathbb{D}$; this fact will be exploited to formulate a condition for robust stability of the formation. Since here we are interested only in stability, we ignore the external reference r and the output signal y in Fig. 5.7 and consider the block diagram in Fig. 5.13 as representation of the formation control loop.

From (5.17) the normalised Laplacian is related to the normalised adjacency matrix by

$$L = I - A.$$

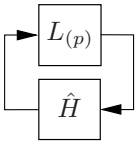


Figure 5.13: Formation control loop

Thus, the block diagram in Fig. 5.13 can be equivalently drawn as in Fig. 5.14.

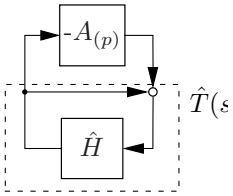


Figure 5.14: Equivalent representation of formation control loop

Note that the feedback loop denoted by $\hat{T}(s)$ is block diagonal and given by

$$\hat{T}(s) = \Big(I - \hat{H}(s)\Big)^{-1} \hat{H}(s),$$

where the blocks along the diagonal are

$$T(s) = \Big(I - H(s)\Big)^{-1} H(s),$$

and $H(s) = P(s)K(s)$, compare Fig. 5.7. Thus, each of these blocks represents a single-agent loop as shown in Fig. 5.15.

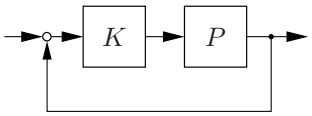


Figure 5.15: Single-agent feedback loop $T(s)$

To assess the stability of the formation control loop as represented in Fig. 5.14, one could use the *small gain theorem* (Theorem 21.1 in the Lecture Notes *Optimal and Robust Control*). The loop transfer function is

$$-A_{(p)}\hat{T}(s) = -(A \otimes I_p)\Big(I_N \otimes T(s)\Big) = -A \otimes T(s),$$

where the mixed-product rule (Lemma 5.1) was used to obtain the last equality. Applying the small gain theorem would yield as sufficient condition for stability that $A \otimes T(s)$ is stable and $\|A \otimes T(s)\|_\infty < 1$. Here we use a slightly stronger result: the *spectral radius*

theorem. The spectral radius $\rho(M)$ of a matrix $M \in \mathbb{R}^{n\times n}$ is defined as

$$\rho(M) = \max_i \ |\lambda_i(M)|,$$

i.e. it is the largest of the magnitudes of the eigenvalues of M.

Theorem 5.5 *Consider a system with transfer function $L(s)$ and a unity feedback loop connecting its input and output. The loop is stable if $L(s)$ is stable and*

$$\rho\Big(L(j\omega)\Big) < 1 \quad \forall\omega.$$

A proof of this result, which is based on the generalised Nyquist theorem for MIMO systems, can be found e.g. in [22]. Note that for each square matrix M we have

$$\rho(M) \leq \bar{\sigma}(M),$$

thus the spectral radius theorem is less conservative than the small gain theorem in the sense that the former is implied by the latter.

To employ Theorem 5.5 in a stability condition for the loop in Fig. 5.14, we need the following property of the Kronecker product. Let $\mathrm{eig}(M)$ denote the set of eigenvalues of M.

Lemma 5.4 *Given two square matrices X and Y, we have*

$$\mathrm{eig}(X \otimes Z) = \{\lambda_j\mu_k : \lambda_j \in \mathrm{eig}(X),\ \mu_k \in \mathrm{eig}(Y)\}\,.$$

This result states that the eigenvalues of a Kronecker product of two square matrices X and Y include all products between pairs of eigenvalues of X and Y, respectively; a proof can be found in **La05**, Chapter 13.

Now Theorem 5.5 requires two things to hold for stability: (i) that $A \otimes T(s)$ is stable, and (ii) that $\mathrm{eig}\Big(A \otimes T(j\omega)\Big) \in \mathbb{D}$. For (i) we note that $A \otimes T(s)$ is stable if and only if $T(s)$ is stable. And from Lemma 5.4 together with the fact that $\mathrm{eig}(A) \in \mathbb{D}$ it follows that (ii) holds true if $\|T(s)\|_\infty < 1$. In fact $\|T(s)\|_\infty < 1$ implies that (ii) holds true for all possible communication graphs $\mathcal{G}$, since $\mathrm{eig}(A) \in \mathbb{D}$ is true for all normalised adjacency matrices.

We then have the following result.

Theorem 5.6 *The formation control loop in Fig. 5.7 is stable for all possible communication graphs $\mathcal{G}$ if and only if $T(s)$ is stable and $\|T(s)\|_\infty < 1$.*

Note that here we have only shown that the above condition is sufficient; a proof that it is also necessary for robust stability under arbitrary communication graphs is given in [21]. An important feature of the above stability condition is that it does not require knowledge of the particular communication graph that determines the information exchange between agents in a given formation.

Exercises — Chapter 5

Problem 5.1 *(Directed Graphs)*

Learning Goals

- Apply the knowledge on convergence conditions of undirected graphs to directed graphs.

Task Description Let $\mathcal{D} = (\mathcal{V}, \mathcal{E})$ be a directed graph with n vertices and m edges, that represent a network of n agents with integrator dynamics. The network dynamics

$$\dot{x}(t) = u(t)$$

with $x, u \in \mathbb{R}^n$ and initial states $x_0 = x(0)$. To reach consensus the linear consensus protocol

$$u(t) = -L(\mathcal{D})x(t)$$

is used, with $L(\mathcal{D})$ being the Laplacian of $\mathcal{D}$. With this consensus protocol the networks dynamics is given as

$$\dot{x}(t) = -L(\mathcal{D})x(t).$$

Prove that the network dynamics converge to

$$\lim_{t\to\infty} x(t) = \left(q_1^T x_0\right)\mathbf{1}$$

with $q_1^T\mathbf{1} = 1$ and q_1 being the left eigenvector corresponding to the zero eigenvalue of $L(\mathcal{D})$, if and only if there exist at least one vertex $v \in \mathcal{V}$, from who every other vertex can be reached by respecting the direction of the edges.

Hint: Use Theorem F.2 for the proof and consider the proof for undirected graphs.

Problem 5.2 *(Convergence Rate)*

Learning Goals

- Recapitulate the consensus protocol and understand the definition of convergence rate.

Task Description Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a undirected graph with n vertices and m edges. Assume a group of n agents with integrator dynamics

$$\dot{x}_i(t) = u_i(t)$$

and initial states $x_i(0)$ for $i = 1, \ldots, n$. The agents are connected over a topology, represented by the graph $\mathcal{G}$ to agree on the average of the initial values, such that

$$\lim_{t\to\infty} x_i = \frac{1}{n}\sum_{j=1}^{n} x_j(0) \quad \forall i = 1, \ldots, n.$$

Therefore the linear consensus protocol

$$u_i(t) = \sum_{v_j \in \mathcal{J}_i} x_j(t) - x_i(t)$$

is used, which is a simplification of that in [23]. With the network states $x(t) = [x_1(t) \ldots x_n(t)]^T$ we get $u(t) = -Lx(t)$, where L is the Laplace matrix corresponding to $\mathcal{G}$. Given this consensus protocol the networks states evolves as

$$\dot{x}(t) = -Lx(t).$$

a) How many edges m are at least necessary, such that $\mathcal{G}$ is connected.

b) The graph with the maximal possible number of edges is called *complete graph*, where every vertex is connected to every other vertices. How many edges has the complete graph.

c) Plot the convergence rate of the consensus protocol as the number of edges increases in the previously determined possible range for $n = 5$. Start therefore with a graph with the minimal possible number of edges and add one iteratively. Repeat that for different numbers of vertices. What do you observe? *Matlab files are provided.*

d) Proof the observation of c) theoretically. Assume therefore a graph $\mathcal{G}$ with λ_2 the 2nd smallest eigenvalue of the respective Laplace matrix L. Let $\bar{\mathcal{G}}$ be $\mathcal{G}$ with one additional added edge and $\bar{\lambda}_2$ the 2nd smallest eigenvalue of the respective $\bar{L}$ and $\bar{x}$ the corresponding eigenvector. Hint: Use $\bar{x}^T \left(\bar{L} - L\right) \bar{x}$.

Problem 5.3 *(Time Delay)*

Learning Goals

- Understand the meaning of communication delays and their influence on stability.

Task Description In practice, when information is exchanged among agents through communication the signal transmission as well as data processing takes some finite time, that has to be considered. Therefore according to [23], [24] the consensus protocol used in the previous task changes to the delayed protocol

$$u_i(t) = \sum_{v_j \in \mathcal{N}_i} x_j(t - \tau_{ij}) - x_i(t - \tau_{ij}),$$

where τ_{ij} denotes the time it takes for information communicated by agent j to reach agent i. If we assume the simple case $\tau_{ij} = \tau$ that all delays are the same, then the delayed consensus protocol can be written as $u(t) = -Lx(t - \tau)$, such that the network states evolve as

$$\dot{x}(t) = -Lx(t - \tau).$$

a) Take a random undirected graph $\mathcal{G}$ with 8 vertices and 10 edges. Derive in simulation the minimal time delay $\tau^\star$ that leads to an unstable network. Start with $\tau = 0.1$. Repeat the same procedure for a graph $\bar{\mathcal{G}}$, where a random edge is added to the graph $\mathcal{G}$. Compare $\tau^\star$ and $\bar{\tau}^\star$, the minimal time delay that destabilizes the network connected by $\bar{\mathcal{G}}$. What do you suspect, how does the number of edges influences the robustness against time delays?

b) Derive the transition matrix $\Phi(s)$ such that $X(s) = \Phi(s)X(0)$ of the network dynamics.

c) Derive $\tau^\star$, such that for $\tau < \tau^\star$ the network is stable, formally. Use the transfer function derived in the previous task therefore and calculate its characteristic polynomial. Show that the maximal eigenvalue λ_n influences the robustness against time delays.

d) In the previous substask it has been shown that the $\tau^\star$ decreases with increasing λ_n. With this knowledge proof the observation of a) theoretically. The proof is similar to the previous Exercise d). Assume a graph $\mathcal{G}$ with λ_n the largest eigenvalue of the respective Laplace matrix L and respective eigenvalue x. Let $\bar{\mathcal{G}}$ be $\mathcal{G}$ with one additional added edge and $\bar{\lambda}_n$ the largest eigenvalue of the respective $\bar{L}$ and $\bar{x}$ the corresponding eigenvector. Discuss the trade-off between robustness and convergence rate.

Problem 5.4 *(Discrete Consensus)*

Learning Goals

- Transfer the knowledge for the continuous consensus protocol to the discrete case.

Task Description Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a undirected graph with n vertices and m edges. Assume a group of n agents with discrete time integrator dynamics

$$x_i(k+1) = x_i(k) + u_i(k)$$

and initial states $x_i(0)$ for $i = 1, \ldots, n$. The agents are connected over a topology, represented by the graph $\mathcal{G}$. The used consensus protocol is

$$u(k) = (W - I)x(k)$$

with $W_{ij} = 0$ if $i, j \notin \mathcal{E}$ and $i \neq j$. We assume in the following that W is symmetric. Then the network dynamics evolve as $x(k+1) = Wx(k)$.

a) Which properties of the spectrum of W are required, such that the network (i) is not unstable and (ii) converges to the average of the initial values.

b) How to define the convergence rate of the network dynamics in analogy to the continuous case? Hint: Note that in the case, where the continous consensus protocol is discretized with a sampling time T, than this would lead to $W = e^{-LT}$.

c) Assume $W = I - \alpha L$, where α is a real positive scalar and L the Laplacian corresponding to the graph $\mathcal{G}$. Do we get convergence to the average value? For what region of α is that discrete network stable? What is the optimal α with regard to convergence speed. Try different values of alpha in simulation.

Chapter 6

Formation Control with LTI and LPV Agents

The previous section presented with Theorem 5.4 a well-known result that allows to reduce the analysis and synthesis of distributed formation control schemes to a problem of the size of a single agent, regardless of the number of agents involved. So far we considered only stability; in this section we will introduce a framework that facilitates the extension of this approach to design for performance. Moreover, we will present an extension to formations of agents that are modelled as LPV systems. The framework we present in this chapter is referred to as the *decomposable systems* approach, first proposed in [25]. We will see that Theorem 5.4 can be seen as a special case of a more general result derived within this framework.

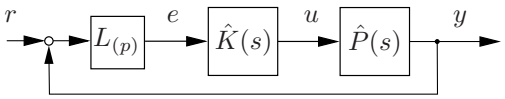


Figure 6.1: Formation control loop with reference input

When we discussed stability of formation control loops in the previous section, we ignored the reference input channel, i.e. we assumed $r = 0$. Because we want to extend the discussion to performance, we now bring the formation reference input back and extend the closed-loop state equation (5.28) into the state space model

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{\zeta} \end{bmatrix} &= \begin{bmatrix} \hat{A}_P & \hat{B}_P\hat{C}_K \\ \hat{B}_K L_{(p)}\hat{C}_P & \hat{A}_K \end{bmatrix} \begin{bmatrix} x \\ \zeta \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{B}_K L_{(p)} \end{bmatrix} r \\ y &= [\hat{C}_P \;\; 0] \begin{bmatrix} x \\ \zeta \end{bmatrix}. \end{aligned} \tag{6.1}$$

This loop is shown in Fig. 6.1. One way of characterising the performance of such a

formation control loop is to consider the response of the formation error to commanded changes in the formation reference r. A fictitious performance output can be defined e.g. as a linear combination of filtered formation error and filtered control effort, as shown in Fig. 6.2 (assuming that identical filters are used for all agents).

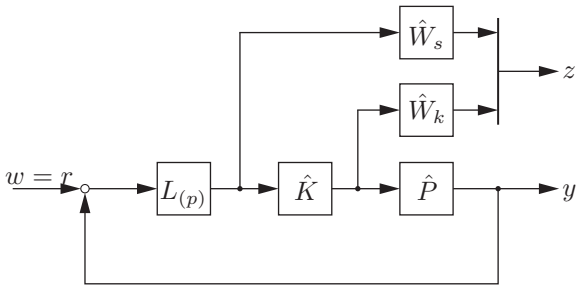


Figure 6.2: Formation control loop with shaping filters

To formulate and solve the analysis and synthesis problems for a formation of LPV agents, we will use an approach that was proposed in [10] and is based on LFT representations of LPV agent models and the *full block S-procedure*. This method can be seen as a refinement of the synthesis result for LPV systems in LFT form outlined in Section 4.2. In Section 6.3 we will present an analysis result for scheduled state feedback without derivation (for details see [10]), which will be used in Exercise 6.7 to design a distributed formation control scheme for a group of wheeled mobile robots.

Throughout this chapter, we will assume that the Laplacian L of the underlying communication graph is diagonalisable, i.e. that there exists a nonsingular matrix Z such that $Z^{-1}LZ$ is diagonal. When discussing formation control with LPV agents, we will even assume that $L = L^T$. In the previous chapter the symbol L was used to denote the scaled Laplacian $\Delta^{-1}L^0$, which is in general not symmetric even for undirected graphs. To satisfy the above assumptions, one could use the unscaled Laplacian, or the *normalised Laplacian* $\Delta^{-1/2}L^0\Delta^{-1/2}$, which are both symmetric if the graph is undirected. In this chapter we will use the symbol L to denote a suitable type of Laplacian that satisfies the required assumption of diagonalisability or symmetry, respectively.

6.1 Decomposable Systems

We begin with the definition of a *decomposable matrix*. This concept is based on the notion of a *pattern matrix* P that determines the interconnection structure between subsystems. In the context discussed in this chapter, we will interpret the pattern matrix as the Laplacian of a communication graph; the decomposable systems approach is however more general and P can also represent different interconnection structures.

Definition 6.1 *Given a pattern matrix $P \in \mathbb{R}^{N\times N}$ and a matrix $M \in \mathbb{R}^{Np\times Nq}$, the matrix M is called decomposable (with respect to P) if there exist matrices $M^d, M^c \in \mathbb{R}^{p\times q}$ such that*

$$M = I_N \otimes M^d + P \otimes M^c. \tag{6.2}$$

Note that this definition implies that a matrix $M \in \mathbb{R}^{k\times l}$ can only be decomposable w.r.t. $P \in \mathbb{R}^{N\times N}$ if k and l are integer multiples of N.

Definition 6.2 *Given a pattern matrix $P \in \mathbb{R}^{N\times N}$ and a state space realisation (A, B, C, D) of an LTI system, the system is called decomposable w.r.t. P if A, B, C and D are each decomposable w.r.t. P.*

It follows from (6.2) that any non-zero off-diagonal element of P represents communication between subsystems; this motivates the superscripts ^{d} and ^{c}, which stand for 'decoupled' and 'coupled', respectively. If the pattern matrix P is diagonalisable, we have the following result.

Lemma 6.1 *Given a diagonalisable pattern matrix $P \in \mathbb{R}^{N\times N}$ and a matrix $Z \in \mathbb{R}^{N\times N}$ such that $\det Z \neq 0$ and $Z^{-1}PZ = \Lambda$, where Λ is diagonal. Then for any matrix $M \in \mathbb{R}^{Np\times Nq}$ that is decomposable w.r.t. P, i.e. for which there exist M^d and M^c such that*

$$M = I_N \otimes M^d + P \otimes M^c,$$

we have

$$Z_{(p)}^{-1} M Z_{(q)} = I_N \otimes M^d + \Lambda \otimes M^c. \tag{6.3}$$

Clearly the right hand side of (6.3) is block diagonal - the lemma says that when Z diagonalises a pattern matrix P, then it can also be used to block-diagonalise any matrix that is decomposable w.r.t. P. This result can be easily shown by applying Lemma 5.2 and Lemma 5.3 to the left hand side of (6.3).

To illustrate the concept of a decomposable system and the significance of Lemma 6.1, we will apply it to a formation control problem. Consider again the formation control loop (6.1), assume that $N = 3$, and that the underlying communication graph is represented by the Laplacian

$$L = \begin{bmatrix} l_{11} & l_{12} & 0 \\ l_{21} & l_{22} & l_{23} \\ 0 & l_{32} & l_{33} \end{bmatrix}.$$

Then a state space model of the closed-loop formation dynamics is

$$
\left[\begin{array}{c} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \hdashline \dot{\zeta}_1 \\ \dot{\zeta}_2 \\ \dot{\zeta}_3 \end{array}\right] = \left[\begin{array}{ccc:ccc} A_P & & & B_P C_K & & \\ & A_P & & & B_P C_K & \\ & & A_P & & & B_P C_K \\ \hdashline l_{11} B_K C_P & l_{12} B_K C_P & 0 & A_K & & \\ l_{21} B_K C_P & l_{22} B_K C_P & l_{23} B_K C_P & & A_K & \\ 0 & l_{32} B_K C_P & l_{33} B_K C_P & & & A_K \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ \hdashline \zeta_1 \\ \zeta_2 \\ \zeta_3 \end{array}\right]
$$

$$
+ \left[\begin{array}{ccc} & 0 & \\ \hdashline l_{11} B_K & l_{12} B_K & 0 \\ l_{21} B_K & l_{22} B_K & l_{23} B_K \\ 0 & l_{32} B_K & l_{33} B_K \end{array}\right] \left[\begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array}\right]
$$

$$
\left[\begin{array}{c} y_1 \\ y_2 \\ y_3 \end{array}\right] = \left[\begin{array}{ccc:ccc} C_P & & & 0 & & \\ & C_P & & & & \\ & & C_P & & & \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ \hdashline \zeta_1 \\ \zeta_2 \\ \zeta_3 \end{array}\right] \tag{6.4}
$$

In contrast to Example 5.2, the (2,1) block of the closed-loop system matrix in (6.1) is not block diagonal, because we are assuming communication between agents. The same is true for the lower part of the input matrix.

If we now apply a state-permuting similarity transformation Π similar to the one that was used in Example 5.2, we obtain

$$
\left[\begin{array}{c} \dot{x}_1 \\ \dot{\zeta}_1 \\ \hdashline \dot{x}_2 \\ \dot{\zeta}_2 \\ \hdashline \dot{x}_3 \\ \dot{\zeta}_3 \end{array}\right] = \left[\begin{array}{cc:cc:cc} A_P & B_P C_K & & & & \\ l_{11} B_K C_P & A_K & l_{12} B_K C_P & & & \\ \hdashline & & A_P & B_P C_K & & \\ l_{21} B_K C_P & & l_{22} B_K C_P & A_K & l_{23} B_K C_P & \\ \hdashline & & & & A_P & B_P C_K \\ & & l_{32} B_K C_P & & l_{33} B_K C_P & A_K \end{array}\right] \left[\begin{array}{c} x_1 \\ \zeta_1 \\ \hdashline x_2 \\ \zeta_2 \\ \hdashline x_3 \\ \zeta_3 \end{array}\right]
$$

$$
+ \left[\begin{array}{c:c:c} 0 & & \\ l_{11} B_K & l_{12} B_K & \\ \hdashline & 0 & \\ l_{21} B_K & l_{22} B_K & l_{23} B_K \\ \hdashline & & 0 \\ & l_{32} B_K & l_{33} B_K \end{array}\right] \left[\begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array}\right]
$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \left[\begin{array}{cc:cc:cc} C_P & 0 & & & & \\ & & C_P & 0 & & \\ & & & & C_P & 0 \end{array}\right] \left[\begin{array}{c} x_1 \\ \zeta_1 \\ \hdashline x_2 \\ \zeta_2 \\ \hdashline x_2 \\ \zeta_3 \end{array}\right]. \tag{6.5}$$

We will use the more compact notation

$$\begin{aligned} \dot{\xi} &= A_F\,\xi + B_F r \\ y &= C_F\,\xi \end{aligned} \tag{6.6}$$

to denote this permuted closed-loop system. It is clear that this system is decomposable with respect to L; the model matrices can be expressed as

$$\begin{aligned} A_F &= I_2 \otimes A_F^d + L \otimes A_F^c \\ B_F &= I_2 \otimes B_F^d + L \otimes B_F^c \\ C_F &= I_2 \otimes C_F^d + L \otimes C_F^c, \end{aligned}$$

where

$$A_F^d = \begin{bmatrix} A_P & B_P C_K \\ 0 & A_K \end{bmatrix}, \quad A_F^c = \begin{bmatrix} 0 & 0 \\ B_K C_P & 0 \end{bmatrix}, \quad B_F^d = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad B_F^c = \begin{bmatrix} 0 \\ B_K \end{bmatrix},$$

and

$$C_F^d = [C_P \;\; 0], \quad C_F^c = [0 \;\; 0].$$

Moreover, if L is diagonalisable and $Z^{-1}LZ = \Lambda$, where Λ is diagonal, Lemma 6.1 tells us how we can block-diagonalise the system: we have

$$Z_{(n)}^{-1} A_F Z_{(n)} = I_N \otimes A_F^d + \Lambda_{(n)} \otimes A_F^c.$$

If we introduce the notation $\tilde{A}_F = Z_{(n)}^{-1} A_F Z_{(n)}$, we have

$$A_F = Z_{(n)} \tilde{A}_F Z_{(n)}^{-1},$$

and similarly

$$\begin{aligned} B_F &= Z_{(m)} \tilde{B}_F Z_{(n)}^{-1} \\ C_F &= Z_{(l)} \tilde{C}_F Z_{(n)}^{-1}. \end{aligned}$$

If we substitute these expressions in the permuted state space model (6.6), we obtain

$$\begin{aligned} Z_{(n)}^{-1}\dot{\xi} &= \tilde{A}_F Z_{(n)}^{-1}\xi + \tilde{B}_F Z_{(m)}^{-1} r \\ Z_{(l)}^{-1} y &= \tilde{C}_F Z_{(n)}^{-1}\xi. \end{aligned} \tag{6.7}$$

Defining new signals

$$\tilde{\xi} = Z_{(n)}^{-1}\xi, \qquad \tilde{r} = Z_{(m)}^{-1}r, \qquad \tilde{y} = Z_{(l)}^{-1}y, \tag{6.8}$$

we can rewrite (6.7) as

$$\begin{aligned} \dot{\tilde{\xi}} &= \tilde{A}_F\tilde{\xi} + \tilde{B}_F\tilde{r} \\ \tilde{y} &= \tilde{C}_F\tilde{\xi}. \end{aligned} \tag{6.9}$$

For the three-agent example above, the state equation in (6.5) is transformed into

$$\begin{bmatrix} \dot{\tilde{x}}_1 \\ \dot{\tilde{\zeta}}_1 \\ \dot{\tilde{x}}_2 \\ \dot{\tilde{\zeta}}_2 \\ \dot{\tilde{x}}_3 \\ \dot{\tilde{\zeta}}_3 \end{bmatrix} = \left[\begin{array}{cc:cc:cc} A_P & B_PC_K & & & & \\ \lambda_1 B_KC_P & A_K & & & & \\ \hdashline & & A_P & B_PC_K & & \\ & & \lambda_2 B_KC_P & A_K & & \\ \hdashline & & & & A_P & B_PC_K \\ & & & & \lambda_3 B_KC_P & A_K \end{array}\right] \begin{bmatrix} \tilde{x}_1 \\ \tilde{\zeta}_1 \\ \tilde{x}_2 \\ \tilde{\zeta}_2 \\ \tilde{x}_2 \\ \tilde{\zeta}_3 \end{bmatrix}$$

$$+ \left[\begin{array}{c:c:c} 0 & & \\ \lambda_1 B_K & & \\ \hdashline & 0 & \\ & \lambda_2 B_K & \\ \hdashline & & 0 \\ & & \lambda_3 B_K \end{array}\right] \begin{bmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{bmatrix}, \tag{6.10}$$

where λ_i, $i = 1, 2, 3$, denote the eigenvalues of L, while the output equation (which does not involve interaction between agents) is not altered by this transformation.

The block diagonal system matrix $\tilde{A}_F$ is similar to A_F, so marginal stability of (6.1) and of (6.9) are equivalent. This diagonalising transformation was illustrated with the three-agent example (6.4); it is clear that a decomposable system with any number N of subsystems can be brought into the form of (6.9), where the new system matrices are

$$\begin{aligned} \tilde{A}_F &= I_N \otimes A_F^d + \Lambda \otimes A_F^c \\ \tilde{B}_F &= I_N \otimes B_F^d + \Lambda \otimes B_F^c \\ \tilde{C}_F &= I_N \otimes C_F^d + \Lambda \otimes C_F^c. \end{aligned} \tag{6.11}$$

All system matrices are block diagonal, so we have a collection of N completely decoupled subsystems

$$\begin{bmatrix} \dot{\tilde{x}}_j \\ \dot{\tilde{\zeta}}_j \end{bmatrix} = \begin{bmatrix} A_P & B_PC_K \\ \lambda_j B_KC_P & A_K \end{bmatrix} \begin{bmatrix} \tilde{x}_j \\ \tilde{\zeta}_j \end{bmatrix} + \begin{bmatrix} 0 \\ \lambda_j B_K \end{bmatrix} \tilde{r}_j$$

$$\tilde{y}_j = \begin{bmatrix} C_P & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_j \\ \tilde{\zeta}_j \end{bmatrix}, \tag{6.12}$$

and stability of the overall system is equivalent to stability of all subsystems. Moreover, the subsystems differ only in a scalar - the eigenvalues of L - that multiplies the ^{c}-part of

the model matrices. These subsystems have the size of a single closed-loop agent; they are referred to as *modal subsystems*. Decomposition into modal subsystems is the key to turning analysis and synthesis problems for groups of N agents into robust analysis and synthesis problems for a single agent.

Vehicle Integral Action Required for Consensus

Note that because $\lambda_1 = 1$, the (1,1) block of the transformed closed-loop system matrix $\tilde{A}_F$ will have the eigenvalues of A_P and A_K, see (6.10). We know that as a condition for reaching consensus $\tilde{A}_F$ must have at least one zero eigenvalue. In formation control problems where A_P represents the system matrix of a vehicle model, this condition will be met because the vehicles display integral action.

While we refer to a system in the form of (6.6) as *decomposable*, we will call a system in the particular block diagonal form of (6.9) as a *decomposed system*.

Note that the transformation of the formation control loop (6.1) into the form of (6.9) involves not only a similarity transformation, but also the *signal transformation* (6.8). The reason for this is that we need not only to transform A_F into block diagonal form, but also B_F and C_F in order to completely decouple the system into its subsystems. Whereas this signal transformation has no effect on stability, it has an effect on the performance of the system.

Performance of Formation Control Loops

We consider the performance of a formation control loop in terms of the H_∞-norm of the closed-loop map from an external input to a performance output. Referring to Fig. 6.2, we assume a performance channel $w \to z$, where w can be thought of as formation reference input r, and z as a combination of filtered formation error and control effort.

The motivation for the diagonalising transformation that brings the system into the form (6.9) is that we want to solve a synthesis problem that has the size of a single agent, and to use the resulting controller for each member of the group of N agents, while being able to obtain guarantees on stability and performance of the whole group. As discussed above, this is possible for stability, since $\tilde{A}_F$ and A_F are similar. The situation is different when it comes to performance, as we shall now see.

Let $T(s)$ denote the closed-loop transfer function from r to z in the system shown in Fig. 6.2, and $\tilde{T}(s)$ that from $\tilde{r}$ to $\tilde{z}$ in the transformed system that has the form of (6.9) but with the physical output y replaced by the performance output z. When we tune a controller for performance, we do it by tuning a single subsystem $\tilde{T}_i(s)$ of $\tilde{T}(s)$, knowing that these subsystems are identical up to scaling by the eigenvalues of L. Since $\tilde{T}(s)$ is a parallel combination of all its subsystems $\tilde{T}_i(s)$, it can be represented as in Fig. 6.3.

The question of interest is now: what is the relationship between the performance $\|\tilde{T}(s)\|_\infty$ of the fictitious, transformed system we are working on, and the performance $\|T(s)\|_\infty$ of

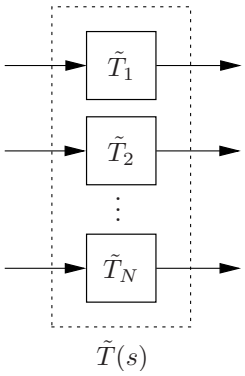


Figure 6.3: Transformed system

the actual formation loop? First we note that we have

$$\|\tilde{T}(s)\|_\infty = \max_i \|\tilde{T}_i(s)\|_\infty,$$

which we can use to tune a controller for good robust performance by minimising $\|\tilde{T}_i(s)\|_\infty$, while treating λ_i as uncertainty. Next, the relationship between $\tilde{T}(s)$ and $T(s)$ is determined by the signal transformation (6.8), and shown in Fig. 6.4. Using the interpretation of the H_∞-norm as the induced $\mathcal{L}_2$-norm, it is then straightforward to show that

$$\frac{\underline{\sigma}(Z)}{\bar{\sigma}(Z)}\|\tilde{T}(s)\|_\infty \leq \|T(s)\|_\infty \leq \frac{\bar{\sigma}(Z)}{\underline{\sigma}(Z)}\|\tilde{T}(s)\|_\infty. \tag{6.13}$$

These bounds were proposed in [25]. Note that if L is symmetric, we can choose Z to be orthogonal, so its condition number is 1 and we have equality in (6.13). If L is not symmetric but diagonalisable, one can scale Z to obtain upper and lower bounds as tight as possible, this is explored in [26].

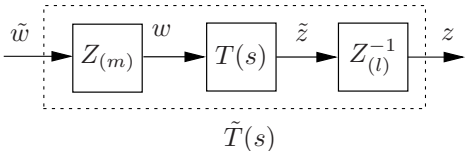


Figure 6.4: Signal transformation on formation control loop $T(s)$

6.2 Non-Holonomic Agents

To motivate the use of LPV models for agents in formation control problems, we will introduce a simple model of an actuated rolling disk, that can be used as a simplified model of wheeled mobile robots. Consider the disk shown in Fig. 6.5, which is rolling

without slipping in a horizontal plane. Here we work with the simplifying assumption that the disk will not tilt away from the vertical. Its position in the plane is given by the coordinates $x(t)$ and $y(t)$, and its orientation by the angle $\phi(t)$.

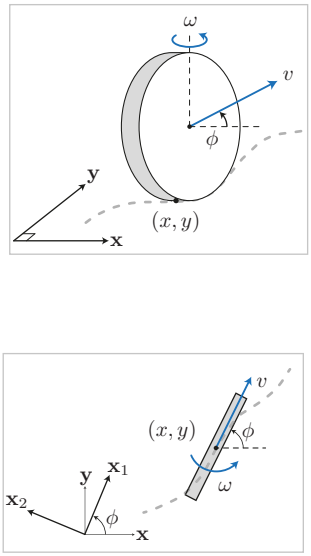


Figure 6.5: Rolling disk

This system is said to be subject to *non-holonomic constraints*, because it is restricted to move in the direction of its orientation. More generally, a mobile robot system is called non-holonomic if the number of controllable degrees of freedom is smaller than the number of total degrees of freedom. The constraint on the disk shown in Fig. 6.5 is typical for non-holonomic constraints on wheeled vehicles or robots: even though they have three total degrees of freedom (position in two axes and orientation), the two controllable degrees of freedom are acceleration (or braking) and steering.

A state space model of the disk is

$$\dot{x}(t) = v(t)\cos\phi(t) \tag{6.14}$$
$$\dot{y}(t) = v(t)\sin\phi(t) \tag{6.15}$$
$$\dot{\phi}(t) = \omega(t) \tag{6.16}$$

with three state variables x, z and ϕ and two control inputs v and ω. The state variables are assumed to be measured and available for feedback.

The non-holonomic constraint on the motion of this disk (no slipping) can be represented as

$$\dot{x}(t)\sin\phi(t) - \dot{y}(t)\cos\phi(t) = 0,$$

or

$$\frac{\dot{y}(t)}{\dot{x}(t)} = \tan\phi(t).$$

In order to construct an LPV model of this system, it is convenient to first apply a coordinate transformation

$$T(t) = \begin{bmatrix} \cos\phi(t) & \sin\phi(t) & 0 \\ -\sin\phi(t) & \cos\phi(t) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{6.17}$$

which leads to the model

$$\dot{\tilde{x}}(t) = v(t) + \omega(t)\tilde{y}(t) \tag{6.18}$$
$$\dot{\tilde{y}}(t) = -\omega(t)\tilde{x}(t) \tag{6.19}$$
$$\dot{\phi} = \omega(t). \tag{6.20}$$

This coordinate transformation aligns the $\tilde{x}$-direction with the orientation of the disk. An LPV model is then obtained by writing

$$\begin{bmatrix} \dot{\tilde{x}} \\ \dot{\tilde{y}} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \phi \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{6.21}$$

which has the form

$$\dot{x}_P(t) = A\Big(\omega(t)\Big)x_P(t) + Bu(t). \tag{6.22}$$

Note that this model is affine in the scheduling parameter ω and has a constant input matrix B.

6.3 Robust State Feedback Synthesis Using the Full Block S-Procedure

In this section we present results that were proposed in [10] and can be used for analysis and synthesis of LPV gain-scheduling controllers. This approach is based on the *full-block S-procedure*; its application to decomposable LTI systems was first reported in [27] for the case of state feedback. An extension to decomposable LPV systems and output feedback was presented in [28]. The approach assumes that the LPV model is represented in linear fractional form.

A Brief Review of LTI Systems

Before presenting a result that can be used for the synthesis of scheduled state feedback control, we briefly review some related facts about LTI systems. Recall that a system

$$\dot{x}(t) = Ax(t)$$

is stable if and only if there exists a symmetric, positive definite matrix X that satisfies

$$A^T X + XA < 0. \tag{6.23}$$

(Here we use X to denote the Lyapunov matrix since P is already used for agent models.) Note that an equivalent formulation of condition (6.23) is

$$[I \;\; A^T] \begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \begin{bmatrix} I \\ A \end{bmatrix} < 0. \tag{6.24}$$

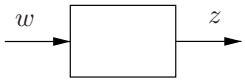


Figure 6.6: Dynamic system

As for performance, consider the system shown in Fig. 6.6. Assume that the system is linear time-invariant and stable, and has a state space realisation (A, B, C, D). Recall that $\|T\|_\infty < \gamma$ for a given $\gamma > 0$ if

$$\frac{\|z\|}{\|w\|} < \gamma \qquad \forall w \neq 0,$$

where we assume $x(0) = 0$. Note that an equivalent formulation is

$$\int_0^\infty [w^T \;\; z^T] \begin{bmatrix} -\gamma I & 0 \\ 0 & \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix} dt \; < \; 0 \qquad \forall w \neq 0. \tag{6.25}$$

Using (6.24) and (6.25), the following necessary and sufficient condition for stability and performance can be derived in a way similar to the the derivation of the bounded real lemma: T is stable and $\|T\|_\infty < \gamma$ if and only if there exists a symmetric $X > 0$ that satisfies

$$\left[\begin{array}{cc:cc} I & A^T & 0 & C^T \\ 0 & B^T & I & D^T \end{array}\right] \left[\begin{array}{cc:cc} 0 & X & 0 & 0 \\ X & 0 & 0 & 0 \\ \hdashline 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} I \end{array}\right] \left[\begin{array}{cc} I & 0 \\ A & B \\ \hdashline 0 & I \\ C & D \end{array}\right] \; < \; 0. \tag{6.26}$$

It is straightforward to check that this inequality is equivalent to

$$\begin{bmatrix} A^T X + X A & X B & C^T \\ B^T X & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0.$$

Systems with Time-Varying Uncertainty

Next we assume that the system shown in Fig. 6.6 is linear but depends on uncertain time-varying parameters, which are collected in a matrix $\Delta(t)$, and that $\Delta(t) \in \mathbf{\Delta} \; \forall t \geq 0$ for some compact set of admissible values $\mathbf{\Delta}$. Let Σ_Δ denote this uncertain, time-varying system, and let a state space representation be given by

$$\Sigma_\Delta : \quad \begin{cases} \dot{x} \; = \; \mathcal{A}(\Delta) x + \mathcal{B}(\Delta) w \\ z \; = \; \mathcal{C}(\Delta) x + \mathcal{D}(\Delta) w \end{cases} \tag{6.27}$$

If the dependence of the model on the parameters in Δ is rational, the system can be represented in LFT form

$$\Sigma: \quad \left\{ \begin{array}{rcl} \dot{x} &=& Ax + B_p p + B_w w \\ q &=& C_q x + D_{qp} p + D_{qw} w \\ z &=& C_z x + D_{zp} p + D_{zw} w \end{array} \right. \qquad \text{and} \qquad p = \Delta q, \tag{6.28}$$

where Σ is an LTI system and the uncertainty Δ enters via the channel $p \to q$ in a linear fractional manner, as shown in Fig. 6.7.

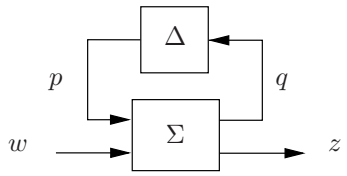


Figure 6.7: LFT representation of uncertain system Σ_Δ

Assuming that the LFT representation is well-posed (Definition 4.2), explicit expressions for the model matrices in (6.27) can be given as

$$\begin{aligned} \mathcal{A}(\Delta) &= A + B_p(I - \Delta D_{qp})^{-1}\Delta C_q \\ \mathcal{B}(\Delta) &= B_w + B_p(I - \Delta D_{qp})^{-1}\Delta D_{qw} \\ \mathcal{C}(\Delta) &= C_z + D_{zp}(I - \Delta D_{qp})^{-1}\Delta C_q \\ \mathcal{D}(\Delta) &= D_{zw} + D_{zp}(I - \Delta D_{qp})^{-1}\Delta D_{qw}. \end{aligned} \tag{6.29}$$

Now applying condition (6.26) to the uncertain system Σ_Δ yields the following result.

Theorem 6.1 *Assume that the interconnection (6.28) is well-posed. The system Σ_Δ is stable and $\|\Sigma_\Delta\|_{\mathcal{L}_2} \leq \gamma$ if there exists X such that $X > 0$ and*

$$\begin{bmatrix} * \end{bmatrix} \left[\begin{array}{cc:cc} 0 & X & 0 & 0 \\ X & 0 & 0 & 0 \\ \hdashline 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & \gamma^{-1} I \end{array}\right] \left[\begin{array}{cc} I & 0 \\ \mathcal{A}(\Delta) & \mathcal{B}(\Delta) \\ \hdashline 0 & I \\ \mathcal{C}(\Delta) & \mathcal{D}(\Delta) \end{array}\right] < 0 \quad \forall \Delta \in \mathbf{\Delta} \tag{6.30}$$

Note that necessity is lost. The condition of this theorem is difficult to evaluate, since it must be checked at infinitely many values of Δ. The following result, proposed in [10], facilitates the evaluation.

Theorem 6.2 *The system Σ_Δ is well-posed, and $X > 0$ satisfies (6.30) if and only if*

there exist $Q = Q^T, R = R^T$ *and* S *such that*

$$\left[\begin{array}{c|c|c} * & & \\ \hline & \left[\begin{array}{cc|cc|cc} 0 & X & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q & S & 0 & 0 \\ 0 & 0 & S^T & R & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & 0 & 0 & \gamma^{-1} I \end{array}\right] & \left[\begin{array}{ccc} I & 0 & 0 \\ A & B_p & B_w \\ \hline 0 & I & 0 \\ C_q & D_{qp} & D_{qw} \\ \hline 0 & 0 & I \\ C_z & D_{zp} & D_{zw} \end{array}\right] \end{array}\right] < 0 \tag{6.31}$$

and

$$\begin{bmatrix} \Delta^T & I \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta \\ I \end{bmatrix} > 0 \quad \forall \Delta \in \mathbf{\Delta} \tag{6.32}$$

For given model matrices of the generalised plant (6.28) and frozen Δ this is an LMI in the matrix variables X, Q, R and S. Note that whereas Theorem 6.1 provides a sufficient condition for stability and performance of the system Σ_Δ, the conditions of Theorem 6.2 are equivalent to the condition of Theorem 6.1; using the latter instead of the former does not incur any additional conservatism. Note also that condition (6.31) involves only the known LTI system (6.28); the uncertainty now appears in condition (6.32) in a form that is easier to handle. This is achieved at the expense of introducing the elements of the *multiplier matrix*

$$W = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}$$

as additional decision variables. We refer to (6.31) as the *nominal condition* and to (6.32) as the *multiplier condition*. The multiplier condition still needs to be checked at infinitely many values of Δ, which can be approximated by checking it on a grid over the admissible parameter range. Now assume that the admissible parameter range $\mathbf{\Delta}$ can be represented as a polytope

$$\mathbf{\Delta} = \ \mathrm{Co}\,\{\Delta_1, \Delta_2, \ldots, \Delta_s\}, \tag{6.33}$$

If the left hand side of (6.32) would depend affinely on Δ, it would be enough to check the multiplier condition in the vertices Δ_i of the polytope. But the dependence is not affine; the matrix

$$\Psi(W, \Delta) = \begin{bmatrix} \Delta^T & I \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta \\ I \end{bmatrix} = \Delta^T Q \Delta + \Delta^T S + S^T \Delta + R$$

is quadratic in Δ. It is nevertheless possible, at the expense of some conservatism, to turn the multiplier condition into a condition that needs to be checked only at the vertices of the polytope $\mathbf{\Delta}$: In Exercise 6.6 it is shown that if the extra condition

$$Q < 0 \tag{6.34}$$

is imposed in addition to (6.32), it follows that

$$\begin{aligned} &\Psi(W, \Delta_i) > 0 \quad \text{and} \quad \Psi(W, \Delta_j) > 0 \\ &\qquad\qquad \Rightarrow \qquad \Psi(W, \alpha\Delta_i + (1-\alpha)\Delta_j) > 0 \quad \forall \alpha : \ 0 \le \alpha \le 1. \end{aligned} \tag{6.35}$$

A consequence of (6.35) is that if the multiplier condition (6.32) is satisfied at the vertices Δ_i of the polytope $\mathbf{\Delta}$ in (6.33), it is satisfied by all $\Delta \in \mathbf{\Delta}$. Note that the extra constraint (6.34) is not required for stability and performance of the formation control loop; adding it to the synthesis conditions is done only to simplify the evaluation of (6.32).

Dual Version of Theorem 6.2

The synthesis of a scheduled state feedback formation control scheme that is presented in the next section requires a linearising change of variables in order to obtain an LMI condition. For this, we will need the dual version of Theorem 6.2, which is also taken from [10].

Theorem 6.3 *The system Σ_Δ is well-posed, and $X > 0$ satisfies (6.30) if and only if there exist $Q = Q^T, R = R^T$ and S such that*

$$\begin{bmatrix} * \end{bmatrix} \left[\begin{array}{cc:cc:cc} 0 & X & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & Q & S & 0 & 0 \\ 0 & 0 & S^T & R & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & -\gamma^2 I & 0 \\ 0 & 0 & 0 & 0 & 0 & I \end{array}\right] \left[\begin{array}{ccc} -A^T & -C_q^T & -C_z^T \\ I & 0 & 0 \\ \hdashline -B_p & -D_{qp}^T & D_{zp}^T \\ 0 & I & 0 \\ \hdashline -B_w^T & -D_{qw}^T & -D_{zw}^T \\ 0 & 0 & I \end{array}\right] > 0 \tag{6.36}$$

and

$$[I \quad -\Delta] \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} I \\ -\Delta^T \end{bmatrix} < 0 \quad \forall \Delta \in \mathbf{\Delta} \tag{6.37}$$

6.4 Scheduled State Feedback Formation Control for LPV Agents

We will now apply the approach presented in the previous section to the design of gain-scheduled formation control for a group of LPV agents. We consider again a formation control loop as shown in Fig. 6.2, where we now assume an individual agent model P to represent an LPV system, and the associated controller K to be an LPV controller scheduled on the same parameter as the agent. Design objectives are stability and a bound on the induced $\mathcal{L}_2$-norm across the performance channel $w \to z$, where we assume identical shaping filters for all agents.

Whereas so far we assumed the Laplacian L to be diagonalisable, we will from now on assume that

$$L = L^T.$$

As mentioned at the beginning of this chapter, this will be the case when the underlying communication graph is undirected and L denotes either the unscaled Laplacian or the normalised Laplacian.

Agent Model

To simplify the presentation, here we will assume that all agent states are available for feedback so that formation control is realised as scheduled state feedback. Moreover, we assume that the agents can be modelled as

$$\dot{x}_i(t) = A\Big(\theta_i(t)\Big)x_i(t) + Bu_i(t), \tag{6.38}$$

where

$$A(\theta_i) = A_0 + \theta_i A_1, \tag{6.39}$$

$\theta_i \in \mathcal{P}_{\theta_0} \subset \mathbb{R}$ is the local scheduling parameter of agent i and $\mathcal{P}_{\theta_0}$ denotes a compact set of admissible values for θ_i, $i = 1, \ldots, N$. Let

$$\theta(t) = [\theta_1(t) \;\; \theta_2(t) \;\; \ldots \;\; \theta_N(t)]^T \tag{6.40}$$

denote the scheduling parameters of individual agents stacked in a column vector. We assume that

$$\theta(t) \in \mathcal{P}_\theta \quad \forall t \geq 0$$

where $\mathcal{P}_\theta$ is a compact set that includes all admissible values of scheduling vectors consistent with $\theta_i \in \mathcal{P}_{\theta_0}$.

Assumptions made here on the agent model to simplify the presentation of the following synthesis results are that B is constant, that agents have only a single scheduling parameter θ_i, and that A is affine in θ_i. Note that this LPV agent model has the same form as the model (6.22) of the actuated disk introduced in Section 6.2; the results presented in this section will be used in Exercise 6.7 to design a gain-scheduled formation control law for a group of wheeled mobile robots that can be represented by such disks.

Scheduled State Feedback

As local formation controllers here we consider LPV state feedback. A general LPV output feedback controller would take the form

$$\begin{array}{rcl} \dot{\zeta}_i(t) & = & A_K(\theta_i)\zeta_i(t) + B_K(\theta_i)e_i(t) \\ u_i(t) & = & C_K(\theta_i)\zeta_i(t) + D_K(\theta_i)e_i(t) \end{array} \tag{6.41}$$

where the local formation error $e_i(t)$ is formed according to

$$e_i(t) = \sum_{j \in \mathcal{J}_i} \Big(y_i(t) - y_j(t)\Big).$$

Recall that when considering LTI formation controllers as in (5.13), we assumed $D_K = 0$ in order to simplify the presentation. In contrast, state feedback can be seen as a special case of (6.41) with $A_K = 0$, $B_K = 0$, $C_K = 0$ and

$$D_K(\theta_i) = F(\theta_i),$$

where F denotes a scheduled state feedback gain matrix. Reflecting the parameter dependence of the agent model, we choose the dependence of the state feedback gain on the scheduling parameter also to be affine. The scheduled state feedback control law is then

$$u_i(t) = F\Big(\theta_i(t)\Big)e_i(t) = \Big(F_0 + \theta_i(t)F_1\Big)e_i(t), \tag{6.42}$$

where

$$e_i(t) = \sum_{j\in\mathcal{J}_i}\Big(x_i(t) - x_j(t)\Big).$$

The formation control loop resulting when this control law is applied to a group of N agents is shown in Fig. 6.8, where we initially assume no formation reference. As before, x, u, e and θ denote the states, inputs, formation errors and scheduling parameters of individual agents stacked together in column vectors.

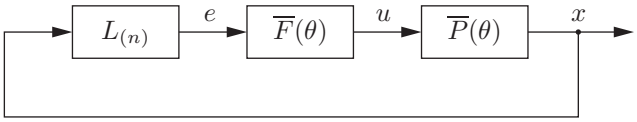


Figure 6.8: Formation state feedback control loop

Define the matrix

$$\Theta(t) = \begin{bmatrix} \theta_1(t) & & \\ & \ddots & \\ & & \theta_N(t) \end{bmatrix} \in \mathbb{R}^{N\times N},$$

where we assume that

$$\Theta(t) \in \mathcal{P}_\Theta \quad \forall t \geq 0$$

and $\mathcal{P}_\Theta$ is the compact set of diagonal $N \times N$ matrices that corresponds to the admissible parameter set $\mathcal{P}_\theta$. Then a state space realisation of the aggregated agent model $\bar{P}(\theta)$ obtained from (6.38) is

$$\dot{x} = \Big(\hat{A}_0 + \Theta_{(n)}\hat{A}_1\Big)x + \hat{B}u, \tag{6.43}$$

where

$$\Theta_{(n)} = \Theta \otimes I_n = \begin{bmatrix} \theta_1 I_n & & \\ & \ddots & \\ & & \theta_N I_n \end{bmatrix}$$

and n denotes the order of the agent models. Substituting the aggregated control law

$$u = \Big(\hat{F}_0 + \Theta_{(m)}\hat{F}_1\Big)L_{(n)}x \tag{6.44}$$

obtained from (6.42), where m denotes the dimension of u_i and $\Theta_{(m)}$ is defined in the same way as $\Theta_{(n)}$, we obtain the closed-loop equation

$$\begin{aligned}\dot{x} &= \left(\hat{A}_0 + \Theta_{(n)}\hat{A}_1\right)x + \hat{B}\hat{F}_0 L_{(n)}x + \hat{B}\Theta_{(m)}\hat{F}_1 L_{(n)}x \\ &= \left(\hat{A}_0 + L \otimes BF_0\right)x + \left(\Theta_{(n)}\hat{A}_1 + \hat{B}\Theta_{(m)}L \otimes F_1\right)x. \end{aligned} \tag{6.45}$$

The first term on the right hand side of (6.45) represents the parameter-independent part of the closed-loop dynamics, and the second term the parameter-dependent one. Note that the mixed product rule has been used to change the order of factors in terms that involve the Laplacian L. To construct an LFT representation of this model, we rewrite the closed-loop formation (6.45) as

$$\dot{x} = \left(\hat{A}_0 + L \otimes BF_0\right)x + [I_{Nn} \;\; \hat{B}] \begin{bmatrix} \Theta_{(n)} & \\ & \Theta_{(m)} \end{bmatrix} \begin{bmatrix} \hat{A}_1 \\ L \otimes F_1 \end{bmatrix} x. \tag{6.46}$$

An LFT representation

$$\begin{aligned} \dot{x} &= \bar{A}x + \bar{B}_p p \\ q &= \bar{C}_q x \end{aligned} \tag{6.47}$$

of the closed-loop formation, with LFT channel $p \to q$ and feedback through

$$p = \bar{\Theta} q,$$

could be obtained from (6.46) by choosing

$$\bar{A} = \hat{A}_0 + L \otimes BF_0, \qquad \bar{B}_p = [I_{Nn} \;\; \hat{B}], \qquad \bar{C}_q = \begin{bmatrix} \hat{A}_1 \\ L \otimes F_1 \end{bmatrix}$$

and

$$\bar{\Theta} = \begin{bmatrix} \Theta_{(n)} & \\ & \Theta_{(m)} \end{bmatrix}.$$

Note that $\bar{A}$ is decomposable w.r.t. L, whereas the matrices $\bar{B}_p$ and $\bar{C}_q$ are not decomposable because of their 1×2 and 2×1 block structure, respectively. For this reason the matrices $\bar{B}_p$ and $\bar{C}_q$ in the LFT representation and also $\bar{\Theta}$ are *not* chosen as above; instead, a permutation is introduced similar to the one discussed in Example 5.2 and in Section 6.1. To illustrate this, assume again $N = 3$ and

$$L = \begin{bmatrix} l_{11} & l_{12} & 0 \\ l_{21} & l_{22} & l_{23} \\ 0 & l_{32} & l_{33} \end{bmatrix}.$$

The term

$$[I_{Nn} \;\; \hat{B}] \begin{bmatrix} \Theta_{(n)} & \\ & \Theta_{(m)} \end{bmatrix} \begin{bmatrix} \hat{A}_1 \\ L \otimes F_1 \end{bmatrix} \tag{6.48}$$

in (6.46) then reads

$$\left[\begin{array}{ccc:ccc} I_n & & & B & & \\ & I_n & & & B & \\ & & I_n & & & B \end{array}\right] \left[\begin{array}{ccc:ccc} \theta_1 I_n & & & & & \\ & \theta_2 I_n & & & & \\ & & \theta_3 I_n & & & \\ \hdashline & & & \theta_1 I_m & & \\ & & & & \theta_2 I_m & \\ & & & & & \theta_3 I_m \end{array}\right] \left[\begin{array}{ccc} A_1 & & \\ & A_1 & \\ & & A_1 \\ \hdashline l_{11}F_1 & l_{12}F_1 & 0 \\ l_{21}F_1 & l_{22}F_1 & l_{23}F_1 \\ 0 & l_{32}F_1 & l_{33}F_1 \end{array}\right].$$

To turn the left and right factors into matrices that are decomposable w.r.t. L, we define a permutation matrix $\Pi \in \mathbb{R}^{N(n+m)\times N(n+m)}$ such that

$$\Pi \left[\begin{array}{c} \hat{A}_1 \\ L \otimes F_1 \end{array}\right] = I_N \otimes \left[\begin{array}{c} A_1 \\ 0 \end{array}\right] + L \otimes \left[\begin{array}{c} 0 \\ F_1 \end{array}\right]$$

and

$$[I_{Nn} \;\; \hat{B}]\Pi = I_N \otimes [I_n \;\; B].$$

Using the fact that if Π is a permutation matrix we have $\Pi^T\Pi = \Pi\,\Pi^T = I$, we can rewrite (6.48) as

$$[I_{Nn} \;\; \hat{B}]\Pi\,\Pi^T \left[\begin{array}{cc} \Theta_{(n)} & \\ & \Theta_{(m)} \end{array}\right] \Pi^T\Pi \left[\begin{array}{c} \hat{A}_1 \\ L \otimes F_1 \end{array}\right] =$$

$$\left[\begin{array}{cc:cc:cc} I_n & B & & & & \\ \hdashline & & I_n & B & & \\ \hdashline & & & & I_n & B \end{array}\right] \left[\begin{array}{cc:cc:cc} \theta_1 I_n & & & & & \\ & \theta_1 I_m & & & & \\ \hdashline & & \theta_2 I_n & & & \\ & & & \theta_2 I_m & & \\ \hdashline & & & & \theta_3 I_n & \\ & & & & & \theta_3 I_m \end{array}\right] \left[\begin{array}{ccc} A_1 & & \\ l_{11}F_1 & l_{12}F_1 & \\ \hdashline & A_1 & \\ l_{21}F_1 & l_{22}F_1 & l_{23}F_1 \\ \hdashline & & A_1 \\ & l_{32}F_1 & l_{33}F_1 \end{array}\right].$$

The left factor is now repeated block diagonal, and the right factor is decomposable w.r.t. L. Thus, the matrices $\bar{B}_p$ and $\bar{C}_q$ in (6.47) are chosen as

$$\bar{B}_p = I_N \otimes [I_n \;\; B], \qquad \bar{C}_q = I_N \otimes \left[\begin{array}{c} A_1 \\ 0 \end{array}\right] + L \otimes \left[\begin{array}{c} 0 \\ F_1 \end{array}\right] \tag{6.49}$$

and the scheduling block $\bar{\Theta}$ as

$$\bar{\Theta} = \left[\begin{array}{ccc} \theta_1 I_{n+m} & & \\ & \ddots & \\ & & \theta_N I_{n+m} \end{array}\right] = \Theta_{(n+m)}, \tag{6.50}$$

while

$$\bar{A} = \hat{A}_0 + L \otimes BF_0 \tag{6.51}$$

is taken as before.

Generalised Plant

In the formation state feedback loop shown in Fig. 6.8 that was considered so far, we assumed for simplicity that the formation reference input r is zero. To formulate a synthesis problem that includes performance specifications, we now introduce a performance channel $w \to z$ into the LFT representation (6.47) and define the generalised plant

$$\begin{aligned} \dot{x} &= \bar{A}x + \bar{B}_p p + \bar{B}_w w \\ q &= \bar{C}_q x \\ z &= \bar{C}_z x + \bar{D}_{zw} w \end{aligned} \tag{6.52}$$

with feedback $p = \bar{\Theta} q$, see Fig. 6.9, subject to

$$\bar{\Theta}(t) \in \mathcal{P}_{\bar{\Theta}} \quad \forall t \geq 0,$$

where $\mathcal{P}_{\bar{\Theta}}$ represents the compact set of admissible diagonal matrices with the structure of $\bar{\Theta}$ that satisfy $\theta \in \mathcal{P}_\theta$. The feedback block $\bar{\Theta}$ and the model matrices $\bar{A}$, $\bar{B}_p$ and $\bar{C}_q$ are defined in (6.50), (6.51) and (6.49), respectively. The matrices $\bar{B}_w$, $\bar{C}_z$ and $\bar{D}_{zw}$ can be used to represent disturbance and performance specifications. We will assume that these specifications are identical for all agents: if they are decoupled the corresponding matrices will be repeated block diagonal, whereas in case of interaction between agents there will be non-zero off-diagonal terms corresponding to non-zero entries in the Laplacian of the communication graph. Note that in either case all model matrices are decomposable w.r.t. the Laplacian of the communication graph, and so is the generalised plant (6.52). A typical situation would be to have repeated block diagonal matrices $\bar{B}_w$ and $\bar{D}_{zw}$, and couplings in

$$\bar{C}_z = \hat{C}_z^d + L \otimes C_z^c$$

that represent weights on the formation errors in the performance channel.

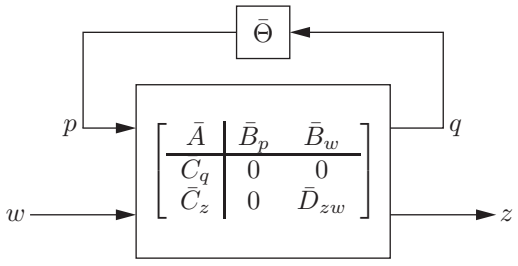


Figure 6.9: Generalised plant model

Scheduled State Feedback Formation Control Problem

We can now formulate the formation control problem we want to solve: find a scheduled state feedback law

$$F(\theta_i) = F_0 + \theta_i F_1 \tag{6.53}$$

for agents $i = 1, 2, \ldots, N$, that stabilises the generalised plant (6.52) and achieves an induced $\mathcal{L}_2$-norm of less than γ. Recall that we assume undirected communication, i.e. L is symmetric; this implies that the performance bound (6.13) on the formation control loop can be made tight.

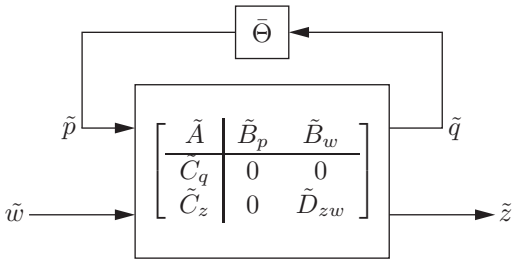


Figure 6.10: Generalised plant in decomposed form, with transformed input and output signals

For symmetric L, there exists a nonsingular $Z \in \mathbb{R}^{N \times N}$ such that

$$Z^{-1} L Z = \Lambda, \qquad \Lambda \text{ diagonal.}$$

Since the generalised plant (6.52) is decomposable, we can use Lemma 6.1 to bring it into the decomposed form of (6.9). This is illustrated in Fig. 6.10, where all model matrices are in block diagonal form and differ only in an affine parameter that represents a different eigenvalue of L for each block. Note however that the signals p, w, q and z have also undergone a transformation. For LTI agents we saw in the previous chapter that this signal transformation has no effect on closed-loop stability and (under undirected communication) no effect on performance either. The situation is however different for LPV agents; to illustrate this in Fig. 6.11 the decomposed generalised plant is shown in terms of the original plant and the block-diagonalising transformation matrices.

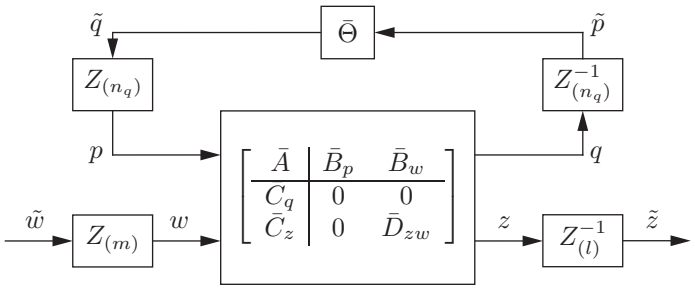


Figure 6.11: Generalised plant with transformation matrices, $n_q = n + m$

Whereas the transformation of signals w and z in the performance channel can be handled as in the LTI case, the transformation of p and q in the LFT channel may actually alter the plant dynamics. At this point we need to distinguish between two alternative assumptions

that can be made about the scheduling policy of the LPV agents, referred to as either *homogeneous* or as *heterogeneous scheduling*, respectively:

- *Homogeneous scheduling* means that the scheduling parameters θ_i take the same values for all agents, i.e.

$$\theta_i(t) = \theta_j(t) = \theta_0(t), \quad 1 \leq i, j \leq N, \ \ \forall t \geq 0$$

- *Heterogeneous scheduling* means that

$$\theta_i(t) \neq \theta_j(t)$$

  is allowed.

In practical terms, homogeneous scheduling might occur when considering e.g. a group of unmanned aerial vehicles (UAV), where altitude and speed are taken as scheduling parameters. Assuming that the UAVs are flying in formation, it is reasonable to assume that altitude and speed take the same values for all agents. In contrast, when considering a group of wheeled mobile robots that are modelled as LPV agents with quasi-LPV models similar to (6.22), the scheduling parameter ω_i that represents the angular velocity of agent i can take different values for each agent.

When scheduling of agents is homogeneous, the LFT feedback block $\bar{\Theta}$ in (6.50) can be written as

$$\bar{\Theta} = \Theta \otimes I_{n+m} = \theta_0 I_{N(n+m)}$$

(recall that here we assume the agents are scheduled on a scalar parameter $\theta_i \in \mathbb{R}$). It then immediately follows that

$$Z_{(n_q)}\bar{\Theta} = \bar{\Theta} Z_{(n_q)} = \theta_0 Z_{(n_q)},$$

so that the effect of the signal transformation in the LFT channel in Fig. 6.11 is eliminated. Thus, when scheduling is homogeneous, the block diagonalising approach that was presented in the previous chapter for LTI agents can be applied to groups of LPV agents as well.

In contrast, when scheduling is heterogeneous, as in the case of the nonholomic agents introduced in Section 6.2, we have in general

$$Z_{(n_q)}\bar{\Theta} \neq \bar{\Theta} Z_{(n_q)}.$$

So in the practically important case of heterogeneous scheduling of agents, the block diagonalising technique of the previous chapter cannot be applied, and we need to take a different approach to the synthesis of a scheduled formation control scheme.

Applying the Full Block S-Procedure

Since the reduction of the synthesis problem to the size of a single agent cannot be achieved by decomposing the overall multi-agent system into modal subsystems as in (6.12), we will apply the conditions of Theorem 6.3 first to the overall closed-loop formation of N agents, represented as generalised plant (6.52), and then decompose these conditions into N small conditions, each the size of a single agent.

The following result is obtained by replacing in (6.36) and (6.37) the uncertain system (6.28) by the closed-loop formation (6.52) and the uncertainty block Δ by $\bar{\Theta}$.

Theorem 6.4 *The closed-loop formation (6.52) is well-posed, stable and has an $\mathcal{L}_2$-gain less than or equal to γ if there exist $\bar{X} > 0$, $\bar{Q} = \bar{Q}^T$, $\bar{R} = \bar{R}^T$ and $\bar{S}$ such that*

$$\begin{bmatrix} * \end{bmatrix} \left[\begin{array}{cc:cc:cc} 0 & \bar{X} & 0 & 0 & 0 & 0 \\ \bar{X} & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & \bar{Q} & \bar{S} & 0 & 0 \\ 0 & 0 & \bar{S}^T & \bar{R} & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & -\gamma^{-2} I_{Nn_w} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{Nn_z} \end{array}\right] \left[\begin{array}{ccc} -\bar{A}^T & -\bar{C}_q^T & -\bar{C}_z^T \\ I_{Nn} & 0 & 0 \\ \hdashline -\bar{B}_p^T & 0 & 0 \\ 0 & I_{Nn_q} & 0 \\ \hdashline -\bar{B}_w^T & 0 & -\bar{D}_{zw}^T \\ 0 & 0 & I_{Nn_z} \end{array}\right] > 0 \qquad (6.54)$$

and

$$\begin{bmatrix} I_{N(n+m)} & -\bar{\Theta} \end{bmatrix} \begin{bmatrix} \bar{Q} & \bar{S} \\ \bar{S}^T & \bar{R} \end{bmatrix} \begin{bmatrix} I_{N(n+m)} \\ -\bar{\Theta}^T \end{bmatrix} < 0 \quad \forall \bar{\Theta} \in \bar{\mathbf{\Theta}}. \qquad (6.55)$$

Note that the model matrices $(\bar{A}, \bar{B}_p, \bar{B}_w, \bar{C}_q, \bar{C}_z)$ which appear in the nominal condition can be of huge size when the number of agents N is large, and so are the decision matrix variables; we have

$$\bar{X} \in \mathbb{R}^{Nn \times Nn}, \qquad \bar{Q}, \bar{R}, \bar{S} \in \mathbb{R}^{N(n+m) \times N(n+m)}.$$

This would even for moderately large N render impractical any attempt to turn the above conditions into LMIs that can be solved for the controller variables. In order to break the problem up into smaller subproblems, rewrite the nominal condition (6.54) as

$$\bar{M}^T \bar{\Psi} \bar{M} > 0,$$

and consider the transformation of the right factor $\bar{M}$

$$\underbrace{\left[\begin{array}{ccc} -\tilde{A}^T & -\tilde{C}_q^T & -\tilde{C}_z^T \\ I & 0 & 0 \\ \hdashline -\tilde{B}_p^T & 0 & 0 \\ 0 & I & 0 \\ \hdashline -\tilde{B}_w^T & 0 & -\tilde{D}_{zw}^T \\ 0 & 0 & I \end{array}\right]}_{\tilde{M}} =$$

$$\underbrace{\left[\begin{array}{ccc:c:c} Z_{(n)}^{-1} & & & & \\ & Z_{(n)}^{-1} & & & \\ \hdashline & & \ddots & & \\ \hdashline & & & \ddots & \end{array}\right]}_{Z_1^{-1}} \underbrace{\left[\begin{array}{ccc} -\bar{A}^T & -\bar{C}_q^T & -\bar{C}_z^T \\ I & 0 & 0 \\ \hdashline -\bar{B}_p^T & 0 & 0 \\ 0 & I & 0 \\ \hdashline -\bar{B}_w^T & 0 & -\bar{D}_{zw}^T \\ 0 & 0 & I \end{array}\right]}_{\bar{M}} \underbrace{\left[\begin{array}{ccc} Z_{(n)} & & \\ & Z_{(n_q)} & \\ & & Z_{(n_z)} \end{array}\right]}_{Z_2}.$$

All model matrices of the generalised plant that appear in $\bar{M}$ are decomposable w.r.t. L, where again we assume that

$$Z^{-1}LZ = \Lambda,$$

with Λ diagonal. As a consequence, all matrix blocks in

$$\tilde{M} = Z_1^{-1}\bar{M}Z_2$$

are in decomposed form. In fact, we have by assumption

$$\tilde{B}_w = \bar{B}_w = \hat{B}_w, \qquad \tilde{B}_p = \bar{B}_p = \hat{B}_p, \qquad \tilde{D}_{zw} = \bar{D}_{zw} = \hat{D}_{zw},$$

because these matrices are not involved in communication between agents. Moreover, the matrices

$$\tilde{A} = \hat{A}_0 + \Lambda \otimes B_u F_0, \qquad \tilde{C}_q = I_N \otimes \left[\begin{array}{c} A_1 \\ 0 \end{array}\right] + \Lambda \otimes \left[\begin{array}{c} 0 \\ F_1 \end{array}\right], \qquad \tilde{C}_z = \hat{C}_z^d + \Lambda \otimes C_z^c$$

are in decomposed form.

This alone would not help to break the synthesis problem up into small subproblems; what is required is a decomposition of the two matrix inequality conditions (6.54) and (6.55). This can be achieved by imposing a structure on the matrix decision variables: we assume that

$$\bar{P} = \hat{P}, \quad \bar{Q} = \hat{Q}, \quad \bar{R} = \hat{R}, \quad \bar{S} = \hat{S}, \tag{6.56}$$

where as before the $\hat{}$ notation means the Kronecker product of I_N with the matrix under the hat, i.e. all decision matrix variables are constrained to have repeated block diagonal

structure, such as

$$\hat{P} = \begin{bmatrix} P & & \\ & \ddots & \\ & & P \end{bmatrix} \qquad \text{etc.}$$

Now replacing the matrix factor $\bar{M}$ in (6.54) by the transformed factor $\tilde{M}$ yields

$$\underbrace{Z_2^T \bar{M}^T Z_1^{-T}}_{\tilde{M}^T} \underbrace{\left[\begin{array}{cc:cc:cc} 0 & \hat{X} & 0 & 0 & 0 & 0 \\ \hat{X} & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & \hat{Q} & \hat{Z} & 0 & 0 \\ 0 & 0 & \hat{Z}^T & \hat{R} & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & -\gamma^{-2} I & 0 \\ 0 & 0 & 0 & 0 & 0 & I \end{array}\right]}_{\bar{\Psi}} \underbrace{Z_1^{-1} \bar{M} Z_2}_{\tilde{M}} > 0$$

or

$$\tilde{M}^T \bar{\Psi} \tilde{M} = Z_2^T \bar{M}^T \left(Z_1^{-1}\right)^T \bar{\Psi} Z_1^{-1} \bar{M} Z_2 > 0. \tag{6.57}$$

In general this condition is not equivalent to (6.54). Since we assumed L to be symmetric, however, we can choose Z to be orthogonal so that we have

$$Z^{-1} = Z^T \qquad \Rightarrow \qquad \left(Z_1^{-1}\right)^T = Z_1.$$

Moreover, it follows from Lemma 5.2 and the structural constraints (6.56) that $\bar{\Psi} Z_1^{-1} = Z_1^{-1} \bar{\Psi}$; thus (6.57) is equivalent to

$$\tilde{M}^T \bar{\Psi} \tilde{M} = Z_2^T \bar{M}^T Z_1 Z_1^{-1} \bar{\Psi} \bar{M} Z_2 = Z_2^T \bar{M}^T \bar{\Psi} \bar{M} Z_2 > 0.$$

This however is simply a congruence transformation of (6.54). We have thus shown that matrices $\hat{X} > 0$, $\hat{Q} = \hat{Q}^T$, $\hat{R} = \hat{R}^T$ and $\hat{S}$ satisfy the conditions of Theorem 6.4 if and only if they satisfy

$$\begin{bmatrix} * \end{bmatrix} \left[\begin{array}{cc:cc:cc} 0 & \hat{X} & 0 & 0 & 0 & 0 \\ \hat{X} & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & \hat{Q} & \hat{S} & 0 & 0 \\ 0 & 0 & \hat{S}^T & \hat{R} & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & -\gamma^{-2} I_{Nn_w} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{Nn_z} \end{array}\right] \left[\begin{array}{ccc} -\tilde{A}^T & -\tilde{C}_q^T & -\tilde{C}_z^T \\ I_{Nn} & 0 & 0 \\ \hdashline -\tilde{B}_p^T & 0 & 0 \\ 0 & I_{N(n+m)} & 0 \\ \hdashline -\tilde{B}_w^T & 0 & -\tilde{D}_{zw}^T \\ 0 & 0 & I_{Nn_z} \end{array}\right] > 0 \tag{6.58}$$

and

$$\begin{bmatrix} I_{N(n+m)} & -\bar{\Theta} \end{bmatrix} \begin{bmatrix} \hat{Q} & \hat{S} \\ \hat{S}^T & \hat{R} \end{bmatrix} \begin{bmatrix} I_{N(n+m)} \\ -\bar{\Theta}^T \end{bmatrix} < 0 \quad \forall \bar{\Theta} \in \bar{\mathbf{\Theta}} \tag{6.59}$$

Since all matrix blocks appearing in the "big" conditions (6.59) and (6.58) are in decomposed form, these conditions decompose into "small" conditions

$$\left[\begin{array}{c|cc|cc} * & & & & \end{array}\right] \left[\begin{array}{cc|cc|cc} 0 & X & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q & S & 0 & 0 \\ 0 & 0 & S^T & R & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -\gamma^{-2} I_{n_w} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{n_z} \end{array}\right] \left[\begin{array}{ccc} -A_i^T & -C_{qi}^T & -C_{zi}^T \\ I_n & 0 & 0 \\ \hline -B_p^T & 0 & 0 \\ 0 & I_{n+m} & 0 \\ \hline -B_w^T & 0 & -D_{zw}^T \\ 0 & 0 & I_{n_z} \end{array}\right] > 0, \qquad i = 1, 2, \ldots, N \tag{6.60}$$

and

$$[I_{n+m} \quad -\theta_0 I_{n+m}] \left[\begin{array}{cc} Q & S \\ S^T & R \end{array}\right] \left[\begin{array}{c} I_{n+m} \\ -\theta_0 I_{n+m} \end{array}\right] < 0 \quad \forall \theta_0 \in \mathcal{P}_{\theta_0} \subset \mathbb{R}, \tag{6.61}$$

where

$$A_i = A_0 + \lambda_i B_u F_0, \qquad C_{qi} = \left[\begin{array}{c} A_1 \\ \lambda_i F_1 \end{array}\right], \qquad C_{zi} = C_z^d + \lambda_i C_z^c$$

and $\lambda_i,\ i = 1, \ldots, N$, denote the eigenvalues of L. Since the matrices B_p, B_w and D_{zw} are not involved in interaction between agents, they are not scaled by eigenvalues of L.

We can summarise the above discussion in the following result.

Theorem 6.5 *The closed-loop formation (6.52) is well-posed, stable and has an $\mathcal{L}_2$-gain less than or equal to γ if there exist $X > 0$, $Q = Q^T$, $R = R^T$ and S that satisfy (6.60) for all $\lambda_i(L)$, $i = 1, \ldots, N$, and (6.61) for all $\theta_0 \in \mathcal{P}_{\theta_0}$.*

Note that the conditions (6.60) and (6.61) have the size of a single agent, and are linear in the variables. These analysis conditions can be turned into LMI synthesis conditions for finding the gains F_0 and F_1 for a scheduled state feedback gain

$$F(\theta_i) = F_0 + \theta_i F_1$$

by defining new variables

$$Y_0 = F_0 X \qquad \text{and} \qquad Y_1 = F_1 X.$$

Exercises — Chapter 6

Problem 6.1 *(Formation Stability for Arbitrary Topologies)*

Learning Goals

- Understand the step from the global formation control problem to a single agent problem

Task Description According to [20], a formation of N identical agents $P(s) = \left[\begin{array}{c|c} A_P & B_P \\ \hline C_P & 0 \end{array}\right]$ equipped with local controllers $K(s)$ is stable if and only if $K(s)$ simultaneously stabilizes the systems

$$\begin{aligned} \dot{x}_i &= A_P x_i + B_P u_i \\ z_i &= \lambda_i C_P x_i \end{aligned} \tag{6.62}$$

for all eigenvalues λ_i of the Laplacian of the communication topology.

a) Assume the Laplacian L not to be diagonalizable. Derive the transformed closed loop state equation in terms of an upper triangular matrix $U = T^{-1}LT$ at the place of L using the Schur transformation T. Assume $r = 0$.

b) From the Schur-transformed state equation, prove that the above stability criterion holds.

Problem 6.2 *(Stability of a Multi-Vehicle Formation)*

Learning Goals

- Illustrate the stability conditions for a formation control problem

Task Description Consider a multi-agent system (MAS) consisting of 5 identical vehicles moving along a line. The dynamics can be described by the model

$$P(s): \quad A_P = \begin{bmatrix} 0 & 1 \\ 0 & -b/m \end{bmatrix}, \qquad B_P = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}, \qquad C_P = \begin{bmatrix} 1 & 0 \end{bmatrix}, \qquad D_P = 0 \tag{6.63}$$

with the parameters $m = 10$ and $b = 0.5$. The agents are connected by the communication topology shown in Fig. 6.12. To achieve a formation movement, negative feedback control is used with a decentralized formation controller

$$K(s) = 20\frac{s + 0.05}{s + 2.4}. \tag{6.64}$$

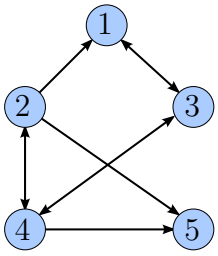


Figure 6.12: Topology of the multi-agent system

a) Using the given information about a single agent, show that the given multi-agent system is stable.

b) Generate the closed loop system of the whole formation using MATLAB, compute the poles of this system and confirm the stability.

c) i) Draw the Nyquist plot of the forward loop $P(s)K(s)$.

ii) In which area are the inverses of the non-zero eigenvalues of the Laplacian allowed to be located for the closed loop MAS to be stable?
Hint: Apply the Nyquist stability criterion to $\lambda_i P(s)K(s)$.

iii) All eigenvalues λ_i of the normalized Laplacian lie within a unit disk shifted by 1 (called *Perron disk*). The boundary of this disk is described by $\lambda = 1 - e^{j\alpha}$. Use this property to determine the region where the inverses of λ_i can be located. Is there any topology which drives the given MAS unstable?

d) Simulate the response of the closed-loop multi-agent system to step references $r = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}^T \sigma(t)$ for zero initial values.

Problem 6.3 *(Design of a Formation Controller)*

Learning Goals

- Apply the knowledge to a realistic problem

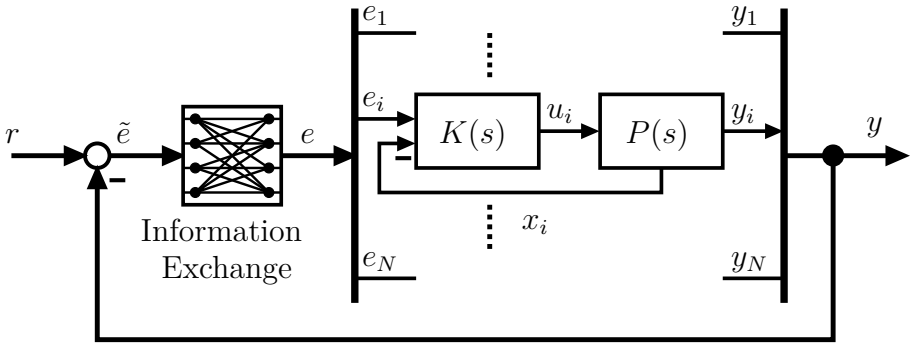


Figure 6.13: Formation control scheme

Task Description As an example of agents in a multi-agent system, quad-rotor helicopters (or short quadrocopters) are used for research at the Institute of Control Systems. A simple linearized model of a quadrocopter is presented in [29], see Appendix G for details.

Now consider the problem to achieve a formation flight of 4 such quadrocopters. We can assume that each quadrocopter is equipped with the necessary sensors to measure all the states of the model from [29]. As the quadrocopter is an unstable plant, additional local state feedback should be used here, as shown in Fig. 6.13. Assume that each agent transmits its position data via wireless communication links.

a) Formulate a stability condition for the quadrocopter formation based on the model of a single quadrocopter and a local controller.

b) Use the property that all eigenvalues of the adjacency matrix lie in the unit disk and modify the quadrocopter model to consider the topology as LFT form uncertainty.

c) i) Draw a generalized plant to design a robust H_∞ formation controller guaranteeing stability for arbitrary communication topologies. For performance measurement choose suitable shaping filters $W_S(s)$ and $W_K(s)$ to shape the formation control error e and the control input u.

   ii) Use the generalized plant in MATLAB to synthesize the H_∞ formation controller.

d) Use Simulink to simulate the response of the quadrocopter swarm to a step reference input in x direction $r_x = \begin{bmatrix} 2 & 4 & 6 & 8 \end{bmatrix}^T \sigma(t)$, assuming zero initial values. You may try different communication topologies.

Problem 6.4 *(Information Flow Filter for Formation Control)*

Learning Goals

- Transfer the knowledge to a different formation control architecture

Task Description Consider the multi-agent system shown in Fig. 6.14, which is proposed in [30] and is a simplification of a formation control scheme from [20]. The block $\hat{R}(s)$ consists of a feedback loop around a system $\hat{F}(s)$, which is called *information flow filter*.

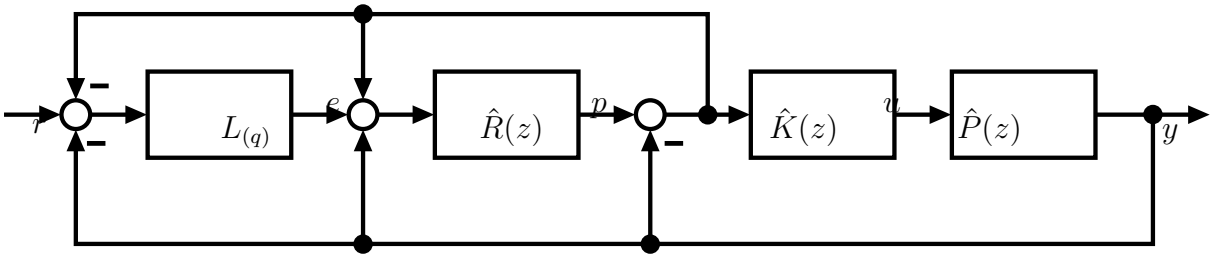


Figure 6.14: Formation control scheme with information flow filter

a) i) Using $\hat{R}(s) = (I_N q + \hat{F}(s))^{-1}\hat{F}(s)$, derive the transfer function from r to p.

ii) Derive the transfer function from p to y.

b) Show that the transfer functions obtained in a) correspond to the control architecture shown in Fig. 6.15. How can the signal p be interpreted physically?

Hint: Use the general formula for the transfer function of a feedback loop

c) Consider the communication topology as unknown. Transform the information flow loop of Fig. 6.15 into a local LFT system considering the unknown topology as uncertainty. Assume the signal communicated by each agent to be scalar (i. e. $q = 1$).

d) Derive a generalized plant for synthesis of an information flow filter which stabilizes the loop for every topology.

e) Use Matlab to synthesize such an information flow filter and test the information flow loop in simulation.

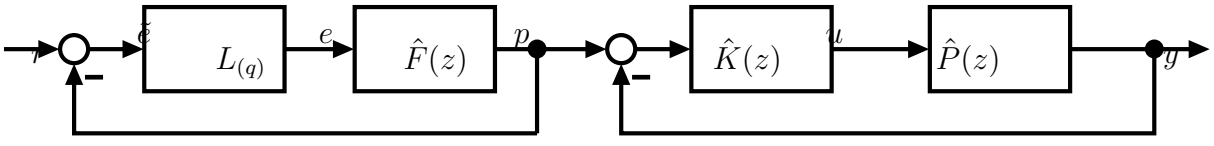


Figure 6.15: Separated formation control scheme

Problem 6.5 *(Disturbance Rejection in Formation Control)*

Learning Goals

- Understand the behavior of the information flow filter scheme and its differences to the cooperative architecture

Task Description Recall the multi-agent system from Problem 6.2. Consider the goal to achieve and maintain the formation $y_d = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}^T$ and assume that agent 1 is affected by an output disturbance $d_y = 2\sigma(t - t_d), t_d = 10s$.

a) Use the formation controller given in Problem 6.2 and simulate the system response to the disturbance acting on agent 1.

b) Simulate the system response to the same scenario using the information flow filter scheme shown in Fig. 6.15 with

$$F(z) = \frac{s + 8.5}{10^{-4}s^2 + s + 0.01} \quad \text{and} \quad K(z) = 0.15\frac{1 + 60s}{1 + 0.6s}. \tag{6.65}$$

Compare the result to that of (a). What difference do you observe?

c) Explain the difference between the two disturbance responses by means of the control schemes.

d) Modify the multi-agent system by adding an agent with fixed position, which transmits data to agent 1 and 2, but does not receive any. Repeat the simulation with the modified system. What difference do you observe? Which function does the fixed-position agent have?

Problem 6.6 *(Scheduled State Feedback Synthesis)*

Learning Goals

- Apply the combined knowledge about LPV-control and decomposable system to synthesize a decomposable scheduled state feedback controller.

Task Description In the following a system of five quasi-LPV agents is considered. Each agent, whose dynamic given by

$$\begin{bmatrix}\dot{x}^k\\ \dot{q}^k\end{bmatrix} = \left[\begin{array}{cc:cc:cc} -1 & -1 & 1 & 0 & 1 & 0\\ 0 & -1 & 0 & 1 & 0 & 1\\ \hdashline 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix}x^k\\ q^k\\ u^k\end{bmatrix} = \left[\begin{array}{c:c:c} A & B_p & B_u\\ \hdashline C_q & 0 & 0 \end{array}\right] \begin{bmatrix}x^k\\ p^k\\ u^k\end{bmatrix}, \quad \begin{aligned} p^k &= \begin{bmatrix}\theta_1^k & \\ & \theta_2^k\end{bmatrix} q^k = \Theta^k q^k,\\ \theta_1^k &= \sin x_1^k,\\ \theta_2^k &= \sin x_2^k, \end{aligned}$$

is a two-dimensional stable systems with poles at -1, whose locations are perturbed by LPV parameters. Since these depends on the system states, we call it a quasi LPV system. The vector $x^k = [x_1,\, x_2]^T$ is the state vector of the k-th agent.

a) How does the state matrices $\hat{A}$, $\hat{B}_p$, $\hat{B}_u$, $\hat{C}_q$ and Θ look like if the state equation for a group of five agents is

$$\begin{bmatrix}\dot{x}\\ \dot{q}\end{bmatrix} = \left[\begin{array}{c:c:c} \hat{A} & \hat{B}_p & \hat{B}_u\\ \hdashline \hat{C}_q & 0 & 0 \end{array}\right] \begin{bmatrix}x\\ p\\ u\end{bmatrix}, \quad p = \Theta q$$

with $x = \left[x^{1T}, \ldots, x^{5T}\right]^T$ and the other signals are composed respectively.

b) Each single agent is to controlled via scheduled state feedback $u^k = F^k e^k$ with the gain $F^k = F^0 + \Theta^k F^1$. Calculate the closed-loop system

$$\begin{bmatrix}\dot{x}\\ \dot{\bar{q}}\end{bmatrix} = \left[\begin{array}{c:c:c} \bar{A} & \bar{B}_p & \bar{B}_u\\ \hdashline \bar{C}_q & \bar{D}_{qq} & \bar{D}_{qw} \end{array}\right] \begin{bmatrix}x\\ \bar{p}\\ w\end{bmatrix}, \quad \bar{p} = \bar{\Theta}\bar{q}$$

of the feedback loop shown in Figure 6.16.

c) In the following weighting filters W_s and W_k are to include to shape the sensitivity and the controller sensitivity.

   i) Include the weighting filters in the feedback loop in Figure 6.16, when the performance outputs are $z_1^k = W_s I_2 e_k$ and $z_2^k = W_k I_2 u_k$.

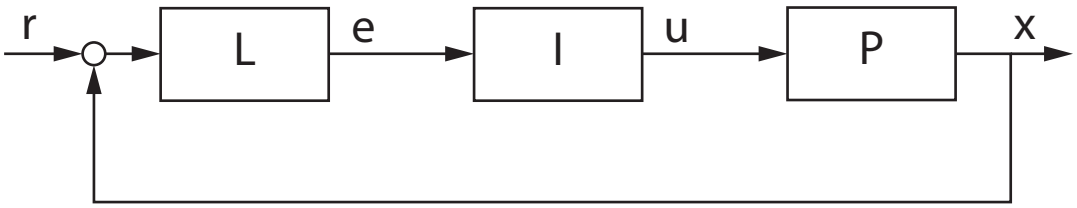


Figure 6.16: Closed-loop of distributed state feedback control

ii) The filters are chosen as

$$W_s = \frac{s^2 + 2s + 1}{s^2 + 0.2 + 0.01} I_2 \quad \text{and} \quad W_k = \frac{100s + 100}{s + 1000} I_2 \,.$$

Why does that choice makes sense if good tracking should be achieved?

iii) Calculate the state space equations of the filters

$$\begin{bmatrix} \dot{x}_s^k \\ \hline z_1 \end{bmatrix} = \left[\begin{array}{c|c} A_s & B_s \\ \hline C_s & D_s \end{array}\right] \begin{bmatrix} x_s^k \\ \hline e^k \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \dot{x}_k^k \\ \hline z_2 \end{bmatrix} = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right] \begin{bmatrix} x_k^k \\ \hline u^k \end{bmatrix}$$

and derive the state equations of the generalized plant for the group of five agents as

$$\begin{bmatrix} \dot{x} \\ \dot{x}_s \\ \dot{x}_k \\ \hline \bar{q} \\ \hline z_1 \\ z_2 \end{bmatrix} = \left[\begin{array}{c|c|c} \bar{\mathbf{A}} & \bar{\mathbf{B}}_p & \bar{\mathbf{B}}_{\mathbf{w}} \\ \hline \bar{\mathbf{C}}_q & \bar{\mathbf{D}}_{qq} & \bar{\mathbf{D}}_{qw} \\ \hline \bar{\mathbf{C}}_z & \bar{\mathbf{D}}_{zq} & \bar{\mathbf{D}}_{zw} \end{array}\right] \begin{bmatrix} x \\ x_s \\ x_k \\ \hline \bar{p} \\ \hline w \end{bmatrix}, \quad \bar{p} = \bar{\Theta}\bar{q} \,.$$

d) Apply the full-block S-procedure to get conditions for stability and an induced $\mathcal{L}_2$-gain less than γ of the closed loop generalized system .

e) The conditions derived in the previous task depends on the number of subsystems and can get thus very large for a large number of subsystems. Lets consider the main condition first, which is of the form $M^T \Psi M < 0$.

i) Repeat from the lecture, what transformation $Z_1^{-1} M Z_2 = \tilde{M}$ built by $(S \otimes I)$'s is necessary, such $\tilde{M}$ contains only block diagonal matrices?

ii) Under what conditions are the transformation matrices Z_1 and Z_2 orthogonal?

iii) We want to block diagonalize both outer factor and write

$$Z_2^{-1} M^T Z_1 \Psi Z_1^{-1} M Z_2 < 0 \,.$$

Under the conditions derived in the previous exercise we have

$$Z_2^T M^T Z_1 \Psi Z_1^T M Z_2 < 0 \,.$$

What structural constraints need to be applied on the Lyapunov matrix and the multiplier matrices such that this is equivalent to $Z_2^T M^T \Psi M Z_2 < 0$? Why does that implies $M^T \Psi M < 0$ and vice versa.

iv) Thus $M^T \Psi M < 0$ is equivalent to $\tilde{M}^T \Psi \tilde{M} < 0$. What is the advantage of that form?

f) With the assumptions of the previous task we have a stability and performance condition for the group of agents, which is only of the dimension of one agent. But nevertheless this condition is not linear, because there are products of the controller matrices F_0 and F_1 and the Lyapunov matrix P and the controller matrices F_0 and F_1 and the multiplier matrices Q, R and S. To get rid of the first products, apply a congruence transformation as known from [31] for state feedback. The resulting condition should only depend on the new variables $Y = P^{-1}$, $D_0 = F_0 Y$ and $D_1 = F_1 Y$ and the unchanged multiplier matrices.

g) There are still products of the multiplier matrices with Y and D_1 left. Use the Dualization Lemma C.1 to eliminate those.

h) The multiplier condition

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta \\ I \end{bmatrix} > 0, \ \forall \Delta \in \boldsymbol{\Delta}$$

has to be solved infinitely many times, because it should be valid for all $\delta \in P_\delta$. It is desired, that solving it in all vertices of the convex hull of P_δ, guarantees that the multiplier condition is fulfilled for all $\delta \in P_\delta$. Why is this implied by the condition $Q < 0$?

i) For simplifications assume Δ to be the scalar δ and

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \begin{bmatrix} q & s \\ s & r \end{bmatrix} \begin{bmatrix} \delta \\ 1 \end{bmatrix} > 0, \ \forall \delta \in \boldsymbol{\delta}\,. \tag{6.66}$$

Why does $q < 0$ and

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \begin{bmatrix} q & s \\ s & r \end{bmatrix} \begin{bmatrix} \delta \\ 1 \end{bmatrix} > 0, \ \forall \delta \in \{\delta_{\min}, \delta_{\max}\}$$

implies (6.66)?

ii) What is the dual of the multiplier condition and what is the respective condition to $Q < 0$ for the dual?

Problem 6.7 *(LPV State Feedback Formation Control for Non-Holonomic Agents)*

Learning Goals

- Apply LPV synthesis tools to a formation control problem for a group of LPV mobile vehicles under time-varying undirected topologies.

Task Description A set of LPV non-holonomic agents must achieve a desired aligned geometric formation under time-varying undirected communication topologies (Figure 6.17). For this purpose a distributed LPV state feedback design is considered. To proceed with this exercise, the student must be familiar with Exercise 6.6.

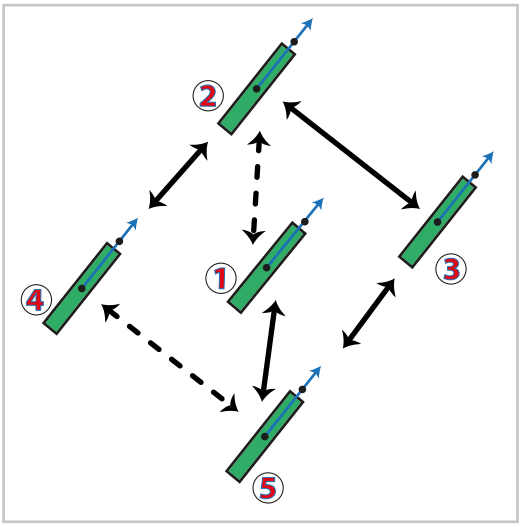


Figure 6.17: Set of mobile agents under possibly time-varying undirected communication.

a) Consider the rolling disk detailed in Section 6.2. The same disk is used here, but now a handle is included, as shown in Fig. 6.18.

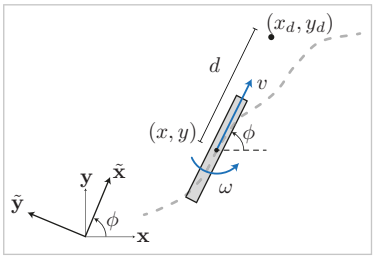


Figure 6.18: Rolling disk with handle.

Obtain the non-linear equations that capture the dynamics of the point (x_d, y_d).

Apply a change of coordinates by using the transformation T (Eq. 6.17) and find a state space model.

A set of N agents is considered, individual agents are labeled with k. Based on the transformed system, derive an LPV-LFT representation (Eq. 6.67) where, $\Theta^k = \theta^k I_2$, $\theta^k \in [-1, 1]$ and $\theta^k = \omega^k/5$. All states are available for feedback.

$$\begin{bmatrix} \dot{x}^k \\ p^k \\ y^k \end{bmatrix} = \begin{bmatrix} A & B_\Theta & B_u \\ C_\Theta & D_{\Theta \mathrm{i}} & D_{\Theta u} \\ C_y & D_{y\Theta} & D_{yu} \end{bmatrix} \begin{bmatrix} x^k \\ q^k \\ u^k \end{bmatrix} \qquad q^k = \Theta^k p^k \tag{6.67}$$

What is the benefit of the state transformation T?
What is the benefit of employing a handle? For this exercise $d = 0.3$ will be used.

Parameters are collected accordingly to:

$$\begin{aligned} \Theta &= \mathrm{diag}(\Theta^1 \dots \Theta^N) \\ \theta &= [\theta^1 \dots \theta^N]^{\mathrm{T}} \end{aligned}$$

b) Consider the closed loop system depicted in Figure 6.19, which contains N subsystems. Each subsystem contains an LPV plant model $P^k(\theta^k)$, a transformation matrix T and an associated LPV state feedback $F^k(\theta^k)$. All are collected according to:

$$\begin{aligned} \bar{F}(\theta) &= I_N \otimes F_0 + \Theta(I_N \otimes F_1) \\ \bar{P}(\theta) &= \mathrm{diag}(P^1(\theta^0) \dots P^N(\theta^N)) \\ \hat{T} &= \mathrm{diag}(T^1 \dots T^N) \\ \hat{W}_S &= I_N \otimes W_S \\ \hat{W}_{KS} &= I_N \otimes W_{KS} \end{aligned}$$

One can see that all systems are *equal*, in the sense that individual agents differ only from each other on how are they scheduled (heterogeneously scheduled).

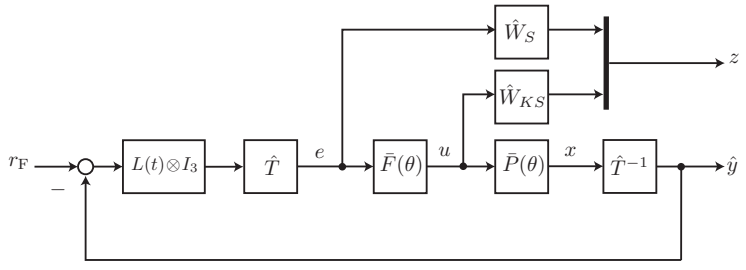


Figure 6.19: Multi-Agent closed loop LPV system.

Notice that in the closed-loop system in Figure 6.19 the states are transformed back to the coordinate system $(\mathbf{x}, \mathbf{y})$, then they are used for feedback purposes. After they are shared through L, they are transformed back to the coordinate system $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$.

Why is this necessary? How can one guarantee that performance and stability will be valid without explicitly introduce $\hat{T}$ and $\hat{T}^{-1}$ into the generalized plant?

One can also notice that all systems are decoupled in the sense that they only share their state information through L. How can a physical communication be avoided and keeping the same interconnection among agents? *Hint: Think of different types of sensors that can be used.*

c) An S/KS framework is used to weight the performance error and the control effort. In Exercise 6.6, an LPV-LFT representation of the same scheme proposed here is obtained. There, the matrix inequalities in Eq. 6.70 were proposed to synthesize a state feedback controller as the one proposed here.

$$[*]^{\mathrm{T}} \left[\begin{array}{c:cc:cc} 0 & I & & & & \\ I & 0 & & & & \\ \hdashline & & \tilde{Q} & \tilde{S} & & \\ & & \tilde{S}^T & \tilde{R} & & \\ \hdashline & & & & -\gamma^2 I & 0 \\ & & & & 0 & I \end{array}\right] \left[\begin{array}{ccc} -\mathbf{A}(Y_1,Y_2,Y_3,D_0)^{\mathrm{T}} & -\mathbf{C}_q(Y_1,D_1)^{\mathrm{T}} & -\mathbf{C}_z(Y_2,Y_3)^{\mathrm{T}} \\ I & 0 & 0 \\ \hdashline -\mathbf{B}_p^{\mathrm{T}} & 0 & -\mathbf{D}_{zp}^{\mathrm{T}} \\ 0 & I & 0 \\ \hdashline -\mathbf{B}_w(Y_1,D_0)^{\mathrm{T}} & -\mathbf{D}_{qw}(D_1)^{\mathrm{T}} & -\mathbf{D}_{zw}(Y_1,D_0)^{\mathrm{T}} \\ 0 & 0 & I \end{array}\right] > 0, \quad \forall \lambda \in \lambda \tag{6.70a}$$

$$[*]^{\mathrm{T}} \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} \begin{bmatrix} I \\ -\Delta^{\mathrm{T}} \end{bmatrix} < 0, \quad \forall \Delta \in \mathbf{\Delta} \tag{6.70b}$$

They are not linear in γ and a substitution of variables cannot be suggested since γ is the decision variable to optimize.

Transform the matrix inequality in Eq. 6.70a to an LMI by using a Schur complement argument.

In Exercise 6.6 it was suggested to impose $\tilde{R} > 0$ to ensure convexity of LMI 6.70b. What other impositions can be made to guarantee that LMI 6.70b is satisfied without evaluation? Why is this conservative?

d) Choose reasonable weights W_S and W_{KS} to penalize the formation error and the control effort. Include your data into the Matlab file `Synthesis\_LPV\_MAS\_SF` to synthesize the state feedback gains F_0 and F_1.

e) Use the Matlab file `Main.m` to simulate your results for a group of 5 agents. Retune your weighting filters to ensure that the control input is between the following ranges:

$$v_i \in [-15, 15], \quad \omega_i \in [-5, 5], \quad \forall i \in \{1 \dots 5\}$$

In the file you can change the number and type of formation references, as well as the communication topology. Both of them randomly change every 30 and 5 seconds according to Figure 6.20.

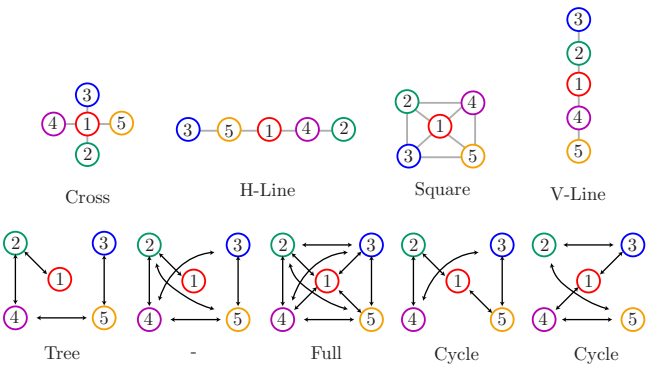


Figure 6.20: Different geometric references and interconnection topologies.

Part III

DISTRIBUTED CONTROL OF SPATIALLY INTERCONNECTED SYSTEMS

Chapter 7

Distributed Control of Spatially Interconnected Systems

7.1 Distributed and Interconnected Systems

Following the study of multiagent systems, we will in this chapter turn to a different type of networked control systems. In contrast to the groups of mobile agents considered in the previous two chapters, which communicate with their neighbours according to a given communication graph, here we consider large distributed systems with dynamics governed by partial differential equations. We assume that an array of sensor-actuator pairs - e.g. micro-electrical mechanical systems (MEMS array) - is attached to the system and that a distributed control scheme is to be used. Features of such a distributed scheme are that there is no centralised feedback controller that collects information from all individual sensors, but that there are local feedback loops that exchange information with neighbouring units. A spatially distributed array of sensor-actuator pairs induces a spatial discretisation of the system, and we may view the overall system as a network of individual subsystems with a regular grid-like interaction structure, referred to as a *spatially interconnected system*. As in the treatment of multiagent systems, when designing and analysing distributed control schemes for such systems it is desirable to solve analysis and synthesis problems with a complexity that is independent of the number of subsystems - ideally with the complexity of a single subsystem. This is indeed possible; in [32] a framework is proposed that provides analysis and synthesis results for spatially interconnected systems with guaranteed $\mathcal{L}_2$-norm, where the latter is extended to distributed systems.

In this chapter we consider two application examples in order to illustrate the approach: we will study the problem of controlling the temperature distribution along a thin metal rod, and in an exercise problem the same approach is applied to the problem of controlling the vibration of an actuated beam.

Example: Controlling the Temperature along a Thin Metal Rod

Consider a long and thin metal rod that has evenly spaced pairs of heating elements and temperature sensors attached to it. Figure 7.1 shows a part of such a rod with three sensor actuator pairs displayed. We will consider the rod as a dynamic system with an input and an output signal that each depend on *two independent variables*: time and space. The output $y(t, \sigma)$ represents the temperature of the rod at time t and location σ, and $u(t, \sigma)$ stands for an external heat flow into the rod at t and σ. Note that whereas in this example (and for simplicity in this whole chapter) we consider systems with only a single spatial dimension σ, it is straightforward to extend the approach presented here to distributed systems with more than one spatial dimension.

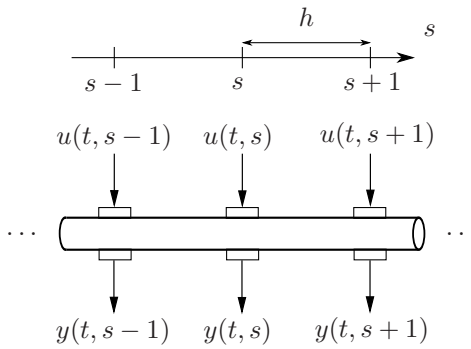


Figure 7.1: Thin metal rod with sensors and actuators

While we will leave time to be represented by the continuous-valued variable t, it will be convenient to spatially discretise the model to account for the evenly spaced sensor-actuator pairs. Assuming that the distance between neighbouring sensor-actuator pairs is h (the spatial sampling interval), we introduce an integer-valued space variable $s = \sigma/h$, *i.e.* $s \in \{\ldots -2, -1, 0, 1, 2, \ldots\}$, as shown in Figure 7.1. The rod - equipped with actuators and sensors - can now be seen as a dynamic system with *spatio-temporal* input and output signals, and a dynamic model can be derived from spatially discretising the partial differential equation governing the distribution of heat - the heat equation. This will be done in Section 7.1.

Multidimensional Signals and Systems

So far the dynamic systems we studied were lumped-parameter systems where input and output signals depend only on the single independent variable time. Assuming m input channels and l output channels, these signals can be seen as maps $u : \mathbb{R}_+ \to \mathbb{R}^m$ and $y : \mathbb{R}_+ \to \mathbb{R}^l$, where $\mathbb{R}_+$ denotes the set of nonnegative real numbers, reflecting the assumption that all signals $x(t)$ satisfy the assumption $x(t) = 0$ if $t < 0$.

Figure 7.2 shows a lumped SISO system that produces a one-dimensional impulse response

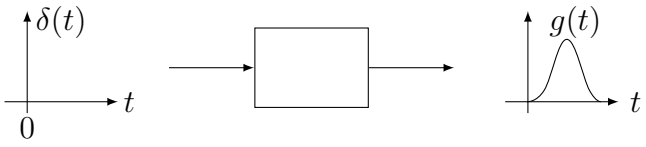


Figure 7.2: Lumped-parameter system

$g(t)$ when excited by a unit impulse $\delta(t)$. In contrast, the input and output signals of a spatially interconnected system system (with a single spatial dimension such as the heat rod in Figure 7.1) can be seen as maps $u : \mathbb{R}_+ \times \mathbb{Z} \to \mathbb{R}^m$ and $y : \mathbb{R}_+ \times \mathbb{Z} \to \mathbb{R}^l$, see Figure 7.3, where $\mathbb{Z}$ denotes the set of integers.

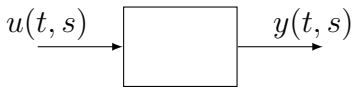


Figure 7.3: Spatially interconnected system

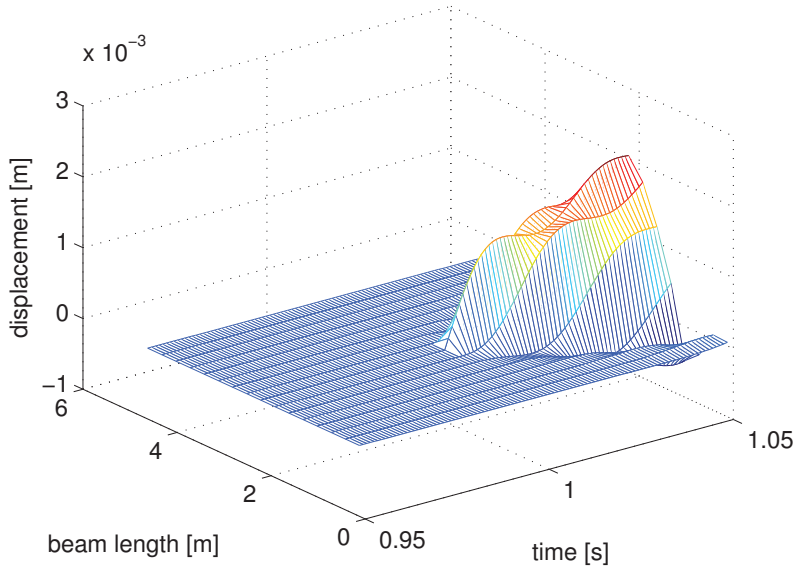


Figure 7.4: 2D-impulse input

The effect of a spatiotemporal delta impulse $\delta(t, s)$ applied at the system input at time $t = 0$ and location $s = 0$, will propagate both in time (only in the future direction since we assume the system to be causal with respect to time) and in space (in two directions, indicating that the system is non-causal with respect to space). An experimentally measured response of the actuated beam introduced in Exercise 7.2 to a spatiotemporal impulse $\delta(t - 1, s - 4)$ is shown in Figure 7.4.

Linear Time- and Space-Invariant Interconnected Systems

In [32] a modeling framework is proposed for spatially interconnected systems that lends itself to LMI-based analysis and synthesis of distributed control schemes. We will illustrate this framework with the heat rod model. Figure 7.5 shows again the heat rod and indicates a spatial discretisation induced by the sensor-actuator array: the rod is partioned into identical segments such that each segment contains a sensor-actuator pair. The idea is now to model the overall system as a network of interconnected subsystems, as shown in Figure 7.6.

Each subsystem has an input and an output signal, which represent the local actuator and sensor signal, respectively. And since the subsystems interact with their neighbours, we need also *interaction signals* between the blocks, which in the case of the heat rod represent heat flowing along the rod from one segment into another. The following notation is used: interaction signals flowing into a block are labeled v, while signals flowing out of a block are labeled w. The direction from left to right is taken as positive, and superscripts indicate the direction in which a signal is flowing. For example, the interaction signal coming from the subsystem at location s at time t flowing to the right is labeled $w^+(t,s)$. Since this signal is also an input from the left to the subsystem located at $s+1$, we have $w^+(t,s) = v^+(t,s+1)$.

Important assumptions made in [32] are: the distributed dynamics of the considered spatially interconnected systems are linear and time invariant; moreover, they are also assumed to be *space invariant*, *i.e.* the dynamic properties are constant over space. Such systems are referred to as *linear time and space-invariant)* LTSI systems. Here we assume that the heat rod used as illustrative example is an LTSI system. When the sensor-actuator pairs are evenly distributed, this implies that all subsystems in Figure 7.5 are identical.

Distributed Control

The approach proposed in [32] is based on the idea of representing the dynamics of the overall distributed, spatially discretised system by the state space model of a single subsystem, and to reduce analysis and controller synthesis problems to the complexity of such a subsystem. As for feedback control, there are different possible control structures that vary in terms of information flow and complexity. Figure 7.7 shows three different

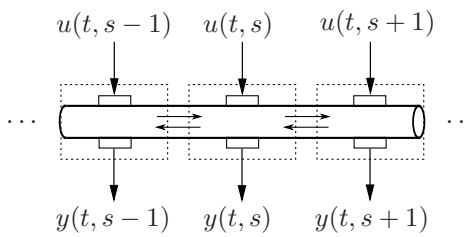


Figure 7.5: Heat rod partitioned into segments

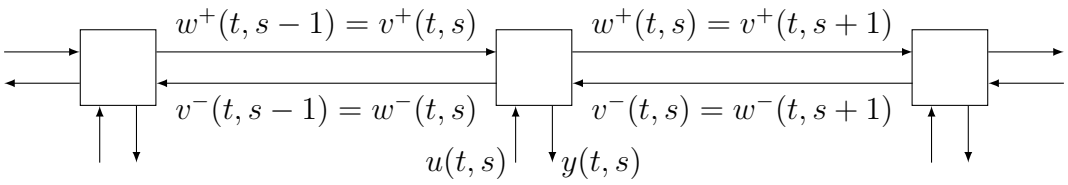


Figure 7.6: Model of heat rod as network of interconnected subsytems

structures. In Figure 7.7.a a *centralised control* structure is indicated where the distributed system is treated as a lumped-parameter system with a huge number of input and output channels, controlled by a MIMO controller. Figure 7.7.b shows a *decentralised control* structure, where control input is generated by isolated local feedback loops. A control structure referred to as *distributed control* is shown in Figure 7.7.c, where local feedback controllers exchange information with neighbouring controllers, using the same interaction structure as the plant. Clearly for a large number of sensor-actuator pairs the centralised control scheme will be too complex both in synthesis and in implementation for practical applications. The scheme with lowest complexity is the decentralised one; however the information available to local controllers is rather restricted. The distributed scheme combines a reasonably low complexity with the benefit of information exchange between neighbouring units; it is this scheme that will be considered in the following chapter.

State Space Model

Before turning to closed-loop distributed control, we introduce a state space model of a single subsystem of the open-loop system shown in Figure 7.6. Each subsystem can be seen as a LTI system with state vector $x(t) \in \mathbb{R}^n$. We also define the interaction signal vectors

$$v(t,s) = \begin{bmatrix} v^+(t,s) \\ v^-(t,s) \end{bmatrix}, \qquad w(t,s) = \begin{bmatrix} w^+(t,s) \\ w^-(t,s) \end{bmatrix} \tag{7.1}$$

where $v(t,s)$ represents interaction input and $w(t,s)$ interaction output at location s. The dynamic interaction of signals entering the subsystem in Figure 7.6 at location s can then be represented as

$$\begin{bmatrix} \dot{x}(t,s) \\ \Delta_S v(t,s) \end{bmatrix} = \begin{bmatrix} A_{TT} & A_{TS} \\ A_{ST} & A_{SS} \end{bmatrix} \begin{bmatrix} x(t,s) \\ v(t,s) \end{bmatrix} + \begin{bmatrix} B_T \\ B_S \end{bmatrix} u(t,s) \tag{7.2}$$

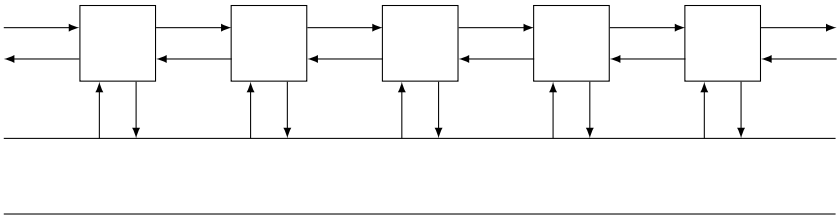
$$y(t,s) = \begin{bmatrix} C_T & C_S \end{bmatrix} \begin{bmatrix} x(t,s) \\ v(t,s) \end{bmatrix} + Du(t,s). \tag{7.3}$$

Next we introduce the spatial shift operator S, defined by

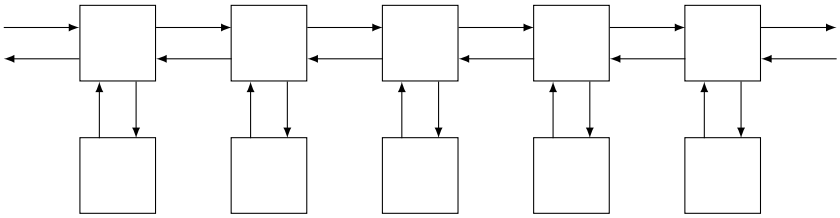
$$Su(t,s) = u(t,s+1)$$

and its inverse by

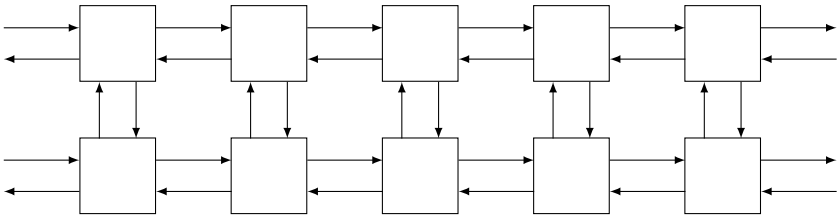
$$S^{-1}u(t,s) = u(t,s-1).$$



(a) Centralised control structure



(b) Decentralised control structure



(c) Distributed control structure

Figure 7.7: Control structures

Assuming that all interaction signals have dimension m, i.e. $v^+, v^-, w^+, w^- \in \mathbb{R}^m$, we introduce the block diagonal shift operator

$$\Delta_S = \begin{bmatrix} SI_m & 0 \\ 0 & S^{-1}I_m \end{bmatrix}$$

and note that from (7.1) and the interconnection structure in Figure 7.6 we have

$$\Delta_S \, v(t,s) = w(t,s).$$

Moreover, defining the block diagonal operator

$$\Delta = \begin{bmatrix} \frac{d}{dt}I_n & 0 \\ 0 & \Delta_S \end{bmatrix}$$

and the vector

$$\xi(t,s) = \begin{bmatrix} x(t,s) \\ v(t,s) \end{bmatrix},$$

we can rewrite (7.2) as

$$\begin{aligned} \Delta\,\xi(t,s) =& A\,\xi(t,s) + B\,u(t,s) \\ y(t,s) =& C\,\xi(t,s) + D\,u(t,s), \end{aligned} \tag{7.4}$$

where

$$A = \begin{bmatrix} A_{TT} & A_{TS} \\ A_{ST} & A_{SS} \end{bmatrix}, \quad B = \begin{bmatrix} B_T \\ B_S \end{bmatrix}, \quad C = [C_T \;\; C_S].$$

The representation (7.4) can be seen as a hybrid continuous-discrete state space model with state vector ξ, where the *temporal state variables* - the elements of x - are governed by a vector differential equation, and the *spatial state variables* - the elements of v - are governed by a vector difference equation, and both differential operator and shift operator are combined in the hybrid operator Δ.

Illustrative Example: Heat Rod

To illustrate the state space model (7.4), we return the to heat rod shown in Figure 7.1 that is equipped with evenly spaced sensor-actuator pairs. Assuming the rod is sufficiently thin and long, the dynamics of heat conduction are represented by the one-dimensional heat equation

$$\frac{\partial y(t,\sigma)}{\partial t} = \frac{\partial^2 y(t,\sigma)}{\partial \sigma^2} + u(t,\sigma) \tag{7.5}$$

where for simplicity the diffusivity constant is taken to be 1. To account for the spatial discretisation that is induced by the distribution of the sensor-actuator pairs, we replace the second partial derivative w.r.t. space by a central difference approximation: at location $\sigma = sh$ we have, neglecting higher order terms

$$\left.\frac{\partial^2 y(t,\sigma)}{\partial \sigma^2}\right|_{\sigma=sh} = \frac{y(t,sh+h) - 2y(t,sh) + y(t,sh-h)}{h^2}).$$

Substituting this in (7.5) and taking the temperature as the single (temporal) state variable in a first order state space model, we obtain the state equation

$$\dot{x}(t,sh) = \frac{1}{h^2}x(t,sh+h) - \frac{2}{h^2}x(t,sh) + \frac{1}{h^2}x(t,sh-h) + u(t,sh) \tag{7.6}$$

together with the output equation

$$y(t,sh) = x(t,sh).$$

If we now see this difference-differential equation in the context of the block diagram in Figure 7.6, we can interpret the interaction signals as the temperature values, i.e.

$$v^+(t,sh) = x(t,sh-h), \quad v^-(t,sh) = x(t,s+h), \quad w^+(t,sh) = w^-(t,sh) = x(t,sh).$$

The state space model (7.4) then takes the form

$$\begin{bmatrix} \dot{x}(t,sh) \\ w^+(t,sh) \\ w^-(t,sh) \end{bmatrix} = \begin{bmatrix} -\frac{2}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(t,sh) \\ v^+(t,sh) \\ v^-(t,sh) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t,sh) \tag{7.7}$$

(7.8)

$$y(t,sh) = [1 \;\; 0 \;\; 0] \begin{bmatrix} x(t,sh) \\ v^+(t,sh) \\ v^-(t,sh) \end{bmatrix}. \tag{7.9}$$

Infinite-Dimensional State Space Model

In order to formulate conditions for stability and performance of spatially interconnected systems, we will now construct a condensed version of the hybrid state space model (7.4) (hybrid in the sense that it includes both temporal state variables - the elements of x, and spatial state variables - the elements of v). Recall that the state equation in (7.4) is a compact representation of (7.2). We can solve the second row of (7.2), *i.e.*

$$\Delta_S\, v(t,s) = A_{ST}\, x(t,s) + A_{SS}\, v(t,s) + B_S\, u(t,s)$$

for

$$v(t,s) = (\Delta_S - A_{SS})^{-1} A_{ST}\, x(t,s) + (\Delta_S - A_{SS})^{-1} B_S\, u(t,s).$$

By substituting this in the first row of (7.2) and in (7.3), we eliminate the spatial state variables v from state and output equation and obtain the model

$$\begin{aligned} \dot{x}(t) &= \mathcal{A}(\Delta_S)\, x(t) + \mathcal{B}(\Delta_S)\, u(t) \\ y(t) &= \mathcal{C}(\Delta_S)\, x(t) + \mathcal{D}(\Delta_S)\, u(t), \end{aligned} \tag{7.10}$$

where

$$\left[\begin{array}{cc} \mathcal{A}(\Delta_S) & \mathcal{B}(\Delta_S) \\ \mathcal{C}(\Delta_S) & \mathcal{D}(\Delta_S) \end{array}\right] = \left[\begin{array}{cc} A_{TT} & B_T \\ C_T & D \end{array}\right] + \left[\begin{array}{c} A_{TS} \\ C_S \end{array}\right] (\Delta_S - A_{SS})^{-1} \left[A_{ST}\ B_S\right]. \tag{7.11}$$

Note that the model matrices $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ are now matrix-valued operators, *i.e.* they are rational functions of the spatial shift operator Δ_S. Eliminating the explicit representation of the spatial state variables v leads to a model where each entry of the temporal state vector $x(t)$ now represents the complete spatial distribution of its values at time t. The notation $x(t)$ in (7.10) thus stands for an infinite sequence

$$x(t) = \{\ldots, x(t,s-1),\ x(t,s),\ x(t,s+1), \ldots\} \tag{7.12}$$

For the case where x is scalar, *i.e.* $x(t,s) \in \mathbb{R}$ for frozen t and s, this is illustrated in Fig. 7.1, where the spatial distribution of a signal $x(t)$ at time instant t_0 is shown.

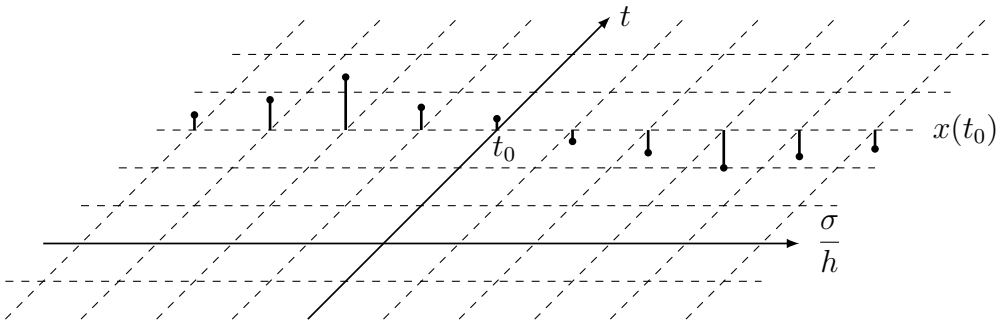


Figure 7.8: Infinite-dimensional temporal signal x at frozen time $t = t_0$

If $x(t,s) \in \mathbb{R}^n$, then the sequence is vector valued, *e.g.* if $n = 2$ we have

$$x(t) = \left\{ \dots, \begin{bmatrix} x_1(t,s-1) \\ x_2(t,s-1) \end{bmatrix}, \begin{bmatrix} x_1(t,s) \\ x_2(t,s) \end{bmatrix}, \begin{bmatrix} x_1(t,s+1) \\ x_2(t,s+1) \end{bmatrix}, \dots \right\}.$$

The spatial dynamics that are shown explicitly in (7.2) and (7.4) are now represented by the operators $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$, which are acting on the infinite-dimensional signals $x(t)$, $y(t)$ and $u(t)$. To illustrate this, consider as a simple example $n = 2$ and

$$\mathcal{A}(\Delta_S) = \begin{bmatrix} S & 0 \\ 0 & S^{-1} \end{bmatrix}.$$

In this case the action of $\mathcal{A}$ on the infinite-dimensional signal $x(t)$ is given by

$$\mathcal{A}(\Delta_S)\, x(t) = \left\{ \dots, \begin{bmatrix} x_1(t,s) \\ x_2(t,s-2) \end{bmatrix}, \begin{bmatrix} x_1(t,s+1) \\ x_2(t,s-1) \end{bmatrix}, \begin{bmatrix} x_1(t,s+2) \\ x_2(t,s) \end{bmatrix}, \dots \right\}.$$

Example: Heat Rod

To illustrate the infinite-dimensional state space model (7.10), we consider again the heat rod example. Assuming for simplicity $h = 1$, we obtain from (7.7) and (7.11)

$$\begin{aligned} \begin{bmatrix} \mathcal{A}(\Delta_S) & \mathcal{B}(\Delta_S) \\ \mathcal{C}(\Delta_S) & \mathcal{D}(\Delta_S) \end{bmatrix} &= \begin{bmatrix} -2 & 1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} S^{-1} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} S + S^{-1} - 2 & 1 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

Thus we have $\mathcal{A}(\Delta_S) = S + S^{-1} - 2$, $\mathcal{B}(\Delta_S) = \mathcal{C}(\Delta_S) = 1$ and $\mathcal{D}(\Delta_S) = 0$. Substituting this in the state space model (7.10) yields

$$\dot{x}(t) = (S + S^{-1} - 2)x(t) + u(t), \qquad y(t) = x(t),$$

which is exactly the difference-differential equation (7.6) together with its output equation.

Signal Norms

The infinite-dimensional state space model (7.10) will be used to formulate analysis conditions on spatially-interconnected systems, that can then be transformed into synthesis conditions for distributed controllers. Before we outline the derivation of such conditions, we first extend the signal and system norms that we used when studying lumped-parameter LTI systems, to spatially-interconnected systems.

Again referring to the infinite sequence (7.12) that represents the spatial distribution of a signal $x(t)$ at a frozen time instant, illustrated in Fig. 7.1 for $t = t_0$, we define a norm on such sequences as

$$\|x(t)\|_{\ell_2} = \left(\sum_{l=-\infty}^{\infty} \|x(t,l)\|^2 \right)^{\frac{1}{2}}, \tag{7.13}$$

where the norm of the terms in the sum on the right hand side is the vector-2-norm. Note that t is a variable here; (7.13) is the norm of the infinite-dimensional signal $x(t)$ at a fixed time t. Obviously this norm is bounded only if the spatial sequence is square summable. The space of all sequences with finite ℓ_2-norm, taking values in $\mathbb{R}^n$, is referred to as the signal space ℓ_2^n: we define it as

$$\ell_2^n = \{x(t) \mid \|x(t)\|_{\ell_2} < \infty\}, \tag{7.14}$$

where again t is a variable frozen at a given value. The signal space ℓ_2 can be seen as a discrete equivalent of the signal space $\mathcal{L}_2$ that was introduced in Chapter 2 when studying the performance of LPV systems. But since the ℓ_2-norm is only a measure of the 'size' of a spatio-temporal signal at a given time t, we still need a measure of the 'size' of that signal when taken as a function of space *and* time. This is provided by the $\mathcal{L}_2$-norm of a spatio-temporal signal, defined as

$$\|x(t,s)\|_{\mathcal{L}2} = \left(\int_{t=0}^{\infty} \|x(t)\|_{\ell_2}^2 \, dt\right)^{\frac{1}{2}}. \tag{7.15}$$

Here we use for the norm on spatio-temporal signals the same symbol $\mathcal{L}_2$ that was used for the one-dimensional (temporal) signals in Chapter 2. The difference is that whereas in the latter case the integration is over the vector-2 norm of finite-dimensional signal vectors, here integration is over the ℓ_2 norms of infinite sequences. The space of all spatio-temporal signals with finite $\mathcal{L}_2$ norm, taking values in $\mathbb{R}^n$, is then

$$\mathcal{L}_2^n = \{x(t,s) \mid \|x(t,s)\|_{\mathcal{L}_2} < \infty\}. \tag{7.16}$$

The Induced $\mathcal{L}_2$-Norm

The induced $\mathcal{L}_2$-norm plays a significant role when analysing and synthesising control loops. For lumped-parameter LTI systems the induced $\mathcal{L}_2$-norm reduces to the H_∞-norm, which can be used to represent performance and / or robustness specifications. Convex analysis and synthesis conditions in the form of LMIs can be formulated with the help of the bounded-real lemma. In Chapter 2 this approach was extended to lumped-parameter LPV systems. Here we will now outline how it can be extended as well to spatially interconnected systems modelled as in (7.10), based on the definitions in (7.15) and (7.13). Thus, let $\mathcal{M}$ denote the system

$$\begin{aligned} \dot{x}(t) &= \mathcal{A}(\Delta_S)\, x(t) + \mathcal{B}(\Delta_S)\, d(t) \\ z(t) &= \mathcal{C}(\Delta_S)\, x(t) + \mathcal{D}(\Delta_S)\, d(t), \end{aligned} \tag{7.17}$$

where the input-output channel $d \to z$ can be thought of as a performance channel. We define the induced $\mathcal{L}_2$-norm of this system as

$$\|\mathcal{M}\|_{\mathcal{L}_2} = \sup_{0 \neq d \in \mathcal{L}_2} \frac{\|z(t,s)\|_{\mathcal{L}_2}}{\|d(t,s)\|_{\mathcal{L}_2}}. \tag{7.18}$$

Analysis Results

In the following we will briefly outline one of the main results presented in [32], an analysis result for spatially interconnected systems proposed in Theorem 1 of that paper. Three properties of such systems need to be established:

- Well-posedness
- Stability
- Performance

The first property, *well-posedness*, was already encountered in Chapter 4 in the context of LFT representations. It means, roughly speaking, that a feedback loop is defined in a physically meaningful way, such that for given external signals the loop signals exist and are unique. This property is required for spatially interconnected systems to make sure that loops resulting from the interconnection of subsystems are physically meaningful. It turns out that a spatially interconnected system is well-posed if $(\Delta_S - A_{SS})$ is invertible. More precisely: since this expression represents an operator that maps ℓ_2 into ℓ_2, the condition for well-posedness is that $(\Delta_S - A_{SS})$ is invertible on ℓ_2.

Stability of spatially interconnected systems can be defined in terms of the $\mathcal{L}_2$-norm of the state vector $x(t)$. We say that a system is *exponentially stable* if for any non-zero initial state the norm of the state vector along the free response is converging to zero at an exponential rate.

Performance is also defined in terms of the $\mathcal{L}_2$-norm: we say that a system $\mathcal{M}$ has performance level γ if $\|\mathcal{M}\|_{\mathcal{L}_2} \leq \gamma$. Moreover, we say $\mathcal{M}$ is *contractive* if $\|\mathcal{M}\|_{\mathcal{L}_2} \leq 1$.

Lyapunov Stability

For lumped-parameter LTI and LPV systems, Lyapunov stability was a key for formulating stability conditions that can be expressed in the form of LMIs. In order to see how this can be extended to spatially interconnected systems, we will briefly review the Lyapunov stability condition for LTI systems and express it in terms of the inner product on the space $\mathbb{R}^n$.

The system

$$\dot{x}(t) = Ax(t) \tag{7.19}$$

is stable if there exists a Lyapunov function $V(x)$ that has the two properties that (i) $V(x) > 0$ for all $x \neq 0$ and (ii) that $\frac{d}{dt}V\Big(x(t)\Big) < 0$ along all possible trajectories $x(t)$ except when $x = 0$. A suitable Lyapunov function candidate is $V(x) = x^T P x$ where P is a positive definite matrix. We note that (i) is satisfied because by assumption P is positive definite, and that (ii) is satisfied if

$$\frac{d}{dt}V\Big(x(t)\Big) = \dot{x}^T(t)Px(t) + x^T(t)P\dot{x}(t) < 0$$

along all possible trajectories $x(t)$ except when $x \leq 0$, or

$$x^T A^T P x + x T P A x < 0 \qquad \forall x \neq 0,$$

which leads to the well-known Lyapunov inequality $A^T P + PA < 0$, which is an LMI in P. In fact it turns out that for LTI systems the existence of a positive definite matrix P that satisfies the Lyapunov inequality is a necessary and sufficient condition for stability.

Inner Product Spaces

To extend this stability result to spatially interconnected systems, we will first reformulate the above result in terms of the *inner product* on the space $\mathbb{R}^n$. Recall that for a given real vector space $\mathcal{V}$, an inner product is a map

$$\langle \cdot, \cdot \rangle : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$$

that satisfies the following three conditions for all $x, y, z \in \mathcal{V}$ and all scalars $a \in \mathbb{R}$:

- Symmetry
$$\langle x, y \rangle = \langle y, x \rangle$$
- Linearity
$$\langle ax, y \rangle = a\langle x, y \rangle \qquad \text{and} \qquad \langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$$
- Positive-definiteness
$$\langle x, x \rangle \geq 0 \qquad \text{and} \qquad \langle x, x \rangle = 0 \;\; \Leftrightarrow \;\; x = 0.$$

A vector space on which an inner product can be defined is called an *inner product space*. The Euclidean space $\mathbb{R}^n$ is an inner product space; its inner product is the dot product $\langle x, y \rangle = x^T y$. Note that when t is frozen the state vector $x(t)$ of an LTI system is in $\mathbb{R}^n$, and the system matrix in (7.19) can be seen as a linear operator $A : \mathbb{R}^n \to \mathbb{R}^n$.

We can now rewrite the stability condition for the LTI system (7.19) in terms of the inner product on $\mathbb{R}^n$: the system is stable if (and only if) there exists $P > 0$ that satisfies

$$\langle x, Px \rangle > 0 \qquad \text{and} \qquad \langle Ax, Px \rangle + \langle Px, Ax \rangle < 0 \;\; \forall 0 \neq x \in \mathbb{R}^n. \tag{7.20}$$

The Signal Space ℓ_2^n as Inner Product Space

The extension of the above Lyapunov approach to spatially interconnected systems with infinite-dimensional state space models of the form (7.10) or (7.17) is achieved by exploiting the fact that the infinite-dimensional real vector space ℓ_2 is an inner product space as well. First we note that whereas for an LTI system with time frozen we have $x(t) \in \mathbb{R}^n$, for a (stable) spatially interconnected system (7.17) with time frozen we have $x(t) \in \ell_2^n$. For

$x(t,s), y(t,s) \in \ell_2^n$ (for clarity we here also display the dependence on the space variable s), the inner product on ℓ_2^n is denoted by $\langle x(t,s), y(t,s)\rangle_{\ell_2}$ and defined as

$$\langle x(t,s), y(t,s)\rangle_{\ell_2} = \sum_{s=-\infty}^{\infty} x^T(t,s)y(t,s). \tag{7.21}$$

Now we are ready to address the problem of selecting a Lyapunov function candidate that can establish stability of the unforced spatially interconnected system

$$\dot{x}(t) = \mathcal{A}(\Delta_S)\, x(t).$$

Inspired by (7.20) we define

$$V\Big(x(t)\Big) = \langle x(t), P_T x(t)\rangle_{\ell_2}$$

where $P_T \in \mathbb{R}^{n\times n}$ is a symmetric, positive definite matrix. One can check that the negative definiteness condition on the Lyapunov derivative is then given by

$$\langle \mathcal{A}x, P_T x\rangle_{\ell_2} + \langle P_T x, \mathcal{A}x\rangle_{\ell_2} < 0 \;\; \forall 0 \neq x \in \ell_2^n. \tag{7.22}$$

Note that $x \in \ell_2^n$ stands here for a spatial sequence, and both P_T and $\mathcal{A}(\Delta_S)$ here represent operators mapping ℓ_2^n into ℓ_2^n. For example we have

$$P_T x(t) = \{\ldots, P_T x(t,s-1), P_T x(t,s), P_T x(t,s+1), \ldots\}.$$

An Analysis Result in LMI Form for Spatially Interconnected Systems

We conclude this chapter by outlining and briefly discussing an analysis result for spatially interconnected systems that has been proposed in [32]. Consider a system $\mathcal{M} = (\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ that is represented in the form of (7.17).

Theorem 7.1 *The system $\mathcal{M} = (\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ is well-posed, stable and contractive, if there exist real matrices $P_T = P_T^T > 0$ and $P_S = P_S^T$ that satisfy*

$$J(P_T, P_S) = M_1^T \Psi_1(P_T, P_S) M_1 + M_2^T \Psi_2(P_T, P_S) M_2 < 0, \tag{7.23}$$

where the block matrices M_1, M_2, Ψ_1 and Ψ_2 are defined below.

In order to define the block matrices in (7.23), we first rewrite the state space model (7.2), (7.3) for partitioned interaction signals

$$v = \begin{bmatrix} v^+ \\ v^- \end{bmatrix}, \qquad w = \begin{bmatrix} w^+ \\ w^- \end{bmatrix}$$

and partition the model matrices accordingly as

$$\left[\begin{array}{c} \dot{x} \\ \hdashline w^+ \\ w^- \end{array}\right] = \left[\begin{array}{c:cc} A_{TT} & A_{TS+} & A_{TS-} \\ \hdashline A_{ST+} & A_{SS++} & A_{SS+-} \\ A_{ST-} & A_{SS-+} & A_{SS--} \end{array}\right] \left[\begin{array}{c} x \\ \hdashline v^+ \\ v^- \end{array}\right] + \left[\begin{array}{c} B_T \\ \hdashline B_{S+} \\ B_{S-} \end{array}\right] u.$$

Now we define the matrices

$$A_{SS}^{+} = \begin{bmatrix} A_{SS++} & A_{SS+-} \\ 0 & I \end{bmatrix}, \quad A_{SS}^{-} = \begin{bmatrix} I & 0 \\ A_{SS-+} & A_{SS--} \end{bmatrix}$$

$$A_{ST}^{+} = \begin{bmatrix} A_{ST+} \\ 0 \end{bmatrix}, \quad A_{ST}^{-} = \begin{bmatrix} 0 \\ A_{ST-} \end{bmatrix}$$

$$B_{S}^{+} = \begin{bmatrix} B_{S+} \\ 0 \end{bmatrix}, \quad B_{S}^{-} = \begin{bmatrix} 0 \\ B_{S-} \end{bmatrix}$$

$$A_{TS}^{+} = [A_{TS+} \;\; 0], \quad A_{TS}^{-} = [0 \;\; A_{TS-}].$$

The matrices M_1, M_2, Ψ_1 and Ψ_2 can then be expressed in terms of the above as

$$\Psi_1(P_T, P_S) = \begin{bmatrix} A_{TT}^T P_T + P_T A_{TT} & P_T A_{TS}^{+} & P_T B_T \\ * & -P_S & 0 \\ * & 0 & -I \end{bmatrix} \tag{7.24}$$

$$\Psi_2(P_T, P_S) = \begin{bmatrix} 0 & P_T A_{TS}^{-} & 0 \\ * & P_S & 0 \\ * & 0 & I \end{bmatrix} \tag{7.25}$$

$$M_1 = \begin{bmatrix} I & 0 & 0 \\ A_{ST}^{-} & A_{SS}^{-} & B_S^{-} \\ 0 & 0 & I \end{bmatrix}, \quad M_2 = \begin{bmatrix} I & 0 & 0 \\ A_{ST}^{+} & A_{SS}^{+} & B_S^{+} \\ C_T & C_S & D \end{bmatrix}. \tag{7.26}$$

Three observations about the above theorem can be made at this point.

1. Condition (7.23) is an LMI in P_T and P_S.

2. In contrast to the corresponding result for LTI systems, the theorem provides only a sufficient condition for stability and performance.

3. The LMI condition that has to be solved has the size of a single subsystem; it is independent of the number of subsystems (which is actually assumed to be infinite).

From (7.24) - (7.26) it is clear that the left hand side $J(P_T, P_S)$ of (7.23) is a 3×3 block matrix, *i.e.* the condition has the form

$$\begin{bmatrix} J_{11} & J_{12} & J_{13} \\ * & J_{22} & J_{23} \\ * & * & J_{33} \end{bmatrix} < 0,$$

where $*$ denotes blocks that are required for symmetry. This inequality is required to hold in order to enforce the condition

$$[x^T \;\; v^T \;\; d^T] \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ * & J_{22} & J_{23} \\ * & * & J_{33} \end{bmatrix} \begin{bmatrix} x \\ v \\ d \end{bmatrix} < 0 \quad \forall 0 \neq \begin{bmatrix} x \\ v \\ d \end{bmatrix} \in \mathbb{R}^{n+2m+n_d},$$

which in turn implies that

$$\left\langle \begin{bmatrix} x(t) \\ v(t) \\ d(t) \end{bmatrix}, J \begin{bmatrix} x(t) \\ v(t) \\ d(t) \end{bmatrix} \right\rangle_{\ell_2} < 0 \quad \forall 0 \neq \begin{bmatrix} x(t) \\ v(t) \\ d(t) \end{bmatrix} \in \ell_2^{n+2m+n_d}.$$

Stability Condition

To illustrate the idea behind the proof of this theorem, a brief outline of a derivation of the stability condition is given here. Note that stability is implied by the existence of matrices $P_T > 0$ and $P_S = P_S^T$ that satisfy

$$\tilde{J} = \begin{bmatrix} J_{11} & J_{12} \\ J_{12}^T & J_{22} \end{bmatrix} < 0, \tag{7.27}$$

because only the signals $x(t)$ and $v(t)$ are involved in the stability condition. Let accordingly $\tilde{M}_1$, $\tilde{M}_2$, $\tilde{\Psi}_1$ and $\tilde{\Psi}_2$ denote the upper left 2×2 submatrices of the matrices M_1, M_2, Ψ_1 and Ψ_2, respectively. Note that (7.27) implies

$$\left\langle \begin{bmatrix} x(t) \\ v(t) \end{bmatrix}, \tilde{J} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \right\rangle_{\ell_2} < 0 \quad \forall 0 \neq \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \in \ell_2. \tag{7.28}$$

To see that the existence of $P_T > 0$ and $P_S = P_S^T$ that satisfy (7.28) is a sufficient condition for stability, we use the definition of the inner product on ℓ_2 to rewrite (7.28) as

$$\sum_{s=-\infty}^{\infty} [x^T(t,s) \;\; v^T(t,s)] \Big(\tilde{M}_1^T \tilde{\Psi}_1 \tilde{M}_1 + \tilde{M}_2^T \tilde{\Psi}_2 \tilde{M}_2 \Big) \begin{bmatrix} x(t,s) \\ v(t,s) \end{bmatrix} < 0 \quad \forall 0 \neq \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \in \ell_2.$$

The left hand side can be decomposed into two infinite sums, the first involving $\tilde{M}_1^T \tilde{\Psi}_1 \tilde{M}_1$ and the second $\tilde{M}_2^T \tilde{\Psi}_2 \tilde{M}_2$. Again using the notation

$$v = \begin{bmatrix} v^+ \\ v^- \end{bmatrix}, \qquad w = \begin{bmatrix} w^+ \\ w^- \end{bmatrix}$$

one can check, using (7.26), that in the first sum we have

$$\tilde{M}_1 \begin{bmatrix} x(t) \\ v^+(t,s) \\ v^-(t,s) \end{bmatrix} = \begin{bmatrix} x(t) \\ v^+(t,s) \\ w^-(t,s) \end{bmatrix}$$

and similarly in the second sum

$$\tilde{M}_2 \begin{bmatrix} x(t) \\ v^+(t,s) \\ v^-(t,s) \end{bmatrix} = \begin{bmatrix} x(t) \\ w^+(t,s) \\ v^-(t,s) \end{bmatrix}.$$

Thus left multiplication of the signal vector by $\tilde{M}_1$ only changes v^- into w^- and leaves the other signal components unchanged. Left multiplication by $\tilde{M}_2$ changes v^+ into w^+. Also note that

$$\begin{bmatrix} v^+(t,s) \\ w^-(t,s) \end{bmatrix} = \begin{bmatrix} SI_m & 0 \\ 0 & S^{-1}I_m \end{bmatrix} \begin{bmatrix} w^+(t,s) \\ v^-(t,s) \end{bmatrix}. \tag{7.29}$$

It is now straightforward to check that (7.28) can be written as

$$\begin{aligned} \left\langle \begin{bmatrix} x(t) \\ v(t) \end{bmatrix}, \tilde{J} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \right\rangle_{\ell_2} &= \langle \mathcal{A}x, P_T x\rangle_{\ell_2} + \langle P_T x, \mathcal{A}x\rangle_{\ell_2} \\ &+ \left\langle \begin{bmatrix} w^+(t,s) \\ v^-(t,s) \end{bmatrix}, \tilde{J} \begin{bmatrix} w^+(t,s) \\ v^-(t,s) \end{bmatrix} \right\rangle_{\ell_2} - \left\langle \begin{bmatrix} v^+(t,s) \\ w^-(t,s) \end{bmatrix}, \tilde{J} \begin{bmatrix} v^+(t,s) \\ w^-(t,s) \end{bmatrix} \right\rangle_{\ell_2} < 0 \\ &\qquad \forall 0 \neq \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \in \ell_2. \end{aligned}$$

Because of (7.29), the last two terms in the inequality represent inner products of shifted versions of the same sequences; they are therefore equal and their difference is zero. As a consequence, (7.28) is equivalent to

$$\langle \mathcal{A}x, P_T x\rangle_{\ell_2} + \langle P_T x, \mathcal{A}x\rangle_{\ell_2} < 0 \qquad \forall 0 \neq x \in \ell_2^n,$$

which is the Lyapunov condition (7.22).

Exercises — Chapter 7

Problem 7.1 *(State Space Realization of a PDE)*

Learning Goals

- Deriving the state space realization of a given partial differential equation (PDE)

Task Description Euler-Bernoulli beam theory is one of the simplest and most useful beam theories in structural analysis. The Euler-Bernoulli equation (7.30) provides a means of calculating the applied load and deflection characteristics of a homogeneous beam.

$$EI\frac{\partial^4 y(t,s)}{\partial s^4} + \rho A\frac{\partial^2 y(t,s)}{\partial t^2} = u(t,s), \tag{7.30}$$

where E denotes the Young's modulus, I the second moment of area of the cross-section, ρ the density, A the area of the cross-section, t and s the continuous temporal and spatial variables, respectively. Under the assumption of Euler-Bernoulli beam theory, the beam is subject only to lateral loads $u(t,s)$, whereas the deflection $y(t,s)$ does not count for shear deformation, as shown in Fig. 7.9.

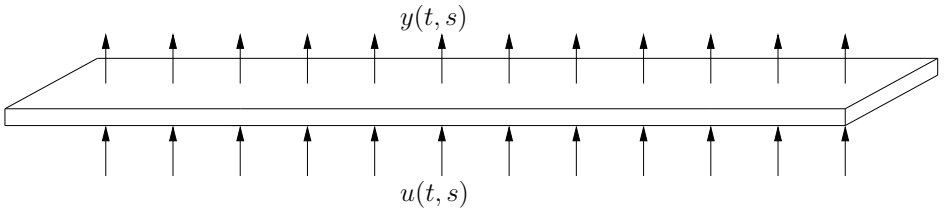


Figure 7.9: An Euler-Bernoulli beam

a) Spatially discretize the whole beam into the interconnection of identical subsystems as in Fig. 7.10. We denote the distance between any two neighbouring subsystems as h, and the discretized spatial variable also as s. Apply the central finite difference method to equation (7.30) in space to derive a 2-D equation of motion, which is temporally continuous and spatially discrete.

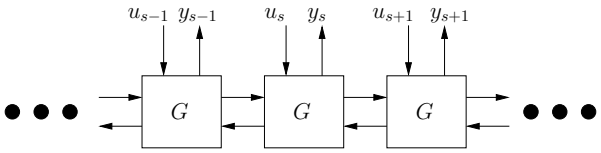


Figure 7.10: Discretize the beam into the interconnection of identical subsystems

b) Realize the state space representation of the 2-D transfer function obtained in a).

c) Discretize equation (7.30) both in time and space using central finite difference method. The sampling time is denoted as T.

d) Realize the state space representation of the 2-D difference equation obtained in c).

Problem 7.2 *(Simulate the Open Loop Response)*

Learning Goals

- Using MD toolbox to simulate the open loop response of a multi-dimensional system

Task Description Consider an aluminium beam of a length 4.8 m, a width 0.04 m and a thickness 0.004 m, as shown in Fig. 7.11. The aluminium has a density of 2710 Kg/m^3 and a Young's modulus 70×10^9 N/m^2. 16 pairs of collocated piezo actuators and sensors spatially discretize the beam into 16 identical subsystems.

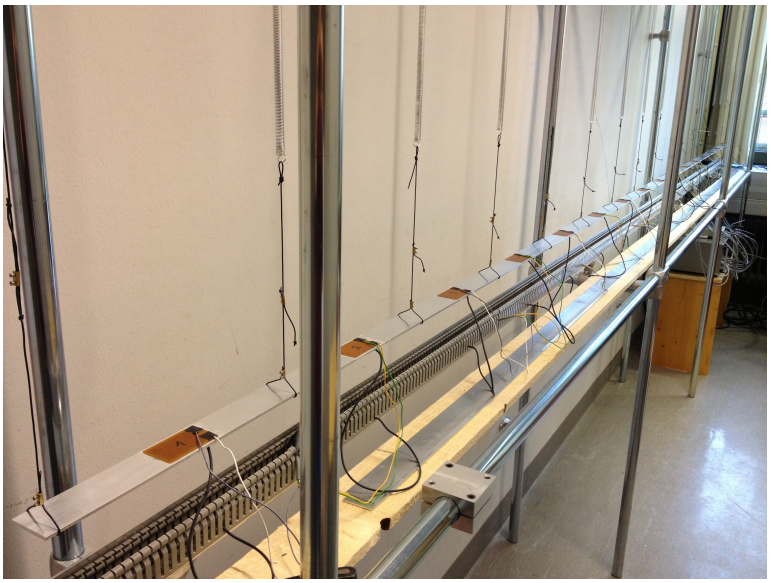


Figure 7.11: An aluminium beam equipped with 16 pairs of collocated actuators and sensors

a) Based on the state space model obtained in problem 7.1, use the MD toolbox (Multidimensional Systems Toolbox) to simulate the beam response to an impulse disturbance. Suppose the impulse is injected at subsystem 8 at time $t = 1$ s. (*A manual of the MD toolbox can be found in StudIP*)

b) Use the MD toolbox to simulate the beam response to a step input at subsystem 8.

Problem 7.3 *(Controller Design and Closed-loop Simulation)*

Learning Goals

- Construct the generalized plant of a single subsystem, shaping both the sensitivity and the control sensitivity.
- Design the subsystem controller K for subsystem G
- Simulate the closed-loop response.

Task Description Consider the Euler-Bernoulli beam equation 7.30 with the physical parameters in Problem 7.2. Fig. 7.12 shows the output amplitude spectrum up to 50 Hz of the subsystems 5 and 8 to an impulse disturbance. Assume that the first 3 modes are of our interest, i.e. $\omega_1 = 0.6$ Hz, $\omega_2 = 2.9$ Hz, and $\omega_3 = 6.8$ Hz. A distributed controller is to be designed to suppress the vibration at the first 3 resonance modes.

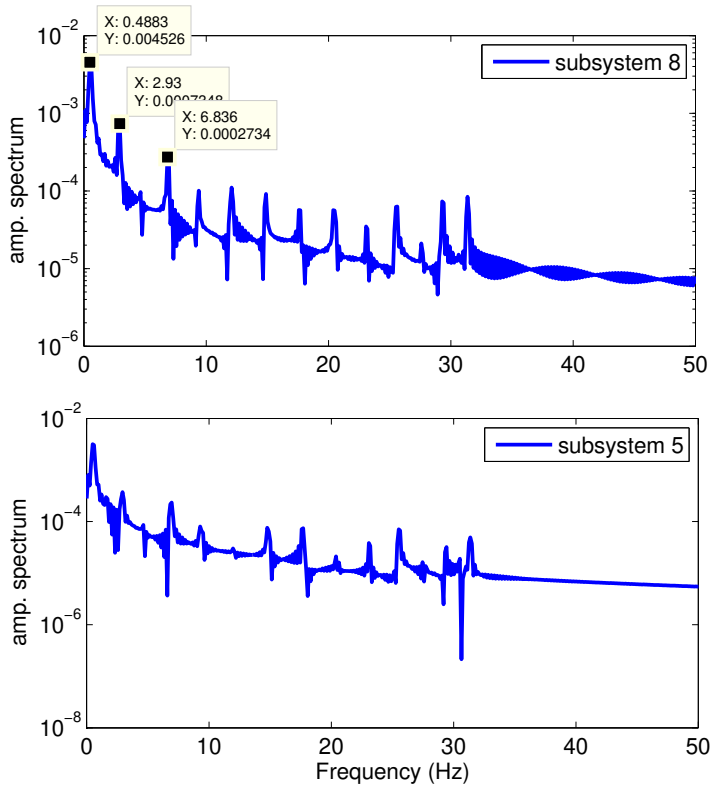


Figure 7.12: Output spectrum of subsystem 5 and 8 to an impulse disturbance

a) The control loop with a generalized plant for subsystem G is sketched in Fig. 7.13, with both the sensitivity and the control sensitivity being shaped. Construct the state space model of the generalized plant in an analytic way.

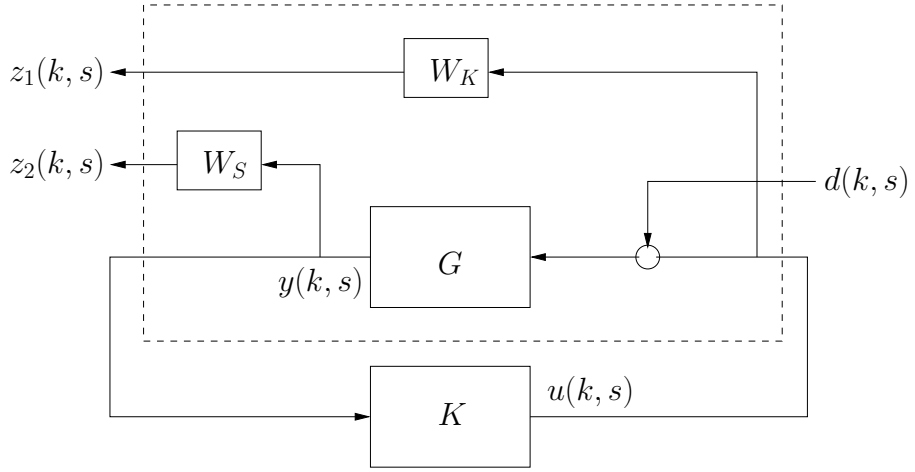


Figure 7.13: Control loop with a generalized plant of subsystem G

b) Suppose that both the shaping functions for control sensitivity and for sensitivity of first order are chosen:

$$W_S = \frac{\omega_S/M_S}{s+\omega_S} \tag{7.31}$$

$$W_K = \frac{c}{M_K}\frac{s+\omega_K}{s+c\omega_K}. \tag{7.32}$$

Use the function 'hfMD.m' of the MD toolbox to design a distributed controller and 'csimulateMD.m' (or 'dsimulateMD.m' or 'dsimulateMD_ATC.m') to simulate the closed-loop response. Tune your controller, such that the first 3 resonance modes are efficiently suppressed, and the maximum control effort does not exceeds 15 N/m.

Hint: you can start with a choice of $\omega_s = 0.1$, $M_s = 1$, $M_k = 10$, $\omega_K = 1$, $c = 100$

Part IV

APPENDICES

Appendix A

Solutions to Exercises

Solutions — LPV Systems I

Problem 1.1 *(Hidden Coupling)*

a) To obtain the equilibrium control input $\bar{u} = \tanh^{-1}(\rho)$, note that the PI controller's state takes the value $\bar{\zeta} = K_I(\rho)^{-1}\bar{u}$. Substituting this in the state space model (1.14)

$$\begin{bmatrix} \dot{\bar{\zeta}} \\ \bar{u} \end{bmatrix} = \left[\begin{array}{c|c} 0 & 1 \\ \hline K_I(\rho) & K_P(\rho) \end{array}\right] \begin{bmatrix} \bar{\zeta} \\ \bar{z} \end{bmatrix} = \left[\begin{array}{c|c} 0 & 1 \\ \hline K_I(\rho) & K_P(\rho) \end{array}\right] \begin{bmatrix} K_I(\rho)^{-1}\bar{u} \\ \hline 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \bar{u} \end{bmatrix}$$

shows that the equilibrium families match.

b) It is observed that the closed loop becomes unstable if $r(t) = 0.7$, even though the linearized closed loop is stable for all ρ.

c) To linearize the controller compute

$$\begin{aligned} \dot{\tilde{\zeta}} &= \left.\frac{\mathrm{d}f_K}{\mathrm{d}\zeta}\right|_{\bar{\zeta},\ldots} \tilde{\zeta} + \left.\frac{\mathrm{d}f_K}{\mathrm{d}y}\right|_{\bar{\zeta},\ldots} \tilde{y} + \left.\frac{\mathrm{d}f_K}{\mathrm{d}r}\right|_{\bar{\zeta},\ldots} \tilde{r}, \\ \tilde{u} &= \left.\frac{\mathrm{d}h_K}{\mathrm{d}\zeta}\right|_{\bar{\zeta},\ldots} \tilde{\zeta} + \left.\frac{\mathrm{d}h_K}{\mathrm{d}y}\right|_{\bar{\zeta},\ldots} \tilde{y} + \left.\frac{\mathrm{d}h_K}{\mathrm{d}r}\right|_{\bar{\zeta}_K,\ldots} \tilde{r}, \end{aligned}$$

which results in

$$\begin{aligned} \dot{\tilde{\zeta}} &= \tilde{r} - \tilde{y}, \\ \tilde{u} &= \frac{1}{1-\rho^2}\tilde{\zeta} + \frac{1}{3(1-\rho^2)}(\tilde{r} - \tilde{y}) + \left.\frac{\partial h_K}{\partial \rho}\frac{\mathrm{d}\rho}{\mathrm{d}y}\right|_{\bar{\zeta},\ldots} \tilde{y}. \end{aligned}$$

Due to $h_K = K_I(\rho)\zeta + K_P(\rho)(r-y)$, we have

$$\begin{aligned}\frac{\partial h_K}{\partial \rho}\frac{\partial \rho}{\partial y} &= \frac{\partial}{\partial \rho}\left(K_I(\rho)\zeta + K_P(\rho)(r-y)\right)\cdot 1,\\ &= \frac{2\rho}{(1-\rho^2)^2}\left(\zeta + \frac{1}{3}(r-y)\right),\end{aligned}$$

which evaluated in the equilibrium $\bar{r} = \bar{y}$, $\bar{\zeta} = (1-\rho^2)\tanh^{-1}(\rho)$, gives

$$\begin{aligned}\dot{\tilde{\zeta}} &= \tilde{r} - \tilde{y},\\ \tilde{u} &= \frac{1}{1-\rho^2}\tilde{\zeta} + \frac{1}{3(1-\rho^2)}(\tilde{r}-\tilde{y}) + \frac{2\rho\tanh^{-1}(\rho)}{(1-\rho^2)}\tilde{y}.\end{aligned}$$

By identifying $\tilde{y} = (1-\rho^2)\tilde{x}$ from the linearized plant above, the closed-loop system description yields

$$\left[\begin{array}{c}\dot{\tilde{x}}\\ \dot{\tilde{\zeta}}\\ \hdashline \dot{\tilde{z}}\end{array}\right] = \left[\begin{array}{cc:c} -\frac{4}{3} + 2\rho\tanh^{-1}(\rho) & \frac{1}{(1-\rho^2)} & \frac{1}{3(1-\rho^2)}\\ -(1-\rho^2) & 0 & 1\\ \hdashline -(1-\rho^2) & 0 & 1\end{array}\right]\left[\begin{array}{c}\tilde{x}\\ \tilde{\zeta}\\ \hdashline \tilde{r}\end{array}\right].$$

Plotting the real parts of the closed-loop eigenvalues for values of $\rho \in [-0.9, 0.9]$ reveals that they are negative for $\rho \in [-0.7, 0.7]$.

d*) The Matlab file `model_Hidden_Coupling_mod.slx` compares both implementations. The only difference is that the integral gain is swapped with the integrator. Note that for time-invariant gains this would not affect the controller at all, but that the time varying gain, although scalar, does not commute with the integrator.

For small steps ($r < 0.4$) the responses are similar, although with slightly less overshoot and hence faster settling time. For larger steps ($r > 0.8$), the response with the modified controller is essentially unchanged and does not become unstable.

This form of implementation is a simple example of what is known as *velocity algorithm* (see Section 4.4 of the Rugh and Shamma Paper for details and references). The velocity algorithm is common in practice and is proven to remove any hidden couplings in gain-scheduled control systems. Note, however, that stability and performance results remain limited to some neighborhood of the equilibrium condition and that transients between different equilibria are not fully incorporated.

Problem 1.2 *(Inferring LTV from LTI Stability)*

a) As the systems are both deprived of damping, the systems are marginally stable as illustrated by the phase plots in Fig. A.2(a). Checking the eigenvector/-value pairs, one observes two distinct purely imaginary eigenvalues λ_1, λ_2 and corresponding eigenvectors v_1, v_2.

A general solution to the ordinary differential equation can be found in

$$y(t) = C_1 v_1 e^{\lambda_1 t} + C_2 v_2 e^{\lambda_2 t},$$

or—more specific to our case of complex conjugate eigenvalues, with $\lambda = \lambda_1 = \bar{\lambda}_2 = \alpha + j\beta$ and $v = v_1 = \bar{v}_2 = u + jw$—one express the solution as

$$y(t) = e^{\alpha t} \left(C_1 \left(u \cos(\beta t) - w \sin(\alpha t) \right) + C_2 \left(u \sin(\beta t) + w \cos(\alpha t) \right) \right).$$

For $\alpha = 0$, it becomes clear that the trajectory will circle endlessly on an ellipsoid. The spring stiffness determines, how far away from the equilibrium the turning point will be. Intuitively, we can imagine that the higher the stiffness, the closer the cart remains to the origin and that it will also achieve higher velocities.

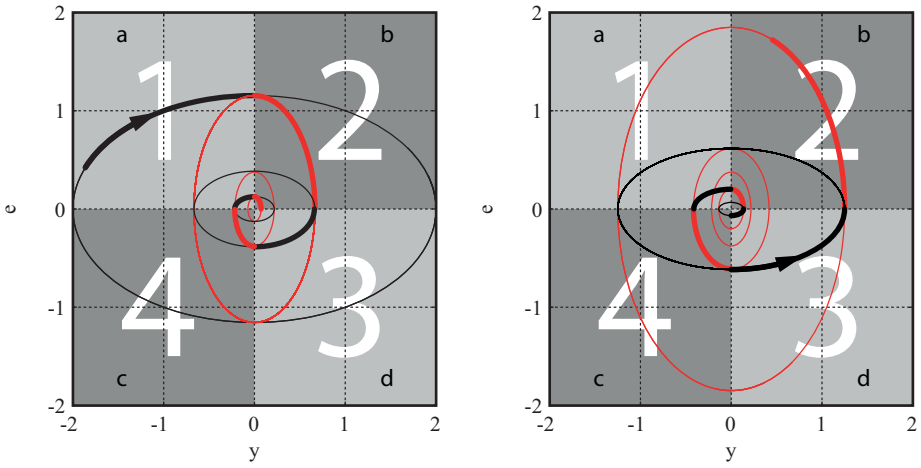
b) As the systems are now damped, the systems are stable as illustrated by the phase plots in Fig. A.2(b). Due to the negative real part the trajectories spiral towards the equilibrium.

c) Observe that the switching law is designed to incur switches at the main axes. The resulting trajectories can be seen in Figs. A.2(c) and A.2(d). To explain the effect, consider the marginally stable case again. In these cases, the switching and resulting (in-)stability is illustrated in Figs. A.1(a) and A.1(b), respectively. It is interesting to note again that both systems individually are marginally stable systems. The only difference to the switching between the stable systems resides in the existence of the spiral trajectories, which are not circling close enough to the equilibrium within a single quadrant.

d) Physically, we simulate an immediate and sudden change in the spring's stiffness. If the stiffness is lowered, however, the spring's potential energy is lowered as well, which is computed from

$$E_{\text{pot}} = \frac{1}{2} k y^2.$$

The system is therefore destabilized by introducing energy at the right time.



(a) Quadrant-wise stabilizing switching (b) Quadrant-wise destabilizing switching

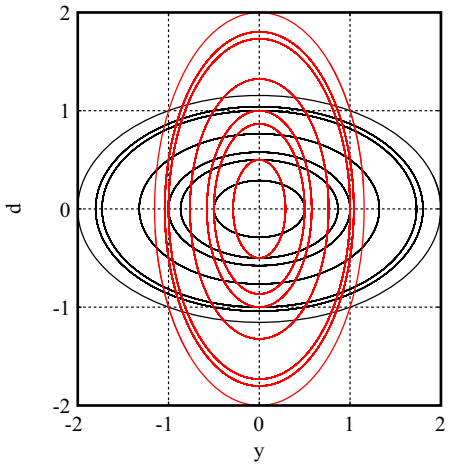
Figure A.1: Quadrant-wise stabilizing switching

e) The proposed smoothly time-varying stiffness appears to not destabilize the system, as apparent from Fig. A.2(e). A variation law for the stiffness that lets the system behave lightly damped could be

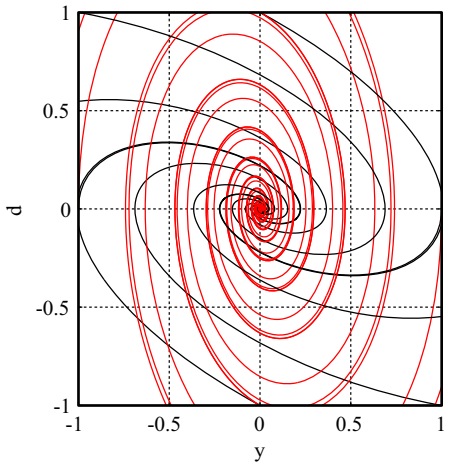
$$k(t) = 1 - \frac{1}{2}\sin(2t).$$

and a resulting trajectory is shown in Fig. A.2(f). Can you find a destabilizing law? In fact, it is possible which makes one realize that the trouble of finding a Lyapunov function is one of the few options to systematically deal with nonlinear systems. Try the following law

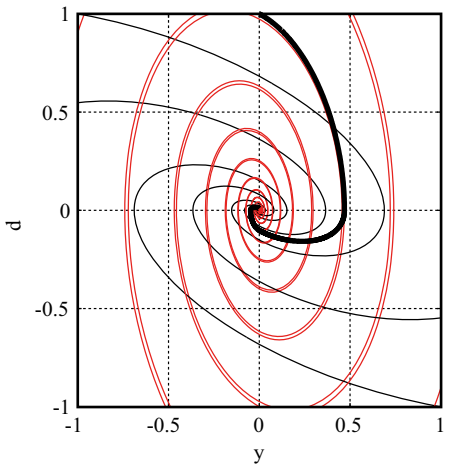
$$k(t) = 2 - 2\cos(\pi t).$$



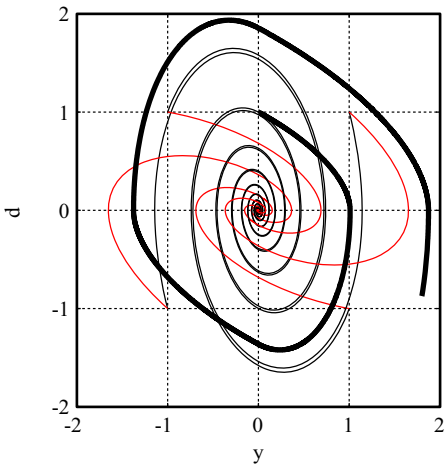
(a) Marginal stability due to lack of damping



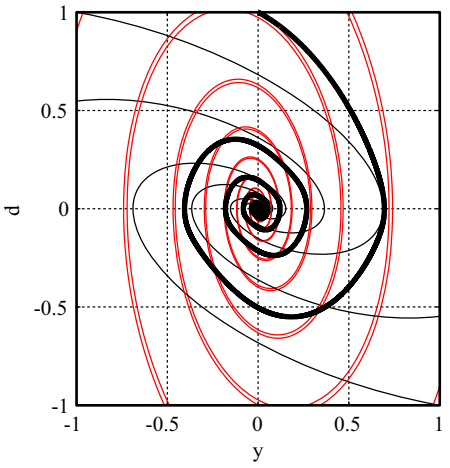
(b) Stable system responses due to positive damping



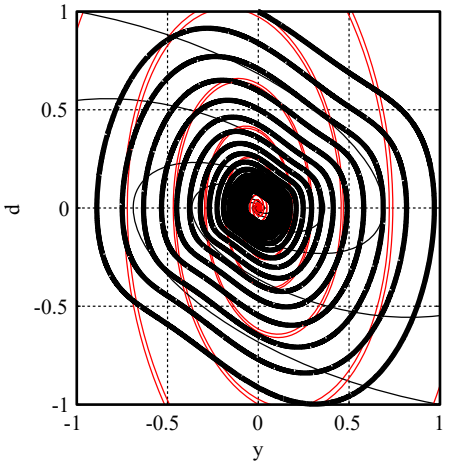
(c) Stable system response due to beneficial quadrant-wise switching law



(d) Unstable system response due to adverse quadrant-wise switching law



(e) Stable system response due to some smooth parameter-variation law



(f) System response showing lightly damped modes due to some smooth parameter-variation law

Figure A.2: Various phase plots for different scenarios

Solutions — LPV Systems II

Problem 2.1 *(Multiple Equilibria of Nonlinear Systems)*

a) The distinct equilibria of the simple undamped pendulum are intuitively found as the ones shown in Fig. A.3, namely $\bar{\alpha}_0 = 0$ and $\bar{\alpha}_1 = \pi$. This is verified by setting

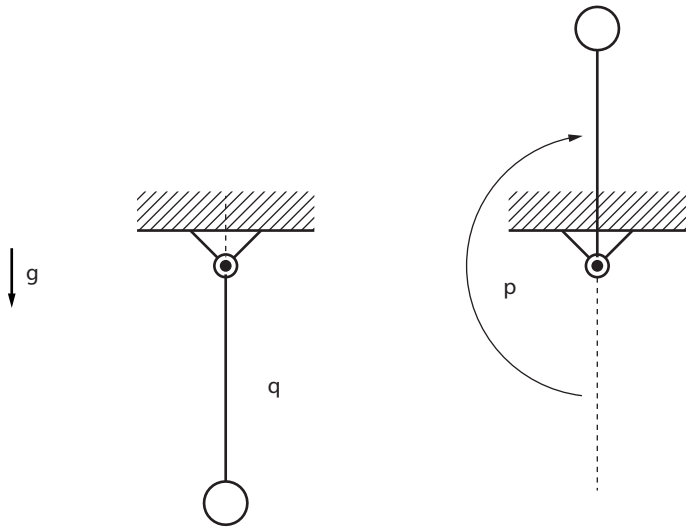


Figure A.3: Equilibria of the simple pendulum

the time derivatives to zero

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{g}{L}\sin(\bar{\alpha}) \end{bmatrix}, \tag{A.1}$$

which is clearly true in the above-mentioned cases. In fact, this holds true for $\bar{\alpha}_k = k\pi$.

b) The phase portrait can be determined by either solving for the trajectories with initial conditions gridded over the state-space, or by evaluating the Jacobian in each grid point to give an indication of the trajectories instantaneous direction.

$$\begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(\alpha) & -b \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(\alpha) & 0 \end{bmatrix}, \text{ for } b = 0.$$

Inserting the equilibria, we obtain

$$\begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(k\pi) & 0 \end{bmatrix}.$$

Whenever k is an odd number the cosine is -1 and we obtain

$$M_{k=2n-1} = \left.\begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(k\pi) & 0 \end{bmatrix}\right|_{k=2n-1} = \begin{bmatrix} 0 & 1 \\ \frac{g}{L} & 0 \end{bmatrix}, \; n \in \mathbb{Z}.$$

On the other hand, whenever k is an even number the cosine is 1 and we obtain

$$M_{k=2n} = \left.\begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(k\pi) & 0 \end{bmatrix}\right|_{k=2n} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} & 0 \end{bmatrix},\ n \in \mathbb{Z}.$$

The associated eigenvalues are therefore

$$\lambda_{k=2n-1} = \lambda\left(M_{k=2n-1}\right) = \pm\sqrt{\frac{g}{L}},$$
$$\lambda_{k=2n} = \lambda\left(M_{k=2n}\right) = \pm j\sqrt{\frac{g}{L}},$$

since $\det\left(M_k - \lambda I\right) = \lambda^2 \pm \frac{g}{L}$.

The zero real parts of $\lambda_{k=2n}$ indicate that we are dealing with a marginally stable equilibrium. In fact, this point is a center, because the product of both eigenvalues $\pm j\sqrt{\frac{g}{L}}$ is positive. In terms of the definitions for stability and attractiveness, the equilibria at $(2n\pi, 0),\ n \in \mathbb{Z}$ are stable, but not attractive. This holds true as well for the saddle points.

Loosely speaking, $\lambda_{k=2n-1}$ indicates saddle points, because of the mixed positive/negative real parts. In conclusion the phase portrait looks as in Fig. A.4. Simply

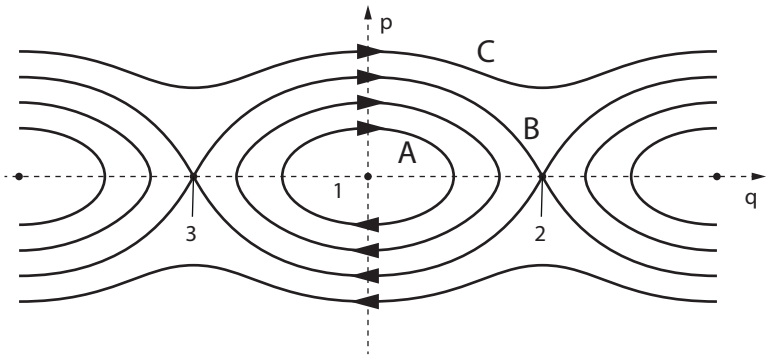


Figure A.4: Phase portrait of the simple undamped pendulum

put, trajectories of type A are regular pendulum motions oscillating back and forth. Trajectories of type B are the limiting case, where the amplitude of the oscillation is strong enough to reach the upper equilibrium point. It can then either return or and therefore move into the regime, where $\dot{\alpha} < 0$ or pass the tipping point and therefore the angular velocity will be sign-definite for all times.

c) The damped pendulum's Jacobian is

$$\begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(\alpha) & -b \end{bmatrix}.$$

We now have

$$M_{k=2n-1} = \left.\begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(k\pi) & -b \end{bmatrix}\right|_{k=2n-1} = \begin{bmatrix} 0 & 1 \\ \frac{g}{L} & -b \end{bmatrix},\ n \in \mathbb{Z}.$$

and

$$M_{k=2n} = \left.\begin{bmatrix} 0 & 1 \\ -\frac{g}{L}\cos(k\pi) & -b \end{bmatrix}\right|_{k=2n} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} & -b \end{bmatrix}, \; n \in \mathbb{Z}.$$

For the respective eigenvalues we obtain

$$\lambda_{k=2n-1} = \lambda\left(M_{k=2n-1}\right) = -\frac{b}{2} \pm \sqrt{\frac{b^2}{4} + \frac{g}{L}},$$
$$\lambda_{k=2n} = \lambda\left(M_{k=2n}\right) = -\frac{b}{2} \pm \sqrt{\frac{b^2}{4} - \frac{g}{L}},$$

The equilibria at $(2n\pi, 0), \; n \in \mathbb{Z}$ are now asymptotically stable, because they are also attractive. The other equilibria remain saddle points and are therefore not attractive.

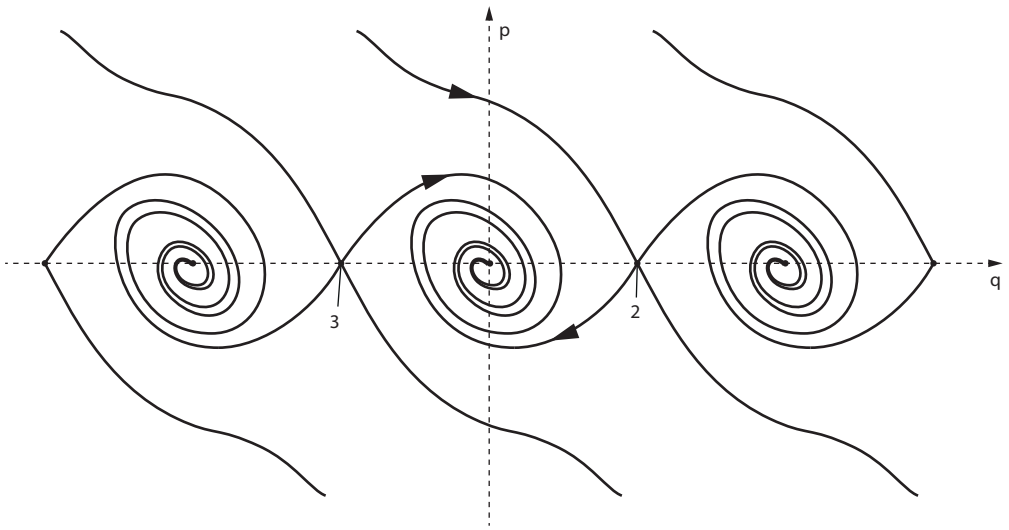


Figure A.5: Phase portrait of the simple damped pendulum

Problem 2.2 *(Prove Stability by Lyapunov Function)*

a) The total energy is found by first determining the kinetic energy

$$E_{\text{kin}}(t) = \frac{1}{2}L^2\dot{\alpha}^2.$$

The potential energy is

$$E_{\text{pot}}(t) = gL(1 - \cos(\alpha)).$$

The total energy is the sum of the energies

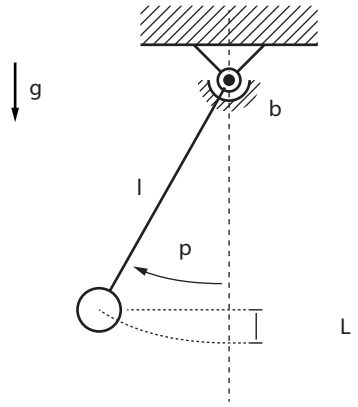


Figure A.6: The simple pendulum

$$E(t) = E_{\text{kin}}(t) + E_{\text{pot}}(t) = \frac{1}{2}L^2\dot{\alpha}^2 + gL(1 - \cos(\alpha)).$$

b) To prove Lyapunov stability, we require

$$V(t) > 0 \quad \text{and} \quad \dot{V}(t) \leq 0, \quad \forall t \geq 0.$$

The energy only contains squared terms and $(1-\cos(\alpha))$, which is always non-negative. Taking the time derivative of $V(t)$, we obtain

$$\begin{aligned}\dot{V}(t) &= \dot{\alpha}(L^2\ddot{\alpha} + gL\sin(\alpha)) \\ &= \dot{\alpha}\left(L^2(-\frac{g}{L}\sin(\alpha) - b\dot{\alpha}) + gL\sin(\alpha)\right) \\ &= -bL^2\dot{\alpha}^2 \leq 0.\end{aligned}$$

It can be observed that in the absence of damping, the conditions on Lyapunov stability are still fulfilled as $\dot{V}(t) = 0$. Asymptotic stability can be established by finding continuous and strictly increasing functions $a(\|x\|)$, $b(\|x\|)$ and $c(\|x\|)$, with

$a(0) = 0$, $b(0) = 0$ and $c(0) = 0$, and a ball around the equilibrium of some radius $r > 0$, denoted $\mathbb{B}_r$, such that

$$\begin{aligned} &\text{(a)} \quad a(\|x\|) \leq V(t,x) \leq b(\|x\|),\ \forall t \geq t_0, \forall x \in \mathbb{B}_r, \\ &\text{(b)} \quad \dot{V}(t,x) \leq -c(\|x\|),\ \forall t \geq t_0, \forall x \in \mathbb{B}_r \end{aligned}$$

If we now consider a ball around the equilbrium $\alpha = 0$ of radius π it is sufficient to show

$$V(t,x) > 0,\ \dot{V}(t,x) < 0,\ \forall x \neq 0 \in \mathbb{B}_r,$$

which is certainly true for the proposed Lyapunov function candidate. This proves asymptotic stability.

Problem 2.3 *(Scheduled Stabilizing State Feedback)*

a) The second state is directly controllable from the input and can therefore be stabilized, even though it is open-loop unstable. The first state is stable, though, and therefore the plant is stabilizable independent of the value for ρ.

b) Affine parameter dependence is easily established by defining $\theta = \frac{\rho}{2-\rho^2}$. We then have

$$A(\theta) = \begin{bmatrix} -1 & \theta \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} + \theta \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = A_0 + \theta A_1.$$

All other matrices remain unchanged. The bounds on the new parameter can be determined to $P_\theta = [-1, 1]$. The bounds on the time derivative are more difficult to derive. From

$$\frac{\mathrm{d}}{\mathrm{d}t}\theta = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\rho}{2-\rho^2} = \frac{2+\rho^2}{(2-\rho^2)^2}\sigma$$

it can be checked that $P_\nu = [-3, 3]$.

c) Starting from the LMI

$$P(\theta)(A(\theta) + BF(\theta)) + A^\top(\theta)P(\theta) + \sum_{i=1}^{n_\theta} \nu_i \frac{\partial P(\theta)}{\partial \theta_i} < 0, \quad \forall \theta \in P_\theta,\ \forall \nu \in P_\nu, \tag{A.2}$$

$$P(\theta) > 0, \quad \forall \theta \in P_\theta \tag{A.3}$$

By pre- and postmultiplying with $Q(\theta) = P^{-1}(\theta) > 0$, and making use of the identity $\frac{\partial P^{-1}(\theta)}{\partial \theta} = -P^{-1}(\theta)\frac{\partial P(\theta)}{\partial \theta}P^{-1}(\theta)$, this is equivalent to

$$A(\theta)Q(\theta) + BF(\theta)Q(\theta) + Q(\theta)A^\top(\theta) + Q(\theta)F^\top(\theta)B^\top + \sum_{i=1}^{n_\theta} \nu_i \frac{\partial Q(\theta)}{\partial \theta_i} < 0, \tag{A.4}$$

$$\forall \theta \in P_\theta,\ \forall \nu \in P_\nu,$$

$$Q(\theta) > 0, \quad \forall \theta \in P_\theta \tag{A.5}$$

By using the change of variable $Y(\theta) = F(\theta)Q(\theta)$, the matrix inequality is rendered linear in $Q(\theta)$ and $Y(\theta)$. Making use of the Ansatz, we have for our problem

$$\mathcal{M}(\theta, \nu) = A(\theta)Q(\theta) + BY(\theta) + Q(\theta)A^\top(\theta) + Y^\top(\theta)B^\top + \sum_{i=1}^{n_\theta} \nu_i \frac{\partial Q(\theta)}{\partial \theta_i} < 0, \tag{A.6}$$

$$\forall (\theta, \nu) \in P_\theta \times P_\nu.$$

Including the multi-convexity condition

$$\frac{\partial^2}{\partial \theta^2}\mathcal{M}(\theta, \nu) \geq 0$$

renders the condition $\mathcal{M}(\theta, \nu) < 0$, which is quadratic in θ, convex. More explicitly, this means that the additional LMI constraint

$$Q_1 A_1^\top + A_1 Q_1 \geq 0 \tag{A.7}$$

needs to be imposed, such that $\mathcal{M}(\theta, \nu) < 0$ can be solved in the four extremal combinations of parameter range and rate of change. Finally, (A.6), (A.7) and

$$Q(\theta) = Q_0 + \theta Q_1 > 0, \quad \forall \theta \in P_\theta \tag{A.8}$$

make up the finite set of LMIs, necessary in order to solve for the decision variables Q_0, Q_1, Y_0 and Y_1 by means of solving a semi-definite program.

Solutions — LPV Systems III

Problem 3.1 *(Parameter-Dependent Lyapunov Function Based Analysis)*

a) We assume that the answer is "No" and try to find at least two points within the parameter range, for which we cannot find a single matrix $P = P^T > 0$, such that

$$\left(A(\rho) + BD_K(\rho)\right)^T P + P\left(A(\rho) + BD_K(\rho)\right) < 0.$$

Let's first assume $F(\rho)$ has the form

$$F(\rho) = \begin{bmatrix} F_1(\rho) & F_2(\rho) \end{bmatrix} \in \mathbb{R}^{2\times 4}.$$

For $\rho_1 = 0$ and $\rho_2 = \pi$, we have

$$\begin{aligned} \left(A(0) + BF(0)\right)^T P + P\left(A(0) + BF(0)\right) < 0, \\ \left(A(\pi) + BF(\pi)\right)^T P + P\left(A(\pi) + BF(\pi)\right) < 0, \end{aligned}$$

or more explicitly

$$\begin{aligned} \left(\begin{bmatrix} A_{11} & I_{2\times 2} \\ 0_{2\times 2} & -\tau I_{2\times 2} \end{bmatrix} + \tau \begin{bmatrix} 0_{2\times 2} & 0_{2\times 2} \\ F_1(0) & F_2(0) \end{bmatrix}\right)^T P + P\,(*) < 0, \\ \left(\begin{bmatrix} A_{11} & -I_{2\times 2} \\ 0_{2\times 2} & -\tau I_{2\times 2} \end{bmatrix} + \tau \begin{bmatrix} 0_{2\times 2} & 0_{2\times 2} \\ F_1(\pi) & F_2(\pi) \end{bmatrix}\right)^T P + P\,(*) < 0. \end{aligned}$$

Adding the above inequalities and dividing by 2 gives

$$\left(\begin{bmatrix} A_{11} & 0_{2\times 2} \\ \frac{1}{2}(F_1(0) + F_1(\pi)) & -\tau I_{2\times 2} + \frac{1}{2}(F_2(0) + F_2(\pi)) \end{bmatrix}\right)^T P + P\,(*) < 0,$$

The actually relevant part is found in

$$A_{11}^T P_{11} + P_{11} A_{11} < 0,$$

if P is partitioned as $P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$. Now it is easy to observe that since $A_{11} > 0$ it is impossible to find a $P_{11} > 0$ that renders the above inequality fulfilled.

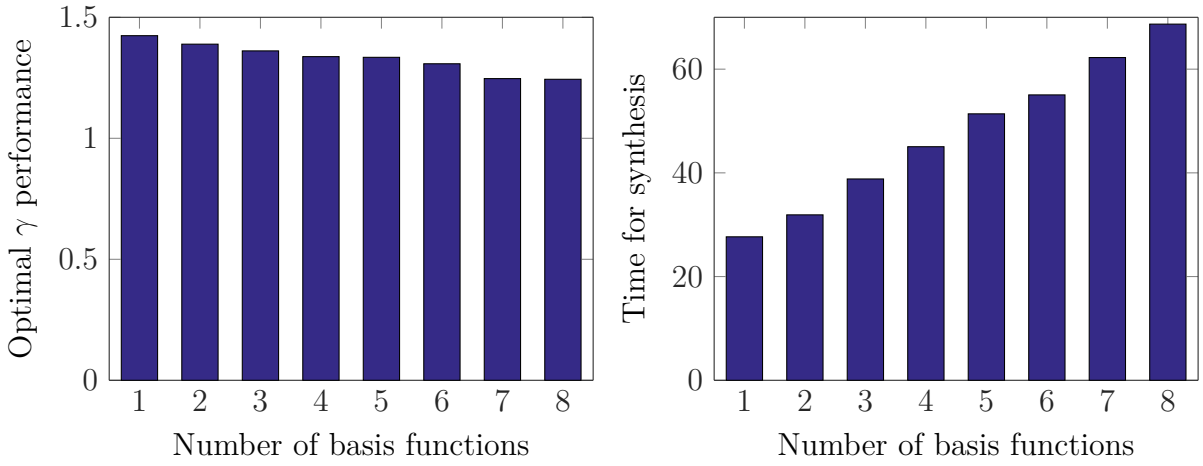
b) Please refer to the MATLAB file `exercise_Analysis_State_Feedback.m` for a detailed solution.

Problem 3.2 *(State Feedback Synthesis with Parameter-Dependent Lyapunov Functions)*

a) The practical solution is implemented as `lpvsfsyn` in the LPVTools. Further, there as a modified synthesis routine available as `lpvsfsynth` that improves the condition number of the Lyapunov matrix P. Note that F is calculated from P^{-1}.

b) With parameter-dependent Lyapunov functions $P(p) = \sum_i^{n_c} f(\rho_i)\, P_i$, the goal is to reduce conservatism (and hence increase performance).

We expect that by providing more degrees of freedom, i. e. more basis functions, it should be easier for the optimization to find a feasible solution and hence that more basis functions lead to better performance. On the other hand, more basis functions increase the number of decision variables in the optimization and this increases computational complexity.



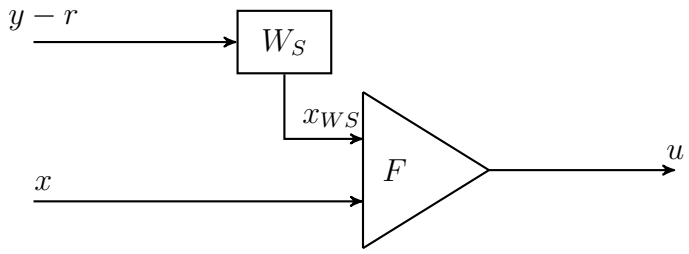
1 No parameter dependence

2–4 Linear parameter dependence ($\dot{q}_1$, q_2, q_3)

5–7 Quadratic parameter dependence ($\dot{q}_1^2$, q_2^2, q_3^2)

8 Multilinear parameter dependence ($\dot{q}_1\, q_2\, q_3$)

c) The controller requires access to all states of the generalized plant. This includes those of the weighting filters. Hence, the filters have to be implemented together with the state feedback gain to form an augmented state feedback controller.



Please refer to the MATLAB file `atc_GyroSimulation` for implementation details.

Implement the explicit state feedback gain formula:

- evaluate parameter dependent B_u and C_1 matrices,
- evaluate parameter dependent P matrix ,
- invert P with LDLT-factorization,
- calculate parameter dependent F from B_u, C_1, P^{-1} and any scalings / transformations used for synthesis,
- implement LTI weighting filters.

This approach is feasible for real-time application and gives the "exact" feedback gain, but the implementation it is rather involved and error-prone.

Implementing the controller based on a lookup table:

- define the lookup table for F,
- implement LTI weighting filters.

This approach is easy to implement and requires less online computation in exchange for possibly more memory requirements. Further, it only provides an approximation of the "exact" feedback gain. Confirm this by looking at the difference of the gain matrices in the Simulink model.

Note that there is also a Block provided by the LPVTools that implements the complete parameter-dependent dynamic controller as an S-Function that interpolates a lookup table representation.

Solutions — LPV Systems IV

Problem 4.1 *(LFTs - Pulling out the Deltas (Simple Example))*

a) Substitute $m(t) = m_0 + \delta_1$ and observe that

$$\frac{1}{m(t)} = \frac{1}{m_0 + \delta_1} = \delta_1 \star \left[\begin{array}{c|c} -m_0^{-1} & m_0^{-1} \\ \hline -m_0^{-1} & m_0^{-1} \end{array}\right] = \delta_1 \star M_m$$

It is useful to factorize the state-space model as

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1/m(t) \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -k(t) & -b \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

and write

$$\begin{bmatrix} 1 & 0 \\ 0 & 1/m(t) \end{bmatrix} = \delta_1 \star \left[\begin{array}{c|cc} -m_0^{-1} & 0 & m_0^{-1} \\ \hline 0 & 1 & 0 \\ -m_0^{-1} & 0 & m_0^{-1} \end{array}\right].$$

Now we just have to substitute and expand to obtain

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \delta_1 \star \left[\begin{array}{c|cc} -m_0^{-1} & -m_0^{-1}k_0 & -m_0^{-1}b \\ \hline 0 & 0 & 1 \\ -m_0^{-1} & -m_0^{-1}k_0 & -m_0^{-1}b \end{array}\right] \begin{bmatrix} y \\ \dot{y} \end{bmatrix}.$$

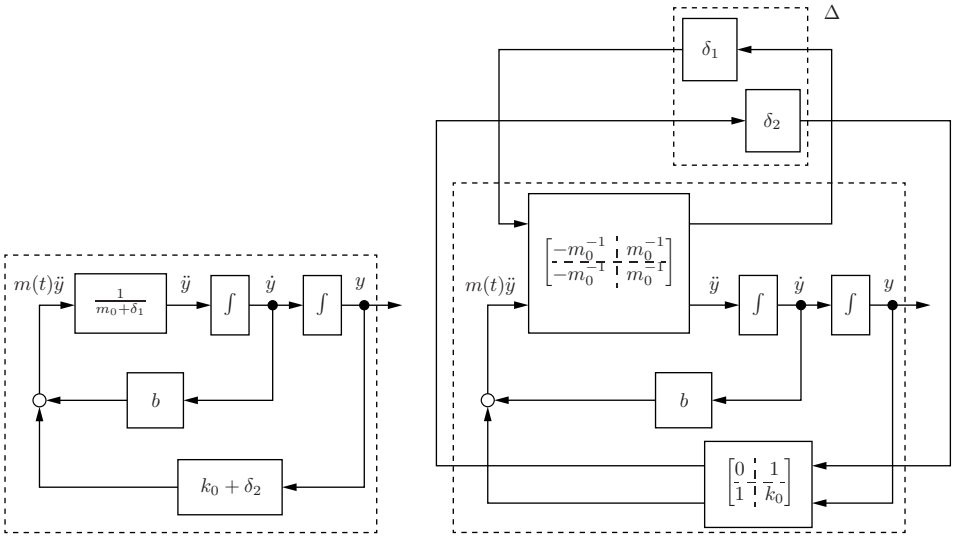
b) Before expanding, we first find an LFT representation for the right hand factor

$$\begin{bmatrix} 0 & 1 \\ -k(t) & -b \end{bmatrix} = \delta_2 \star \left[\begin{array}{c|cc} 0 & 1 & 0 \\ \hline 0 & 0 & 1 \\ -1 & -k_0 & -b \end{array}\right] = \delta_2 \star \left[\begin{array}{c|c} M_{k,11} & M_{k,12} \\ \hline M_{k,21} & M_{k,22} \end{array}\right].$$

Note that $M_{k,11} = 0$, since the LFT is affine in δ_2. Finally, we just have to apply the rules for multiplying two LFTs to obtain

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} \delta_1 & \\ & \delta_2 \end{bmatrix} \star \left[\begin{array}{cc|cc} -m_0^{-1} & -m_0^{-1} & -m_0^{-1}k_0 & -m_0^{-1}b \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 \\ -m_0^{-1} & -m_0^{-1} & -m_0^{-1}k_0 & -m_0^{-1}b \end{array}\right] \begin{bmatrix} y \\ \dot{y} \end{bmatrix}.$$

Fig. A.7 recovers the same representation by graphical interpretation.



(a) General representation (b) "Deltas" pulled out

Figure A.7: Block diagram of the mass-spring-damper system's differential equations in linear fractional representation

Problem 4.2 *(LFT of system with repeated parameters)*

a) Starting with the output equation

$$y(\rho) = (1 - \rho^2),$$

we can expand and rewrite it as

$$y(\rho) = x - \rho\rho x.$$

By declaring new inputs $w_1 = \rho w_2$, $w_2 = \rho x$ we can then rewrite the output equation as

$$y(\rho) = x - w_1$$

. As usual, the input-output mapping for these new inputs is $w_1 = \rho z_1$, $w_2 = \rho z_2$, which means $z_1 = x$ and $z_2 = w_1$.

We can now proceed to write the above equations in matrix form

content...

$$\left(\begin{bmatrix} A_{11} & 0_{2\times 2} \\ \frac{1}{2}(F_1(0) + F_1(\pi)) & -\tau I_{2\times 2} + \frac{1}{2}(F_2(0) + F_2(\pi)) \end{bmatrix}\right)^T P + P\,(*) < 0,$$

The actually relevant part is found in

$$A_{11}^T P_{11} + P_{11} A_{11} < 0,$$

if P is partitioned as $P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$. Now it is easy to observe that since $A_{11} > 0$ it is impossible to find a $P_{11} > 0$ that renders the above inequality fulfilled.

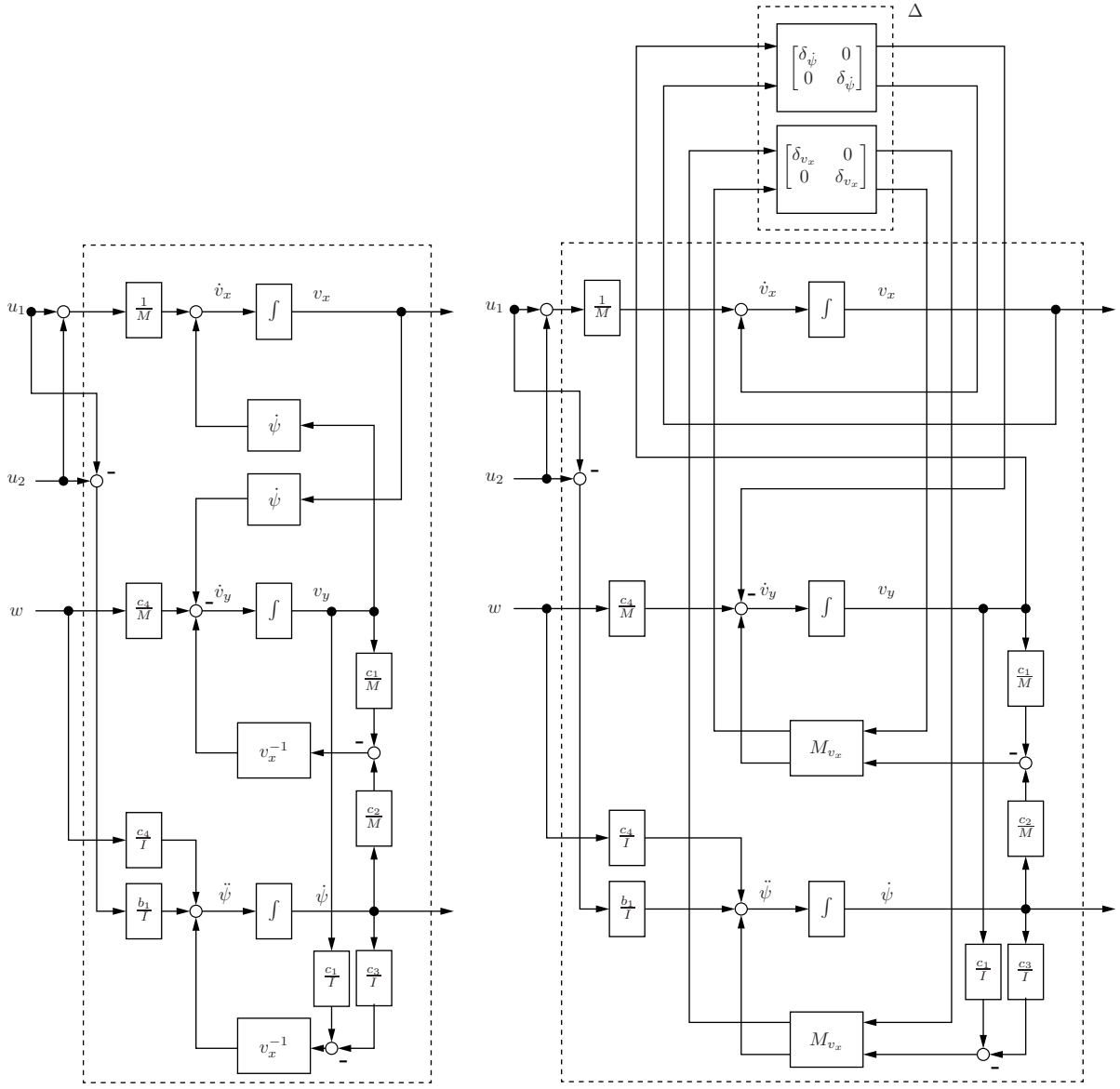
b) Please refer to the MATLAB file `exercise_Analysis_State_Feedback.m` for a detailed solution.

Problem 4.3 *(LFTs - Pulling out the Deltas)*

a) Substitute $v_x = v_{x,0} + \delta_{v_x}$ and observe that

$$\frac{1}{v_x} = \frac{1}{v_{x,0} + \delta_{v_x}} = \delta_{v_x} \star \left[\begin{array}{c|c} -v_{x,0}^{-1} & v_{x,0}^{-1} \\ \hline -v_{x,0}^{-1} & v_{x,0}^{-1} \end{array}\right] = \delta_{v_x} \star M_{v_x}$$

For the state-space description we can derive



(a) General representation (b) "Deltas" pulled out

Figure A.8: Block diagram of the vehicle's differential equations in linear fractional representation

$$\begin{bmatrix} \dot{v}_x \\ \dot{v}_y \\ \ddot{\psi} \\ \hdashline z_\Delta \\ \hdashline v_x \\ v_y \end{bmatrix} = \left[\begin{array}{ccc:ccc:ccc:cc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \frac{1}{M} & \frac{1}{M} \\ 0 & \frac{c_1}{Mv_{x,0}} & \frac{c_2}{Mv_{x,0}} & -v_{x,0}^{-1} & 0 & 0 & -1 & \frac{c_4}{M} & 0 & 0 \\ 0 & \frac{c_1}{Iv_{x,0}} & -\frac{c_3}{Iv_{x,0}} & 0 & -v_{x,0}^{-1} & 0 & 0 & \frac{c_4}{I} & -\frac{b_1}{I} & \frac{b_1}{I} \\ \hdashline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{c_1}{Iv_{x,0}} & -\frac{c_3}{Iv_{x,0}} & 0 & -v_{x,0}^{-1} & 0 & 0 & 0 & 0 & 0 \\ -\frac{c_2}{Mv_{x,0}} & \frac{c_1}{Mv_{x,0}} & 0 & -v_{x,0}^{-1} & 0 & 0 & 0 & 0 & 0 & 0 \\ \hdashline 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} v_x \\ v_y \\ \dot{\psi} \\ \hdashline w_\Delta \\ \hdashline w \\ u_1 \\ u_2 \end{bmatrix} \tag{A.9}$$

$$\begin{bmatrix} \dot{x} \\ \hdashline z_\Delta \\ y \end{bmatrix} = \left[\begin{array}{c:ccc} A & B_\Delta & B_w & B_u \\ \hdashline C_\Delta & D_{\Delta\Delta} & D_{\Delta w} & D_{\Delta u} \\ C_z & D_{z\Delta} & D_{zw} & D_{zu} \\ C_y & D_{y\Delta} & D_{yw} & D_{yu} \end{array}\right] \begin{bmatrix} x \\ \hdashline w_\Delta \\ w \\ u \end{bmatrix} \tag{A.10}$$

Problem 4.4 *(Torque Vectoring by LPV Control)*

a/b) For an extensive solution to the problem, please consult the MATLAB files.

Problem 4.5 *(Parameter Set Mapping vs. LFT (Simple Example))*

a) Using $\theta_1 = \frac{1}{L}$, $\theta_2 = \frac{1}{L^2}$ and $\theta_3 = \frac{\dot{\varphi}}{L}$ an affine LPV model reads as

$$\dot{x} = \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\varphi} \\ \ddot{\varphi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{m_T} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{b}{m_T}\theta_1 & -\frac{d}{m_L}\theta_2 & -g\theta_1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{m_T} & 0 \\ 0 & 1 \\ -\frac{1}{m_T}\theta_1 & -2\theta_3 \end{bmatrix} \begin{bmatrix} u_T \\ u_L \end{bmatrix} \tag{A.11}$$

b) When investigating the relation between $\theta_1 = \frac{1}{L}$ and $\theta_2 = \frac{1}{L^2}$ by plotting them on two axes, cf. Fig. A.9(a), it becomes apparent that only a line is actually physically possible. If simply the combinations of parameter bounds are used to derive the bounding box, any controller synthesis based on these parameter limits would guarantee performance and stability for many parameter pairs that can never occur in reality.

A solution exists in considering the actual convex hull of the parameters, shown in Fig. A.9(b). However, this leads to numerous vertices to be considered due to the curve-like shape of the physically admissible pairs of parameters.

It is intuitive to then consider a tighter, rotated, rectangular bounding box as a convex region as in Fig. A.9(c). This box can be systematically derived via a method called parameter set mapping, which is based on a principle component analysis.

c) By decomposing Θ into

$$\Theta = U\Sigma V^*$$

one can obtain information on the linear dependency of the rows. The matrix U contains both the bases for the column space of Θ as well as the basis of the left nullspace of Θ, cf. Fig. A.10. Now since Θ is a fat matrix and the rows will not be linearly dependent, we can expect the left nullspace to be empty. In turn, U will provide a unitary basis for the column space of our parameter samples.

d) Reinterpret $\Phi = \Sigma V^*$ as horizontally concatenated vectors of coefficients that are used to build weighted sums of the basis vectors contained in U. Therefore from

$$\Theta = U\Phi, \qquad U^\top\Theta = \Phi,$$

we can perform a coordinate transformation into directions ϕ_1, ϕ_2 that are maximally independent. Accordingly define

$$\theta = U\phi, \qquad U^\top\theta = \phi, \qquad \theta = [\theta_1\ \theta_2]^\top,\ \phi = [\phi_1\ \phi_2]^\top.$$

By identifying bounds on the values of ϕ and converting them back into coordinates in terms of θ by premultiplication with U (linear combination of the basis vectors), the bounding box shown in Fig. A.9(c) is obtained.

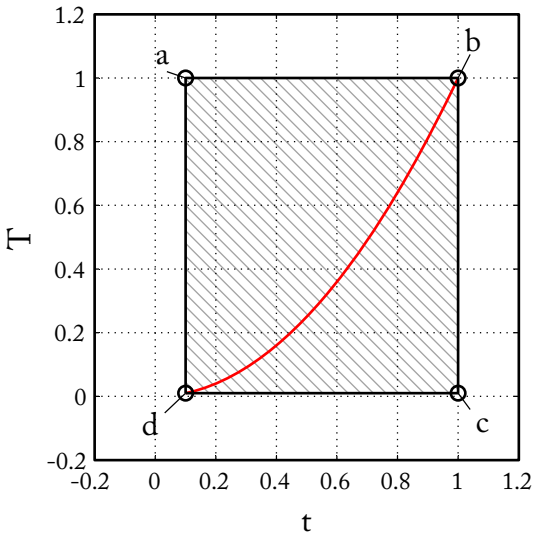
e) With

$$U = \begin{bmatrix} U_s & U_n \end{bmatrix} = \begin{bmatrix} -0.7915 & -0.6112 \\ -0.6112 & 0.7915 \end{bmatrix},$$
$$\Sigma = \begin{bmatrix} \sigma_s & 0 & \dots & 0 \\ 0 & \sigma_n & \dots & 0 \end{bmatrix} = \begin{bmatrix} 7.6655 & 0 & 0 & \dots & 0 \\ 0 & 0.9378 & 0 & \dots & 0 \end{bmatrix},$$

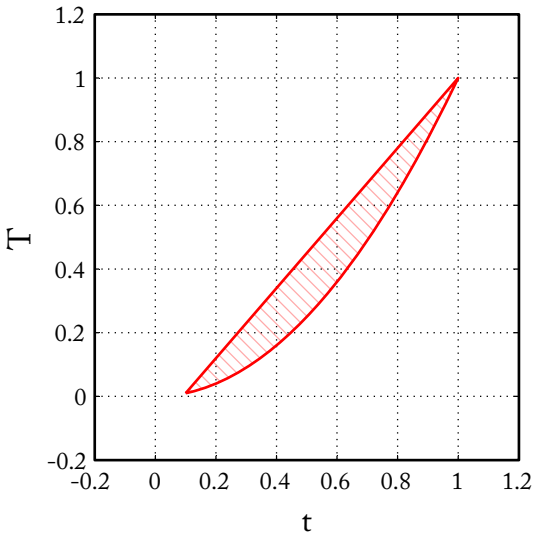
the significant singular value is identified via the drop in an order of magnitude versus the second singular value. Accordingly the first column of U corresponds to the significant basis vector.

A reduced parameter set is obtained by projection

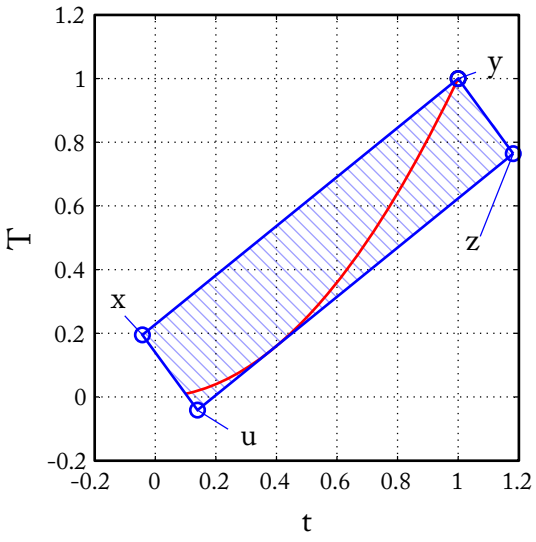
$$\phi = U_s^\top \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}.$$



(a) Naive bounding box derived from maximum and minimum parameter values



(b) Tight convex hull involving many vertices (not shown)



(c) Tight rectangular bounding box derivex by parameter set mapping

Figure A.9: Overbounding in different parameter sets

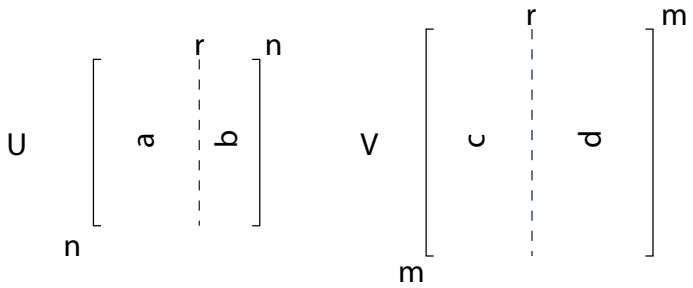


Figure A.10: Basis of matrix spaces using SVD of a matrix $A \in \mathbb{C}^{n \times m}$ with the first r singular values being non-zero

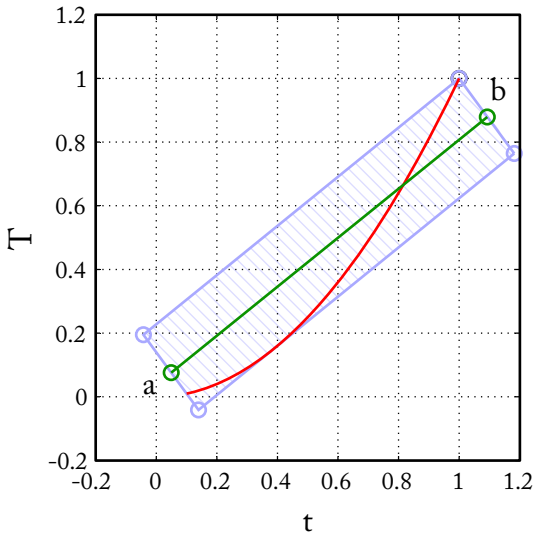


Figure A.11: Reduced parameter set ϕ in green. The non-approximated parameter set is shown in light blue.

Problem 4.6 *(Parameter Set Mapping and Control — 2-DOF Robotic Manipulator)*

a/b) For an extensive solution to the problem, please consult the MATLAB files.

Problem 4.7 *(LPV Control of a 2-DOF Robotic Manipulator)*

a/b) For an extensive solution to the problem, please consult the MATLAB files.

Solutions — Multi-Agent Systems I

Problem 5.1 *(Directed Graphs)*

The solution for the network states is given by $x(t) = e^{-L(\mathcal{D})t}x_0$. Similar to the undirected case, we will use a decomposition for $L(\mathcal{D})$. Whereas $L(\mathcal{G})$ is symmetric and can thus be diagonalized, $L(\mathcal{D})$ is not symmetric, why the Jordan decomposition is used. Let $L(\mathcal{D}) = PJP^{-1}$ be the Jordan decomposition with

$$J = \begin{bmatrix} J(0) & 0 & \cdots & 0 \\ 0 & J(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & J(\lambda_n) \end{bmatrix}.$$

Here the λ_is are the eigenvalues of $L(\mathcal{D})$ and $J(\lambda_i)$ is the respective Jordan block with the dimension of the algebraic multiplicity of λ_i. From Theorem F.2 it is known that the algebraic multiplicity of the zero eigenvalue is one, if and only if there exist at least one vertex $v \in \mathcal{V}$, from who every other vertex can be reached by respecting the direction of the edges. Therefore the Jordan block $J(0) = 0$.

Similar to the undirected case, the decomposition is used to show that

$$\begin{aligned} x(t) &= e^{-L(\mathcal{D})t}x_0 \\ &= e^{-PJP^{-1}t}x_0 \\ &= Pe^{-Jt}P^{-1}x_0 \\ &= P\begin{bmatrix} e^0 & 0 & \cdots & 0 \\ 0 & e^{-J(\lambda_2)t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{-J(\lambda_n)t} \end{bmatrix} P^{-1}x_0. \end{aligned}$$

Since all nonzero eigenvalues of $L(\mathcal{D})$ have positive real parts, for all $i > 1$,

$$\lim_{t\to\infty} e^{-J(\lambda_i)t} = 0.$$

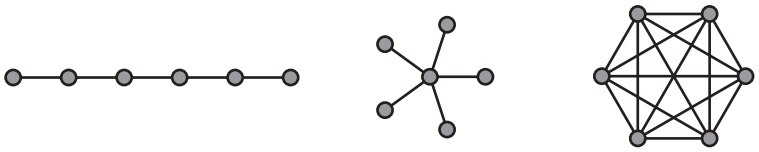
Therefore $\lim_{t\to\infty} x(t) = \left(q_1^T x_0\right) p_1$, where p_1 is the right eigenvector (1. column of P) and q_1 the respective left eigenvector (1. row of P^{-1}) corresponding to the zero eigenvalue. Due to construction of the Laplacian we know that $p_1 = \mathbf{1}$, and from the fact that $P^{-1}P = I$, $q_1^T\mathbf{1} = 1$ results.

Problem 5.2 *(Convergence Rate)*

a) To connect n vertices there are at least $m = n-1$ edges necessary. Those graphs can be for example the line graph or the star graph, shown in Fig. A.12(a) and A.12(b).

b) The complete graph shown in Fig. A.12(c) has $m = \frac{n(n-1)}{2}$ edges. If we count the edges, we get $n-1$ for the first vertex, $n-2$ new ones for the second on, etc. The last vertex then has 0 remaining edges.This can be written as the arithmetical series

$$m = \sum_{i=1}^{n} n - i = \frac{n(n-1)}{2}.$$



(a) Line graph (b) Star graph (c) Complete graph

Figure A.12: Basic graphs

c) For $n = 8$ the convergence rate is shown in Fig. A.13 depending on the number of edges, whereby for increasing the number of edges by one, a new edge is added to the previous graph. It can be observed that with increasing number of added edges the convergence rate increases as well.

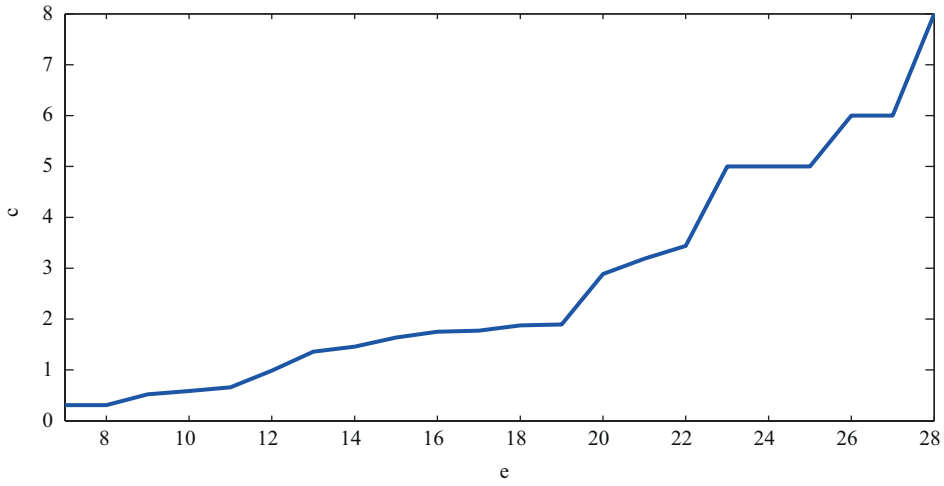


Figure A.13: Convergence rate for increasing number of edges

d) The graph $\mathcal{G}$ has λ_2 as 2nd smallest eigenvalue of the respective Laplace matrix L. The graph $\bar{\mathcal{G}}$ is $\mathcal{G}$ with the additional added edge $\{ij\}$ and $\bar{\lambda}_2$ the 2nd smallest

eigenvalue of the respective $\bar{L}$. Then we have

$$\bar{L} - L = \begin{array}{c} \begin{matrix} & & i & & j & & \end{matrix} \\ \begin{bmatrix} 0 & & \cdots & & & 0 \\ & \ddots & & & & \\ & & 1 & & -1 & & \\ \vdots & & & \ddots & & \vdots \\ & & -1 & & 1 & & \\ & & & & & \ddots & \\ 0 & & \cdots & & & 0 \end{bmatrix} \end{array} \begin{matrix} \\ \\ i \\ \\ j \\ \\ \end{matrix} .$$

Therefore the term hinted results in

$$\bar{x}^T \left(\bar{L} - L\right) \bar{x} = \bar{x}_i^2 - 2\bar{x}_i\bar{x}_j + \bar{x}_j^2 = (\bar{x}_i - \bar{x}_j)^2 \geq 0 \quad \bar{x}^T L\bar{x} \leq \bar{x}^T \bar{L}\bar{x}. \tag{A.12}$$

Assume that x is the eigenvector of L corresponding to λ_2, then

$$\lambda_2 = x^T Lx \leq \bar{x}^T L\bar{x}. \tag{A.13}$$

The inequality holds, since $\bar{x}$, as eigenvector of a Laplace matrix not corresponding to $\lambda_1 = 0$, is perpendicular to $\mathbf{1}$, which is the eigenvector corresponding to $\lambda_1 = 0$ of any Laplacian. With (A.12) and (A.13) we get the final result

$$\lambda_2 = x^T Lx \leq \bar{x}^T L\bar{x} \leq \bar{x}^T \bar{L}\bar{x} = \bar{\lambda}_2.$$

Problem 5.3 *(Time Delay)*

a) Fig. A.14 shows a possible graph $\mathcal{G}$ with 8 vertices and 10 edges (bold) and $\bar{\mathcal{G}}$ with additional added edge (dashed). The corresponding minimal time delays to destabilize the network are $\tau^\star = 0.3071$ and $\bar{\tau}^\star = 0.2824$. Thus we suspect, that an increasing number of edges leads to a decrease in the robustness against time delays.

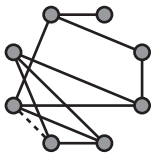


Figure A.14: Graph $\mathcal{G}$ ($\bar{\mathcal{G}}$) with 8 vertices and 10 (11) edges

b) The Laplace transformation of the network dynamics lead to

$$sX(s) - x(0) = -Le^{-s\tau}X(s).$$

Thus the transition matrix evolves to

$$X(s) = (sI + Le^{-s\tau})^{-1}x(0) = \Phi(s)x(0).$$

c) With the transition matrix derived in the previous task, the characteristic polynomial of $G(s)$ is

$$p(s) = \det(sI + Le^{-s\tau}) = \prod_{i=1}^{n}(s + \lambda_i e^{-s\tau}) = \prod_{i=1}^{n} p_i(s).$$

Therefore for stability of the network dynamics all solutions $s = \sigma \pm j\omega$ that satisfy $p(s) = s + \lambda_i e^{-s\tau} = 0$ have to have $\sigma < 0$. To find the minimal time delay that leads to instability, we determine the minimal time delay, such that $p(s)$ has a zero on the imaginary axis, such that $s = j\omega$. Substituted in $p(s)$ this leads to

$$p_i(j\omega) = \omega j + \lambda_i e^{-j\omega\tau} = \omega j + \lambda_i(\cos\omega\tau - j\sin\omega\tau) = 0.$$

Comparison of the real and imaginary part of the last equality leads to

$$\begin{aligned} \cos\omega\tau &= 0, \\ \omega - \lambda_i \sin\omega\tau &= 0. \end{aligned}$$

The first equation is solved for $\omega = \frac{\pi}{2\tau}$. Substituted in the second equality and solved for τ leads to $\tau = \frac{\pi}{2\lambda_i}$. Since we are looking for the minimal τ that leads to instability, we get

$$\tau^\star = \min_{i=1,\ldots,n} \frac{\pi}{2\lambda_i} = \frac{\pi}{2\lambda_n}.$$

d) The proof that $\bar{\lambda}_n \geq \lambda_n$ is very similar to the previous exercise d) and follows

$$\bar{\lambda}_n = \bar{x}^T \bar{L} \bar{x} \geq x^T \bar{L} x \geq x^T L x = \lambda_n.$$

Fig. A.15 shows the minimal destabilizing time delay $\tau^\star$ and the convergence rate for the same graphs as in the previous exercise c). As proofed the convergence rate increases with increasing number of added edges, while $\tau^\star$ decreases. Since large values of both characteristics are desired, a trade-off has to be made.

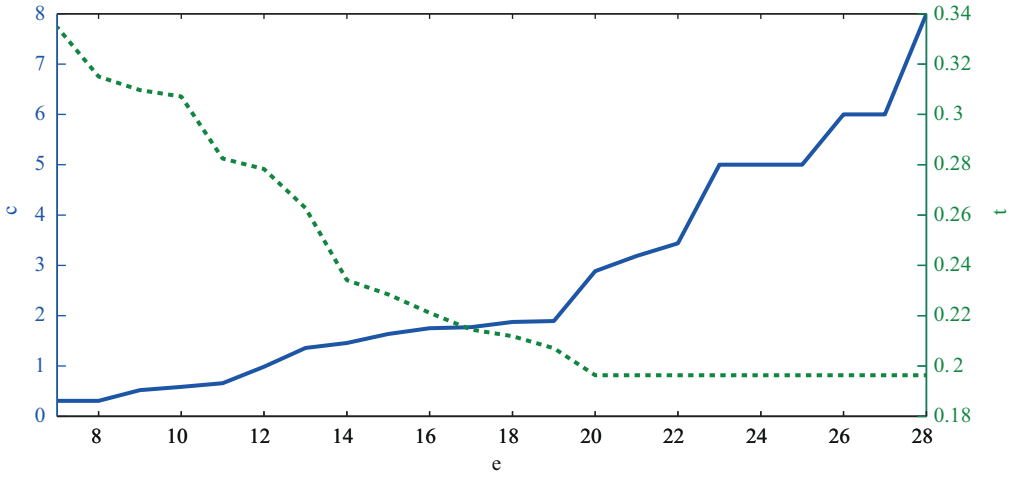


Figure A.15: Convergence rate and minimal destabilizing time delay for increasing number of edges

Problem 5.4 *(Discrete Consensus)*

a) For discrete systems we know that the system is unstable if there is an eigenvalues with absolute value greater than one. Therefore the real eigenvalues $\psi_n \leq \ldots \leq \psi_1$ of the symmetric matrix W have to fulfill $|\psi_i| \leq 1$ for all $i = 1, \ldots, n$. The networks dynamics are given by $x(k) = W^k x(0)$. Such that we have convergence to the average of the initial values, we need to enforce that

$$\lim_{k\to\infty} x(k) = \frac{\mathbf{1}\mathbf{1}^T}{n} x(0). \tag{A.14}$$

We know that $\lim_{k\to\infty} x(k) = \lim_{k\to\infty} W^k x(0)$. Using the decomposition $W = P^T \Psi P$ with $\Psi = \operatorname{diag}(\psi_1, \psi_1, \ldots, \psi_n)$ leads to

$$\begin{aligned}\lim_{k\to\infty} x(k) &= \lim_{k\to\infty} W^k x(0) \\ &= \lim_{k\to\infty} P^T \Lambda^k P x(0) \\ &= \left(\psi_1^k p_1^T p_1 + \psi_2^k p_2^T p_2 + \cdots \psi_n^k p_n^T p_n\right) x(0).\end{aligned} \tag{A.15}$$

All terms with $|\psi_i| < 1$ vanish. Comparing (A.14) and (A.15) leads to the conclusion that W needs one unique eigenvalue $\psi_1 = 1$ with corresponding eigenvector $\mathbf{1}$, while $|\psi_i| < 1$ for $i = 2, \ldots, n$ such that

$$-1 < \psi_n < \ldots < \psi_2 < \psi_1 = 1.$$

b) In the continuous case, where the closed-loop eigenvalues are $-\lambda_n, \ldots, -\lambda_2, 0$, the convergence rate is determined by λ_2: Thus by the distance between the imaginary axis and the largest closed-loop nonzero eigenvalue. With the discretization transformation $z = e^{sT}$ the imaginary axis in the continuous plane is mapped to the unit circle in the discrete plane and any parallel to the imaginary axis with negative real part to circles around the origin with radius smaller than one. Thus the distance, that determines the convergence rate in the continuous case, is in the discrete case the distance between one and the maximal absolute value of all eigenvalues of the discrete close-loop eigenvalues not equal to one. Thus the convergence rate of the discrete network is given by

$$\min_{i=2,\ldots n} 1 - |\psi_i| = 1 - \max_{i=2,\ldots n} |\psi_i| = 1 - \max(|\psi_2|, |\psi_n|).$$

This is shown schematically in Fig. A.16. The larger the gap the faster the convergence rate.

c) From a) we know, that the network converges to the average if $\mathbf{1}$ is an eigenvector of W corresponding to the eigenvalue 1. With the given W we have

$$W\mathbf{1} = (I - \alpha L)\mathbf{1} = \mathbf{1}$$

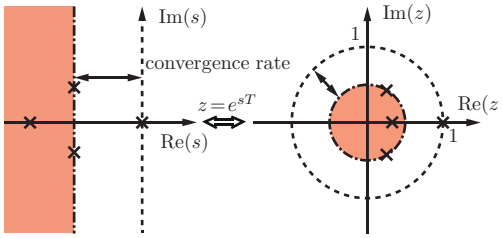


Figure A.16: Mapping between s and z plane to understand the definition of convergence rate

since $\mathbf{1}$ is an eigenvector of L corresponding to the eigenvalue 0. Thus we get average consensus. The eigenvalues of $W = I - \alpha L$ are $\psi_i = 1 - \alpha\lambda_i$.

For stability they have to fulfill $|\psi_i| < 1$ for $i = 2, \ldots, n$. This is not trivial if $\psi_i < 0$. Here the critical case is the smallest eigenvalue of W, that is $\psi_n = 1 - \alpha\lambda_n$, resulting from the largest eigenvalue of L, such that we need to fulfill $|1 - \alpha\lambda_n| < 1$, what leads to $\alpha < \frac{2}{\lambda_n}$. Thus the stabilizing region of alpha is $0 < \alpha < \frac{2}{\lambda_n}$.

By enlarging α we push ψ_2 away from 1 towards 0, thus $1 - \psi_2$ gets larger, but at the same time ψ_n is pushed towards -1, thus $1 - |\psi_n|$ gets smaller. Thus the $\alpha^\star$ for optimal convergence rate is achieved if

$$\begin{aligned} 1 - \psi_2 &= 1 - |\psi_n| \\ 1 - (1 - \alpha^\star\lambda_2) &= 1 - |1 - \alpha^\star\lambda_n| \\ \alpha^\star\lambda_2 &= 1 - (-1 + \alpha^\star\lambda_n) \\ \alpha^\star &= \frac{2}{\lambda_2 + \lambda_n}. \end{aligned}$$

For the bold graph in Fig. A.14 the convergence rate depending on α is shown in Fig. A.17. The optimal alpha determine here to $\alpha^\star = \frac{2}{\lambda_2+\lambda_n} = \frac{2}{0.59+5.11} = 0.35$.

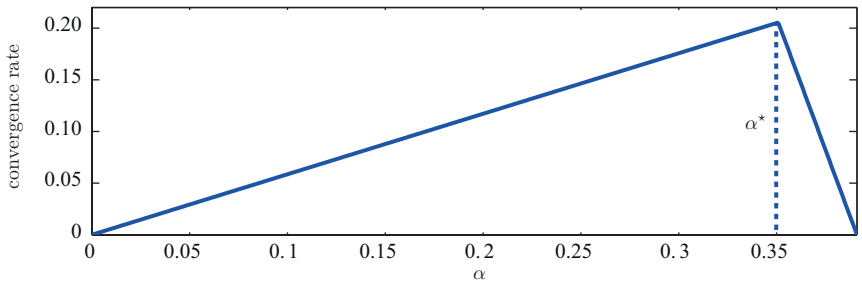


Figure A.17: Convergence rate depending on α

Solutions — Multi-Agent Systems II

Problem 6.1 *(Formation Stability for Arbitrary Topologies)*

a) The state equation of the closed loop system can be obtained as

$$\begin{bmatrix}\dot{x}\\ \dot{\zeta}\end{bmatrix} = \underbrace{\begin{bmatrix}\hat{A}_P + \hat{B}_P\hat{D}_K\hat{C}_P L_{(n)} & \hat{B}_P\hat{C}_K\\ \hat{B}_K\hat{C}_P L_{(n)} & \hat{A}_K\end{bmatrix}}_{\mathcal{A}}\begin{bmatrix}x\\ \zeta\end{bmatrix}. \tag{A.16}$$

Using the Schur transformation according to [20], we can transform $L_{(n)}$ into $U_{(n)}$, where U is an upper triangular matrix. Therefore, we perform the transformation $x = T_{(n)}\tilde{x}$ and $\zeta = T_{(n)}\tilde{\zeta}$ and achieve

$$\begin{bmatrix}T_{(n)} & 0\\ 0 & T_{(n)}\end{bmatrix}\begin{bmatrix}\dot{\tilde{x}}\\ \dot{\tilde{\zeta}}\end{bmatrix} = \begin{bmatrix}\hat{A}_P + \hat{B}_P\hat{D}_K\hat{C}_P L_{(n)} & \hat{B}_P\hat{C}_K\\ \hat{B}_K\hat{C}_P L_{(n)} & \hat{A}_K\end{bmatrix}\begin{bmatrix}T_{(n)} & 0\\ 0 & T_{(n)}\end{bmatrix}\begin{bmatrix}\tilde{x}\\ \tilde{\zeta}\end{bmatrix}$$

$$\Rightarrow \begin{bmatrix}\dot{\tilde{x}}\\ \dot{\tilde{\zeta}}\end{bmatrix} = \begin{bmatrix}T_{(n)}^{-1} & 0\\ 0 & T_{(n)}^{-1}\end{bmatrix}\begin{bmatrix}\hat{A}_P + \hat{B}_P\hat{D}_K\hat{C}_P L_{(n)} & \hat{B}_P\hat{C}_K\\ \hat{B}_K\hat{C}_P L_{(n)} & \hat{A}_K\end{bmatrix}\begin{bmatrix}T_{(n)} & 0\\ 0 & T_{(n)}\end{bmatrix}\begin{bmatrix}\tilde{x}\\ \tilde{\zeta}\end{bmatrix}$$

Using the rule $(A \otimes B)(C \otimes D) = AC \otimes BD$, we can rewrite the blocks of $\mathcal{A}$ like

$$\begin{aligned}&(T^{-1} \otimes I_n)(I_N \otimes A_P)(T \otimes I_n) = (T^{-1}I_N \otimes I_n A_P)(T \otimes I_n)\\ &= T^{-1}T \otimes A_P I_n = I_N \otimes A_P = \hat{A}_P\end{aligned} \tag{A.17}$$

and

$$\begin{aligned}&(T^{-1} \otimes I_n)(I_N \otimes B_P D_K C_P)(L \otimes I_n)(T \otimes I_n)\\ &= (T^{-1} \otimes B_P D_K C_P)(LT \otimes I_n)\\ &= T^{-1}LT \otimes B_P D_K C_P = (I_N \otimes B_P D_K C_P)(U \otimes I_n).\end{aligned} \tag{A.18}$$

This yields the transformed state equation

$$\begin{bmatrix}\dot{\tilde{x}}\\ \dot{\tilde{\zeta}}\end{bmatrix} = \underbrace{\begin{bmatrix}\hat{A}_P + \hat{B}_P\hat{D}_K\hat{C}_P U_{(n)} & \hat{B}_P\hat{C}_K\\ \hat{B}_K\hat{C}_P U_{(n)} & \hat{A}_K\end{bmatrix}}_{\tilde{\mathcal{A}}}\begin{bmatrix}\tilde{x}\\ \tilde{\zeta}\end{bmatrix}. \tag{A.19}$$

b) With the shorthand notation $\Psi = B_P D_K C_P$ and $\Phi = B_K C_P$, we can write (A.19)

as

$$\begin{bmatrix}\dot{\tilde{x}}\\ \dot{\tilde{\zeta}}\end{bmatrix} = \left[\begin{array}{cccc:cccc} A_P+\Psi\lambda_1 & \Psi u_{12} & \cdots & \Psi u_{1N} & B_PC_K & & & 0\\ 0 & \ddots & \ddots & \vdots & 0 & \ddots & & \\ \vdots & \ddots & \ddots & \Psi u_{N-1,N} & \vdots & & \ddots & \\ 0 & \cdots & 0 & A_P+\Psi\lambda_N & 0 & \cdots & 0 & B_PC_K \\ \hdashline \Phi\lambda_1 & \Phi u_{12} & \cdots & \Phi u_{1N} & A_K & & & 0\\ 0 & \ddots & \ddots & \vdots & 0 & \ddots & & \\ \vdots & \ddots & \ddots & \Phi u_{N-1,N} & \vdots & & \ddots & \\ 0 & \cdots & 0 & \Phi\lambda_N & 0 & \cdots & 0 & A_K \end{array}\right] \begin{bmatrix}\tilde{x}\\ \tilde{\zeta}\end{bmatrix} \tag{A.20}$$

Due to the triangular structure of U, the equations for $\dot{\tilde{x}}_N$ and $\dot{\tilde{\zeta}}_N$ do not contain any other states than $\tilde{x}_N$ and $\tilde{\zeta}_N$, thus they have the form of a closed loop around a single system (6.62) with $i = N$, which is stable iff $K(s)$ stabilizes (6.62) for $i = N$.

The equations for subsystem $N-1$ have the form of a closed loop system with states $\begin{bmatrix}\tilde{x}_{N-1}^T & \tilde{\zeta}_{N-1}^T\end{bmatrix}^T$ and external inputs $\tilde{x}_N$ and $\tilde{\zeta}_N$. The latter are bounded if subsystem N is stable and in this case do not influence the stability. This leads to the stability problem of single system (6.62) for $i = N-1$. The same observation is true for subsystem $N-2$ if the subsystems N and $N-1$ are stable. This argumentation can be continued until subsystem 1, which completes the proof.

Problem 6.2 *(Stability of a Multi-Vehicle Formation)*

a) Using the result from Problem 6.1, the stability condition on a closed loop multi-agent system can be reduced to a stability condition on a set of single agent systems parameterized with the eigenvalues of the Laplacian. In particular, the MAS with local controllers $K(s)$ is stable if and only if $K(s)$ simultaneously stabilizes the systems

$$\begin{aligned} \dot{x}_i &= A_P x_i + B_P u_i \\ z_i &= \lambda_i C_P x_i \end{aligned} \tag{A.21}$$

for all eigenvalues λ_i of the Laplacian.

For the topology shown in Fig. 6.12, the Laplacian matrix is obtained as

$$L = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 1 & -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & -\frac{1}{2} & 0 & -\frac{1}{2} & 1 \end{bmatrix}, \tag{A.22}$$

the eigenvalues are $eig(L) = \{0, 1, 1, 1, 2\}$. Computing the closed loop poles for the systems (A.21) and $K(s)$ yields

λ_i	0	1	2
poles	0.0000 + 0.0000i	-1.2000 + 0.7483i	-1.2000 + 1.6000i
	-2.4000 + 0.0000i	-1.2000 - 0.7483i	-1.2000 - 1.6000i
	-0.0500 + 0.0000i	-0.0500 + 0.0000i	-0.0500 + 0.0000i

As clearly visible, in all cases all poles are located in the closed left half plane, thus the systems (A.21) are at least marginally stable for all λ_i, from which we can infer stability of the whole MAS.

b) Computing the eigenvalues of $\mathcal{A}$ shows that none of them is located in the right half plane, thus the closed loop multi-agent system is stable. However, there is one zero eigenvalue of $\mathcal{A}$, which originates from the zero eigenvalue of L and corresponds to a non-zero equilibrium.

c) i) The Nyquist plot of the forward chain $P(s)K(s)$ is shown in Fig A.18. From evaluation of the real test point $s = \epsilon$ one can see that the Nyquist plot closes to the right.

ii) From (A.21), the stability of the formation is equivalent to the stability of the loop around $L_{\lambda_i}(s) = \lambda_i P(s)K(s)$ for all eigenvalues λ_i of the Laplacian. According to the Nyquist stability criterion (assuming $L_{\lambda_i}(s)$ to be open-loop stable), the Nyquist plot of $L_{\lambda_i}(s)$ must not encircle the point -1 for the loop

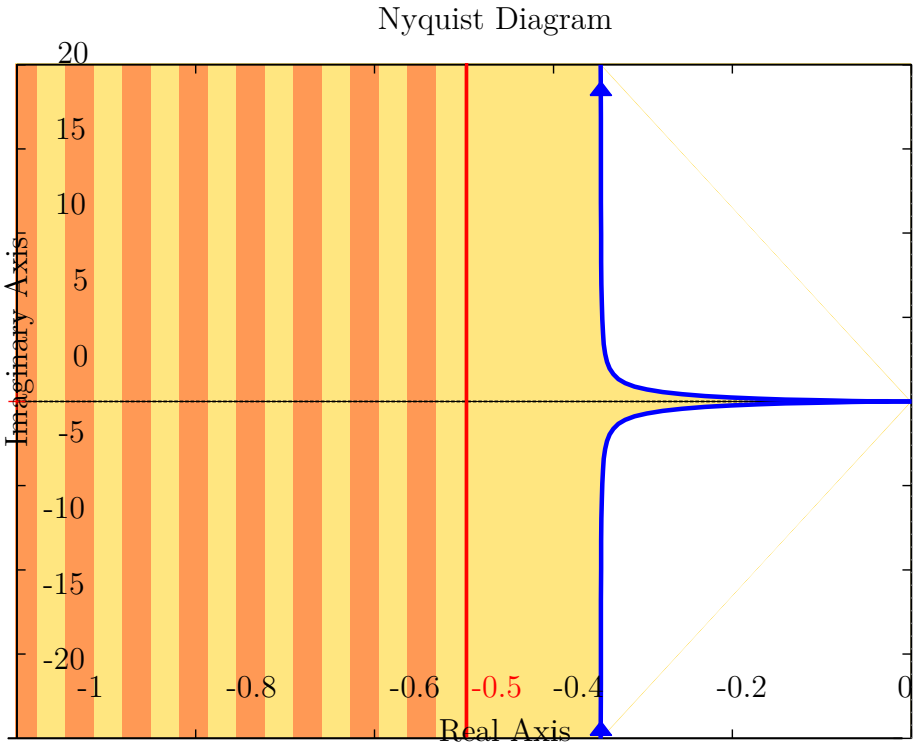


Figure A.18: Nyquist plot of the forward chain $P(s)K(s)$ (blue), "allowed" region for $-1/\lambda_i$ (yellow) and possible locations of $-1/\lambda_i$ (orange)

around $L_{\lambda_i}(s)$ to be stable. Rescaling the complex plane, it is easy to see that this is equivalent to the Nyquist plot of $L(s) = P(s)K(s)$ not encircling the point $-1/\lambda_i$. With respect to the Nyqist plot shown in Fig. A.18, the points $-1/\lambda_i$ are accordingly allowed to be located in the region left of the Nyquist plot, the points $1/\lambda_i$ accordingly in the image of this region mirrored along the imaginary axis.

iii) For any point λ in the Perron disk, we can generally write

$$\begin{aligned}
\lambda_i &= 1 - \beta e^{j\alpha_i} = 1 - \beta cos(\alpha_i) - j\beta sin(\alpha_i), \quad \beta \in [0,1] \\
\Rightarrow -\frac{1}{\lambda_i} &= -\frac{1}{1 - \beta cos(\alpha_i) - j\beta sin(\alpha_i)} = -\frac{1 - \beta cos(\alpha_i) + j\beta sin(\alpha_i)}{(1 - \beta cos(\alpha_i))^2 + \beta^2 sin^2(\alpha_i)} \\
&= -\frac{1 - \beta cos(\alpha_i) + j\beta sin(\alpha_i)}{1 - 2\beta cos(\alpha_i) + \beta^2 \underbrace{(cos^2(\alpha_i) + sin^2(\alpha_i))}_{=1}} \\
&= -\frac{1 - \beta cos(\alpha_i)}{(1 + \beta^2) - 2\beta cos(\alpha_i)} - j\frac{\beta sin(\alpha_i)}{(1 + \beta^2) - 2\beta cos(\alpha_i)}
\end{aligned}$$

In the worst case, meaning λ_i located on the boundary, we have $\beta = 1$, with which the real part of $1/\lambda_i$ becomes $-1/2$. Thus, the boundary of the Perron disk is mapped to a parallel of the imaginary axis located at -0.5. For $Re(-1/\lambda_i)$, this case marks an upper bound, accordingly $1/\lambda_i$ are always located on or

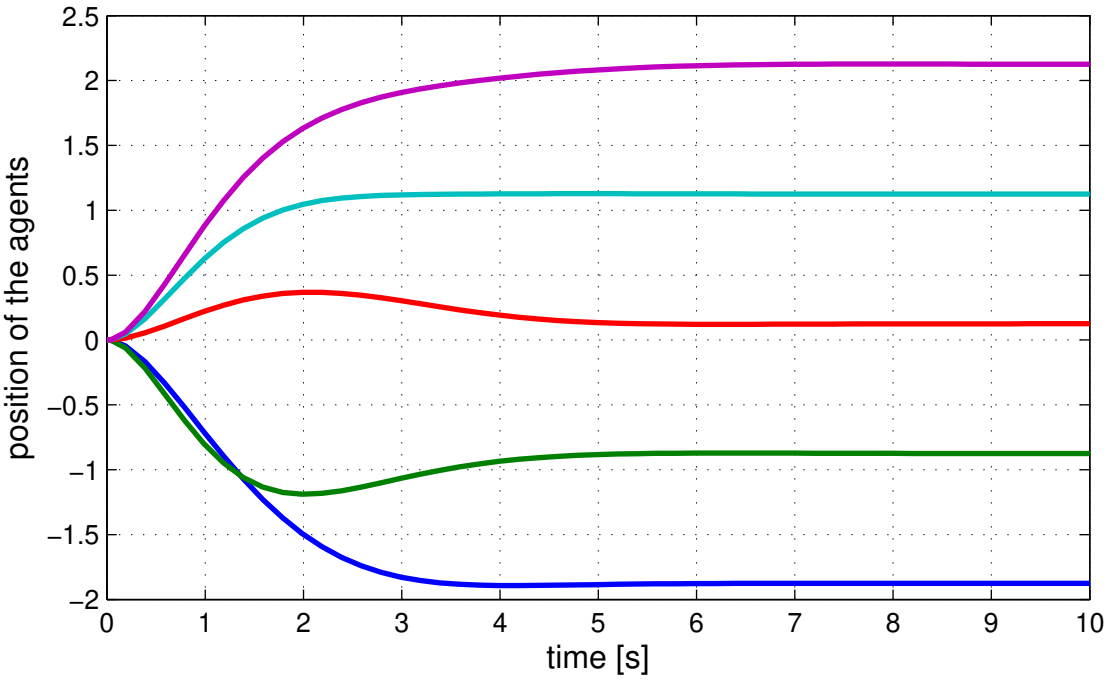


Figure A.19: Simulation results for the formation

left of the -0.5 vertical line. This area is entirely within the ”allowed” region determined in (ii), thus the given MAS is stable for all topologies.

d) From the simulation of the given formation, the results shown in Fig. A.19 were generated. As one can see, the agents reach the desired formation, meaning the desired relative distances among each other. However, with the architecture used here it is not possible to control the absolute position of the formation. For this reason the actual formation is shifted w.r.t. the reference values by some offset.

The reason for this behavior is that the controller only has access to the formation control error

$$e = L(r - y) = Lr - Ly \tag{A.23}$$

$$e_i = \frac{1}{|N|}\left(\sum_j (r_i - r_j) - \sum_j (y_i - y_j)\right), \tag{A.24}$$

which depends on the reference distances between the agents, but not on the absolute reference positions.

Problem 6.3 *(Design of a Formation Controller)*

a) A linear model describing the dynamic behavior of a quad-rotor helicopter is given in [29] and Appendix G as a state space model

$$\begin{aligned} \dot{\xi} &= A\xi + Bu \\ y &= C\xi. \end{aligned} \tag{A.25}$$

Unlike in the previous exercises, here input and output of an agent are vector signals. Nevertheless, the equivalence proven in Problem 6.1 also holds in this case: The formation is stable iff the controller $K(s)$ stabilizes all systems

$$\begin{aligned} \dot{\xi}_i &= A\xi_i + Bu_i \\ v_i &= \lambda_i C\xi_i \end{aligned} \tag{A.26}$$

for all eigenvalues λ_i of the Laplacian of the communication topology.

b) Using $\lambda_i = 1 + \delta_i$, the systems (A.26) can be rewritten as

$$\dot{\xi}_i = A\xi_i + Bu_i \tag{A.27}$$

$$v_i = C\xi_i + \delta_i C\xi_i. \tag{A.28}$$

Factorizing $C = D_\delta C_\delta$ (e.g. using singular value decomposition), the second term of (A.28) has a form that allows "pulling out the delta", which yields

$$\begin{aligned} \dot{\xi}_i &= A\xi_i + Bu_i \\ z_\delta &= C_\delta \xi_i \\ v_i &= C\xi_i + D_\delta w_\delta \\ w_\delta &= \underbrace{\delta_i I}_{\Delta} z_\delta. \end{aligned} \tag{A.29}$$

Due to the properties of the Laplacian, we already know $|\delta_i| < 1$, with which the condition $\|\Delta\|_\infty < 1$ for using the small gain theorem is automatically fulfilled.

c) i) A generalized plant for H_∞ formation controller design is shown in Fig. A.20. For the shaping filters, a suitable choice is

$$W_S(s) = \frac{\omega_S}{M_S} \cdot \frac{1}{s + \omega_S} I_3 \quad W_K(s) = \frac{c_K}{M_K} \cdot \frac{s + \omega_K}{s + c_K \omega_K} I_3. \tag{A.30}$$

ii) The H_∞ controller synthesis can be easily done using the function `hinfsyn` of the Robust Control Toolbox, see the MATLAB file for more details. Suitable values of the shaping filter parameters are

ω_S	M_S	ω_K	M_K	c_K
0.001	0.001	1000	10	1000

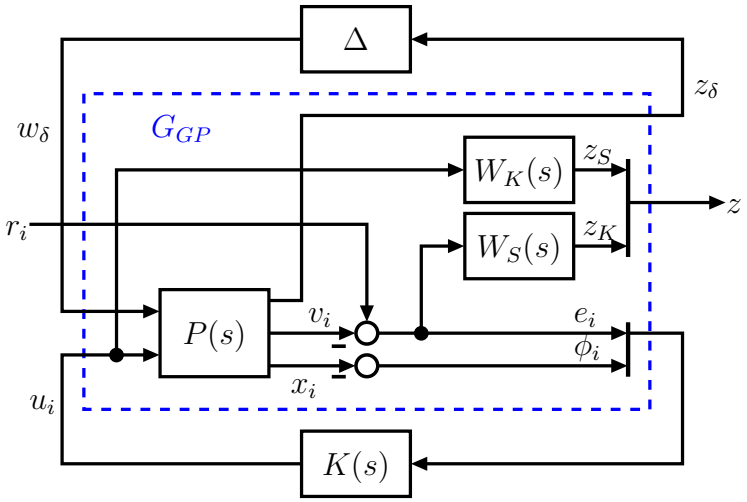


Figure A.20: Generalized plant for H_∞ formation controller design

d) The simulated positions in x-direction of the quadrocopters are shown in Fig. A.21. In this case, a topology was assumed with the Laplacian

$$L = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ -1/2 & -1/2 & 1 & 0 \\ -1/2 & 0 & -1/2 & 1 \end{bmatrix}.$$

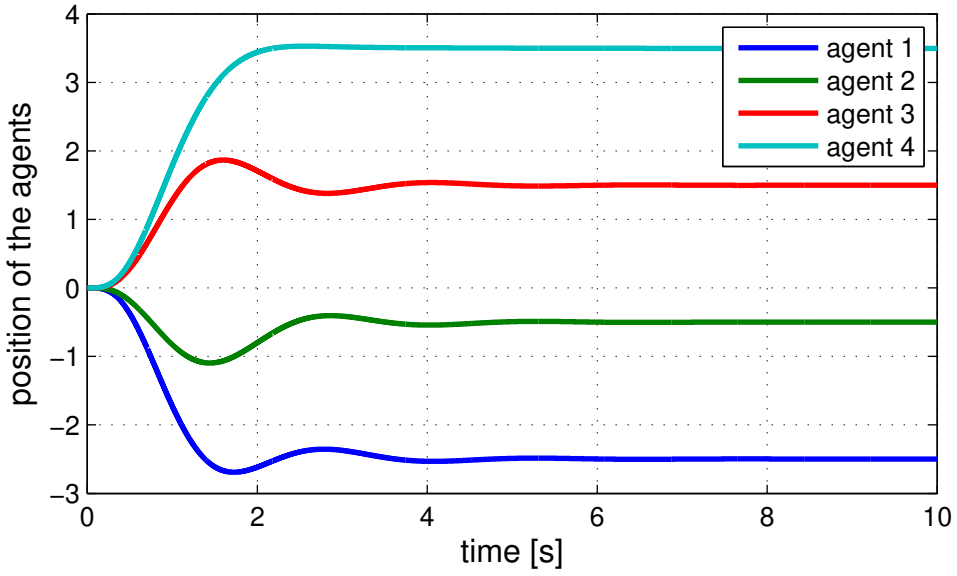


Figure A.21: Positions of the agents in x-direction

As one can see, the relative positioning of the agents meets the reference, whereas the absolute position of the formation is not controlled.

Problem 6.4 *(Information Flow Filter for Formation Control)*

a) i) According to Fig. 6.14 and with $\hat{R}(z) = (I_N q + \hat{F}(z))^{-1}\hat{F}(z)$, we can write

$$\begin{aligned}
p &= \hat{R}\left(L_{(q)}(r-(p-y)-y)+(p-y)+y\right) = \hat{R}L_{(q)}(r-p)+\hat{R}p \\
&= \hat{R}L_{(q)}r + \hat{R}(I-L_{(q)})p \\
\Rightarrow &(I-\hat{R}+\hat{R}L_{(q)})p = \hat{R}L_{(q)}r \\
\Rightarrow &\left(I-(I+\hat{F})^{-1}\hat{F}+(I+\hat{F})^{-1}\hat{F}L_{(q)}\right)p = (I+\hat{F})^{-1}\hat{F}L_{(q)}r \\
\Rightarrow &\left(I+\hat{F}-\hat{F}+\hat{F}L_{(q)}\right)p = \hat{F}L_{(q)}r \\
\Rightarrow &p = \underbrace{(I+\hat{F}L_{(q)})^{-1}\hat{F}L_{(q)}}_{G_{pr}}r
\end{aligned}$$

ii) For the transfer function from p to y we obtain

$$y = \hat{P}\hat{K}(p-y) = \hat{P}\hat{K}p - \hat{P}\hat{K}y \tag{A.31}$$

$$\Rightarrow (I+\hat{P}\hat{K})y = \hat{P}\hat{K}p \tag{A.32}$$

$$\Rightarrow y = \underbrace{(I+\hat{P}\hat{K})^{-1}\hat{P}\hat{K}}_{G_{yp}}p \tag{A.33}$$

b) Based on the rule "Loop gain = forward gain by 1 - loop gain", for a loop with negative feedback around a system $G(s)$ generally $G_{loop}(s) = (I+G(s))^{-1}G(s)$ holds. With this formula, G_{pr} can be identified as a negative feedback loop around a series connection of $L_{(q)}$ and $\hat{F}(s)$. Likewise, G_{yp} is identified as a loop around a series of $\hat{K}(s)$ and $\hat{P}(s)$. This is exactly what is shown in Fig. 6.15.

From the structure of this setup some important observations can be made: The two loops are only connected by a feed-forward path, thus they can be treated as independent systems. With the position y as feedback signal, the loop on the right acts as a position control loop with the signal p as reference input. Accordingly, p can be interpreted as an estimated absolute reference position for each agent. As all components of the right loop are block-diagonal, it can be decomposed into independent systems for each agent. Conversely, the loop on the left depends on the topology, but not on the agent dynamics. With p as output it can be interpreted as a consensus algorithm to estimate the agents' reference position.

c) For an information flow filter $F = \left[\begin{array}{c|c} A_F & B_F \\ \hline C_F & 0 \end{array}\right]$, the closed information flow loop is obtained as

$$\begin{aligned}
\dot{x} &= (\hat{A}_F - \hat{B}_F\hat{C}_F L_{(n)})x + \hat{B}_F L r \\
p &= \hat{C}_F x.
\end{aligned} \tag{A.34}$$

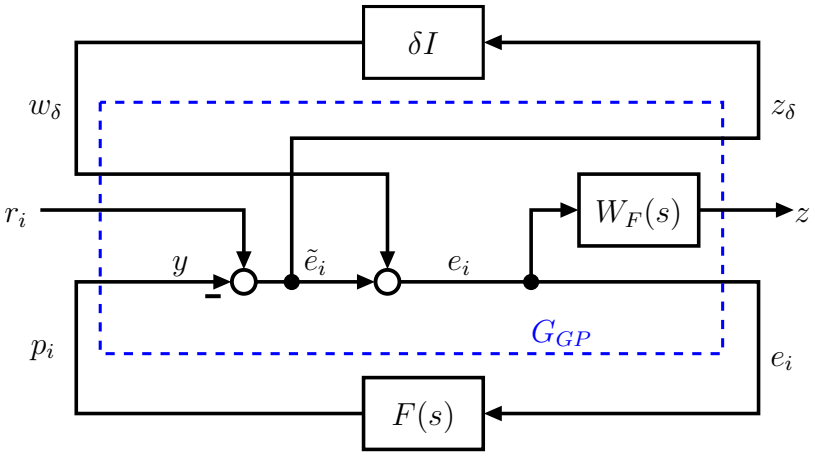


Figure A.22: Generalized plant for information flow filter design

In the same manner as in Problem 6.1, we can use the Schur transformation on the state equation and achieve

$$\dot{x} = (\hat{A}_F - \hat{B}_F \hat{C}_F U_{(n)})x + \hat{B}_F U r. \tag{A.35}$$

Accordingly, the stability of this loop is equivalent to the stability of a single system

$$\dot{x}_i = (A_F - \lambda_i B_F C_F)x_i + \lambda_i B_F r_i \tag{A.36}$$

for all $\lambda_i \in eig(L)$. With $\lambda_i = 1 + \delta$ and $C_F = D_\delta C_\delta = I \cdot C_F$, we can rewrite this as

$$\dot{x}_i = (A_F - B_F C_F - \delta B_F C_F)x_i + B_F r_i + \delta B_F r_i \tag{A.37}$$

and "pull out" the factor δ as uncertainty:

$$\begin{aligned} \dot{x}_i &= (A_F - B_F C_F)x_i + B_F w_\delta + B_F r_i \\ z_\delta &= -C_F x_i + r_i \\ w_\delta &= \delta I z_\delta. \end{aligned} \tag{A.38}$$

d) A generalized plant to design the information flow filter is shown in Fig. A.22. The shaping filter can be suitably chosen as

$$W_F(s) = \frac{\omega_F}{M_F} \cdot \frac{1}{s + \omega_F}. \tag{A.39}$$

e) The synthesis of the information flow filter can be done using the command "hinfsyn", for testing the information flow loop can be implemented in Simulink. A design with the values was used to generate the results shown in Fig. A.23, adopting the topology from Problem 6.2. When the reference switches from zero to $\begin{bmatrix}1 & 2 & 3 & 4 & 5\end{bmatrix}^T$, the agents exchange their information and reach a consensus about every agent's desired position within one second. Note that the estimated positions are equal to the positions actually reached by the cooperative formation controller in Problem 6.2.

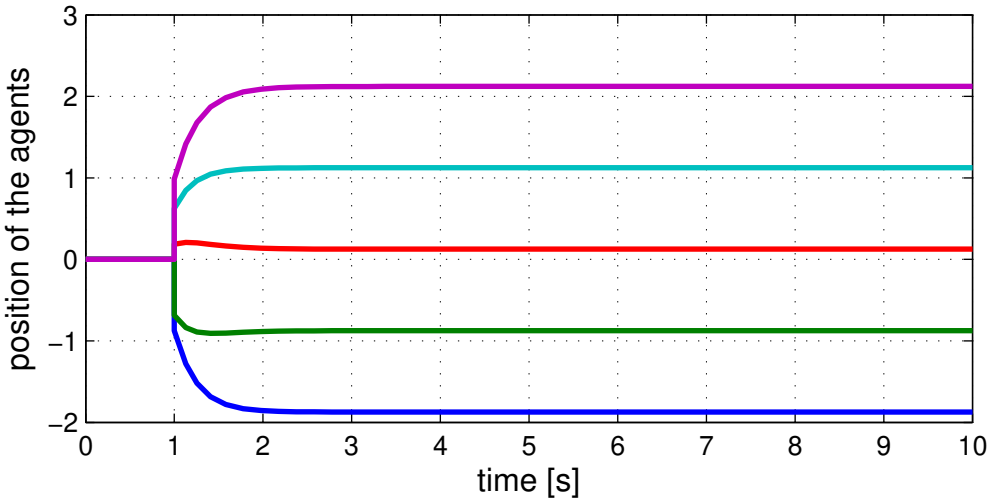
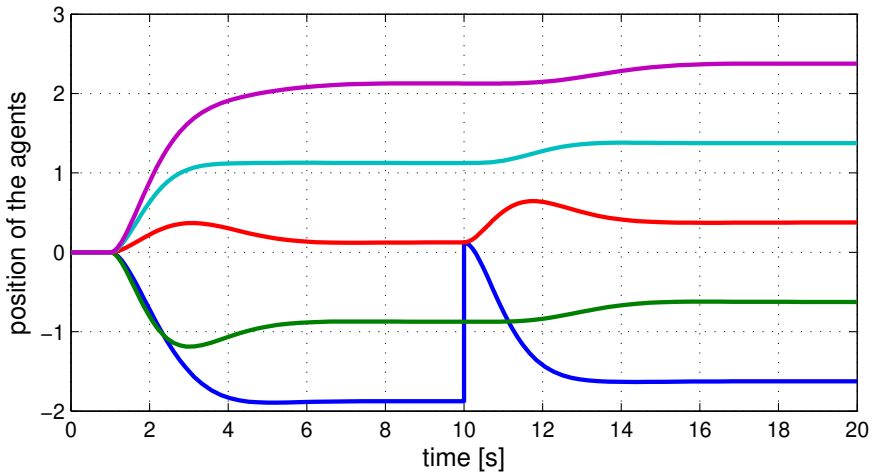
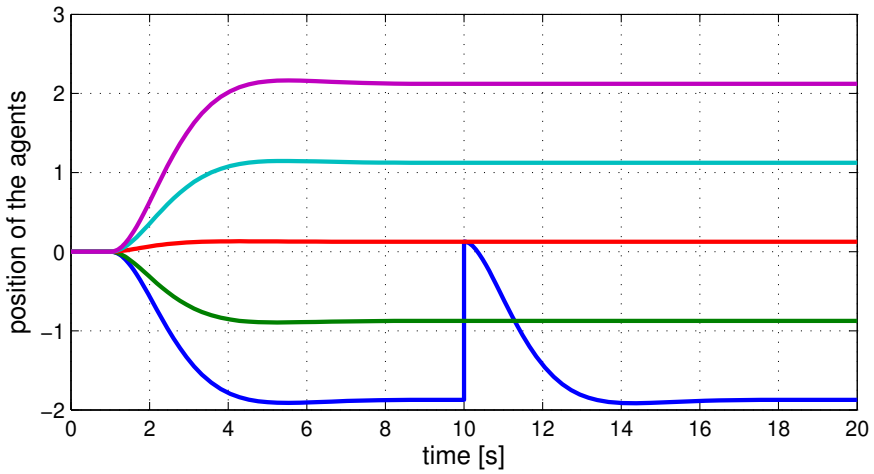


Figure A.23: Estimation values computed by the information flow loop

Problem 6.5 *(Disturbance Rejection in Formation Control)*



(a) cooperative formation controller from Problem 6.2



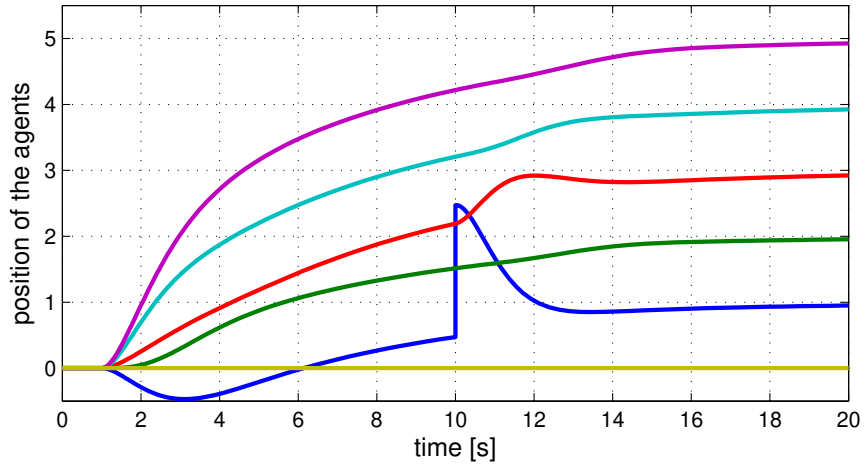
(b) information flow filter scheme, plant output y

Figure A.24: Comparison of cooperative controller and information flow scheme

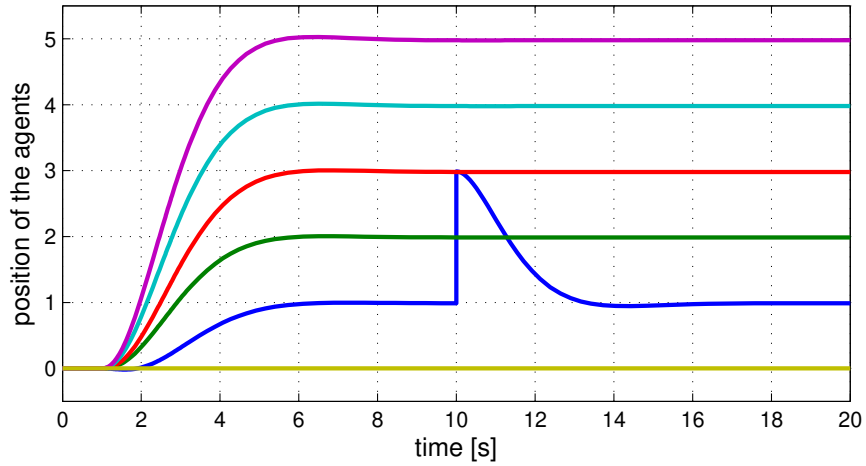
a,b) Simulation results from both control schemes are shown in Fig. A.24. The major difference to be observed here is that in case of the information flow filter only the directly disturbed agent is affected by the disturbance, whereas the other agents remain at their position. In contrast, in case of the cooperative control scheme, all other agents react on the disturbed agent in different amounts. Note that this leads to a different final position of the formation.

c) From the information flow filter scheme shown in Fig. 6.15, it is visible that there is no feedback connection from the plant output y to the information flow loop. Accordingly, the information flow loop is unable to react on any disturbances acting

on the plant. The position controller is indeed able to react on the disturbance (as visible in the reaction of the disturbed agent), but it does not contain any interaction between the agents. In contrast, the cooperative controller both interacts with the neighbor agents and has access to their real output data.



(a) cooperative formation controller from Problem 6.3



(b) information flow filter scheme, plant output y

Figure A.25: Results achieved with the modified topology

d) Results generated with the modified topology are shown in Fig. A.25, where the yellow curve belongs to the new agent with fixed position. Whereas in previous results only the relative positions matched with the reference, here also the absolute positions match. The reason is that the control schemes are only able to control the relative position, but in case of a fixed agent all relative positions are controlled with respect to this position. Accordingly, the fixed agent acts as a leader of the formation and defines its absolute position. Note that in this scenario the information flow filter shows a significant performance benefit w/r/t the cooperative scheme.

Problem 6.6 *(Scheduled State Feedback Synthesis)*

a) The state equation for a group of five agents is

$$\begin{bmatrix}\dot{x}\\ q\end{bmatrix} = \left[\begin{array}{c|c|c} I_5 \otimes A & I_5 \otimes B_p & I_5 \otimes B_u \\ \hline I_5 \otimes C_q & 0 & 0 \end{array}\right] \begin{bmatrix} x \\ p \\ u \end{bmatrix}, \quad p = \begin{bmatrix} \Theta^1 & & & & \\ & \Theta^2 & & & \\ & & \Theta^3 & & \\ & & & \Theta^4 & \\ & & & & \Theta^5 \end{bmatrix} q\,.$$

b) The block $\hat{P}$ is described by the state equation of the previous task. The block $\hat{F}$ is described by $F = I_5 \otimes F_0 + \Theta\,(I_5 \otimes F_1)$, thus

$$\begin{aligned} u &= \left(I_5 \otimes F_0 + \Theta\,(I_5 \otimes F_1)\,(L \otimes I_2)\right)(r + x) \\ &= \left(L \otimes F_0 + \Theta\,(L \otimes F_1)\right)(w + x)\,. \end{aligned}$$

If this control input is substituted in the open-loop state equation from the previous task, we get

$$\begin{aligned} \dot{x} =& (I_5 \otimes A)x + (I_5 \otimes B_p)p + (I_5 \otimes B_u)u \\ =& (I_5 \otimes A)x + (I_5 \otimes B_p)p + (I_5 \otimes B_u)\left(L \otimes F_0 + \Theta\,(L \otimes F_1)\right)(w + x) \\ =& \left(I_5 \otimes A + L \otimes B_u F_0 + (I_5 \otimes B_u)\Theta\,(L \otimes F_1)\right)x + (I_5 \otimes B_p)p \\ &+ \left(L \otimes B_u F_0 + (I_5 \otimes B_u)\Theta\,(L \otimes F_1)\right)w\,. \end{aligned}$$

Through the control input the uncertainty Θ appeared again, thus the LFT channel has to be adapted, leading to

$$\begin{bmatrix}\dot{x}\\ \bar{q}\end{bmatrix} = \left[\begin{array}{c|c|c} I_5 \otimes A + L \otimes B_u F_0 & I_5 \otimes B_p \quad I_5 \otimes B_u & L \otimes B_u F_0 \\ \hline I_5 \otimes C_q & 0 \quad 0 & 0 \\ L \otimes F_1 & 0 \quad 0 & L \otimes F_1 \end{array}\right] \begin{bmatrix} x \\ \bar{p} \\ w \end{bmatrix}, \quad \bar{p} = \begin{bmatrix} \Theta & \\ & \Theta \end{bmatrix} \bar{q}\,.$$

c) i) The closed-loop systems with included weighting filters is shown in Figure A.26.

ii) The plots of $1/|W_s|$ and $1/|W_k|$ are shown in Figure A.27. To achieve good tracking in position and velocity, $1/|W_s|$ should have positive slope of 40dB/dec at small frequencies. Since the filter needs to be stable, a pure double integrator is not possible, therefore two poles with frequency 0.1 are chosen, with lower magnitude than the system poles, whose frequency is around 1. The control sensitivity should roll off at higher frequencies for a reasonable control effort. This frequency is chosen here to 1. To have a proper filter an additional pole is added at very high frequency, here chosen to 1000.

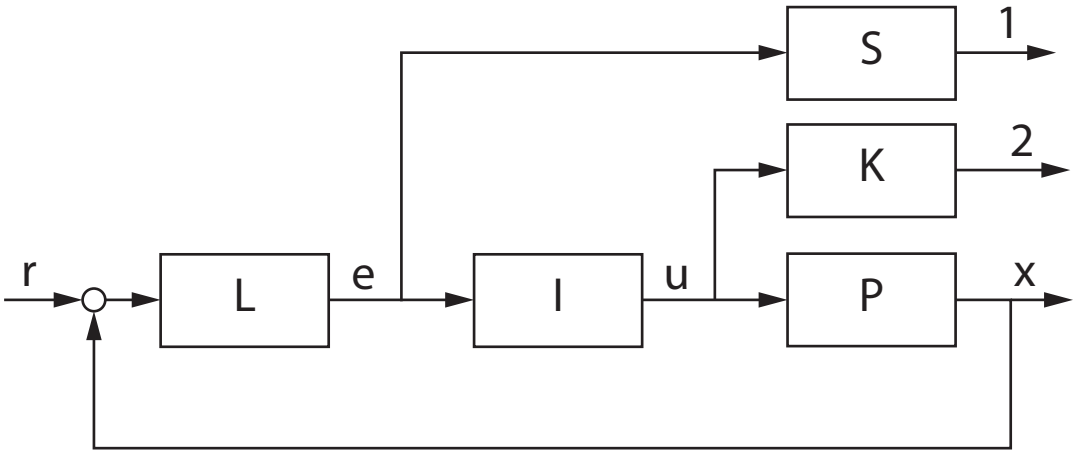


Figure A.26: Generalized plant for distributed state feedback control

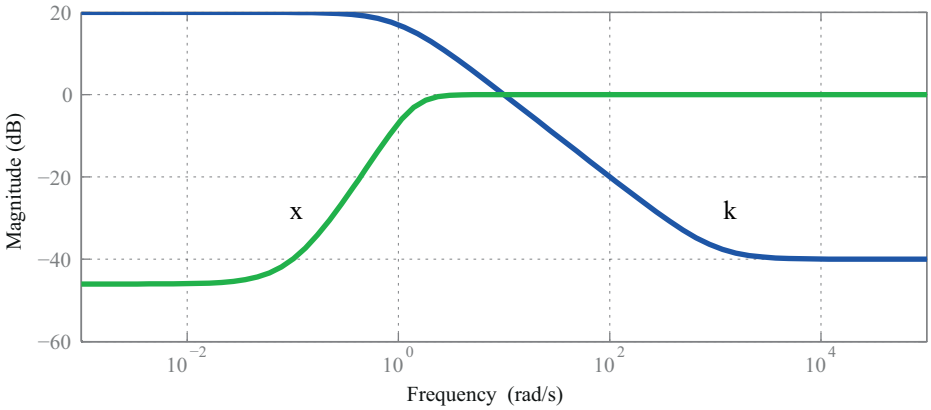


Figure A.27: Magnitude plot of the inverse filter functions

iii) One possibility for the state space equations of the filters are calculated by Matlab to

$$\begin{bmatrix} \dot{x}_s^k \\ z_1 \end{bmatrix} = \left[\begin{array}{cc|c} -0.2 & -0.08 & 4 \\ 0.125 & 0 & 0 \\ \hline 0.45 & 3.98 & 1 \end{array}\right] \begin{bmatrix} x_s^k \\ e^k \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \dot{x}_k^k \\ z_2 \end{bmatrix} = \left[\begin{array}{cc|c} -1000 & 256 \\ -390.2 & 100 \end{array}\right] \begin{bmatrix} x_k^k \\ u^k \end{bmatrix} .$$

The generalized system including the filters is then described by

$$\begin{bmatrix} \dot{x} \\ \dot{x}_s \\ \dot{x}_k \\ \dot{\bar{q}} \\ z_1 \\ z_2 \end{bmatrix} = \left[\begin{array}{ccc|cc|c} I_5 \otimes A + L \otimes B_u F_0 & 0 & 0 & I_5 \otimes B_p I_5 \otimes B_u & & L \otimes B_u F_0 \\ L \otimes (I_2 \otimes B_s) & I_{10} \otimes A_s & 0 & 0 & 0 & L \otimes (I_2 \otimes B_s) \\ L \otimes (I_2 \otimes B_k) F_0 & 0 & I_{10} \otimes A_k & 0 & I_{10} \otimes B_k & L \otimes (I_2 \otimes B_k) F_0 \\ \hline I_5 \otimes C_q & 0 & 0 & 0 & 0 & 0 \\ L \otimes F_1 & 0 & 0 & 0 & 0 & L \otimes F_1 \\ \hline 0 & I_{10} \otimes C_s & 0 & 0 & 0 & L \otimes (I_2 \otimes D_s) \\ 0 & 0 & I_{10} \otimes C_k & 0 & I_{10} \otimes D_k & L \otimes (I_2 \otimes D_k) F_0 \end{array}\right] \begin{bmatrix} x \\ x_s \\ x_k \\ \bar{p} \\ w \end{bmatrix},$$

$$\bar{p} = \begin{bmatrix} \Theta & \\ & \Theta \end{bmatrix} \bar{q}.$$

d) The group of five agents is stable and achieves an induced $\mathcal{L}_2$-gain less than γ if there exist a $\bar{P} = \bar{P}^T > 0$, $\bar{Q} = \bar{Q}^T < 0$, $\bar{R} = \bar{R}^T$ and $\bar{S}$, such that

$$\left[*\right]^T \left[\begin{array}{cc:cc:c} 0 & \bar{P} & & & \\ \bar{P} & 0 & & & \\ \hdashline & & \bar{Q} & \bar{S} & \\ & & \bar{S}^T & \bar{R} & \\ \hdashline & & & & -\frac{1}{\gamma^2} I \\ & & & & & I \end{array}\right] \left[\begin{array}{ccc} I & 0 & 0 \\ \bar{\mathbf{A}} & \bar{\mathbf{B}}_p & \bar{\mathbf{B}}_{\mathbf{w}} \\ \hdashline 0 & I & 0 \\ \bar{\mathbf{C}}_q & \bar{\mathbf{D}}_{qq} & \bar{\mathbf{D}}_{qw} \\ \hdashline 0 & 0 & I \\ \bar{\mathbf{C}}_z & \bar{\mathbf{D}}_{zq} & \bar{\mathbf{D}}_{zw} \end{array}\right],$$

$$\left[\begin{array}{c} * \\ * \end{array}\right]^\top \left[\begin{array}{cc} \bar{Q} & \bar{S} \\ \bar{S}^T & \bar{R} \end{array}\right] \left[\begin{array}{c} \bar{\Theta} \\ I \end{array}\right] > 0, \ \forall \bar{\Theta} \in \bar{\mathbf{\Theta}}.$$

e) In the following the first condition of the full-block S-procedure is called main condition and is abbreviated as $M^T \Psi M < 0$. The results of this part are similar to [28].

i) The transformation such that $\tilde{M}$ contains only block diagonal matrices is given by

$$Z_2 = \left[\begin{array}{ccc:ccc:c} S \otimes I_2 & & & & & & \\ & S \otimes I_4 & & & & & \\ & & S \otimes I_2 & & & & \\ \hdashline & & & S \otimes I_2 & & & \\ & & & & S \otimes I_2 & & \\ \hdashline & & & & & S \otimes I_2 & \\ & & & & & & S \otimes I_2 \end{array}\right] \quad \text{and}$$

$$Z_1 = \left[\begin{array}{ccccccc:ccccccc} S \otimes I_2 & & & & & & & & & & & & & \\ & S \otimes I_4 & & & & & & & & & & & & \\ & & S \otimes I_2 & & & & & & & & & & & \\ & & & S \otimes I_2 & & & & & & & & & & \\ & & & & S \otimes I_4 & & & & & & & & & \\ & & & & & S \otimes I_2 & & & & & & & & \\ \hdashline & & & & & & S \otimes I_2 & & & & & & & \\ & & & & & & & S \otimes I_2 & & & & & & \\ & & & & & & & & S \otimes I_2 & & & & & \\ & & & & & & & & & S \otimes I_2 & & & & \\ & & & & & & & & & & S \otimes I_2 & & & \\ \hdashline & & & & & & & & & & & S \otimes I_2 & & \\ & & & & & & & & & & & & S \otimes I_2 & \\ & & & & & & & & & & & & & S \otimes I_2 \end{array}\right].$$

It is shown exemplarily how $\hat{\mathbf{A}}$ is diagonalized

$$
\begin{aligned}
&\begin{bmatrix} S\otimes I_2 & & \\ & S\otimes I_4 & \\ & & S\otimes I_2 \end{bmatrix}^{-1}\begin{bmatrix} I_5\otimes A + L\otimes B_u F_0 & 0 & 0 \\ L\otimes(I_2\otimes B_s) & I_{10}\otimes A_s & 0 \\ L\otimes(I_2\otimes B_k)F_0 & 0 & I_{10}\otimes A_k \end{bmatrix}\begin{bmatrix} S\otimes I_2 & & \\ & S\otimes I_4 & \\ & & S\otimes I_2 \end{bmatrix} \\
=&\begin{bmatrix} S\otimes I_2 & & \\ & S\otimes I_4 & \\ & & S\otimes I_2 \end{bmatrix}^{-1}\begin{bmatrix} S\otimes A + LS\otimes B_u F_0 & 0 & 0 \\ LS\otimes(I_2\otimes B_s) & S\otimes(I_2\otimes A_s) & 0 \\ LS\otimes(I_2\otimes B_k)F_0 & 0 & S\otimes(I_2\otimes A_k) \end{bmatrix} \\
=&\begin{bmatrix} S^{-1}S\otimes A + S^{-1}LS\otimes B_u F_0 & 0 & 0 \\ S^{-1}LS\otimes(I_2\otimes B_s) & S^{-1}S\otimes(I_2\otimes A_s) & 0 \\ S^{-1}LS\otimes(I_2\otimes B_k)F_0 & 0 & S^{-1}S\otimes(I_2\otimes A_k) \end{bmatrix} \\
=&\begin{bmatrix} I_5\otimes A + \Lambda\otimes B_u F_0 & 0 & 0 \\ \Lambda\otimes(I_2\otimes B_s) & I_5\otimes(I_2\otimes A_s) & 0 \\ \Lambda\otimes(I_2\otimes B_k)F_0 & 0 & I_5\otimes(I_2\otimes A_k) \end{bmatrix}
\end{aligned}
$$

ii) If L is symmetric its diagonalizing transformation $\Lambda = S^{-1}LS$ is orthogonal, thus Z_1 and Z_2 are orthogonal. Thus $Z_1^T = Z_1^{-1}$ and $Z_2^T = Z_2^{-1}$.

iii) The Lyapunov matrices and the multiplier matrices must be structured such that Ψ commutes with Z_1. Therefore they have to be partitioned as the transformation matrix as

$$
\begin{aligned}
\bar{P} &= \begin{bmatrix} I_5\otimes P_1 & & \\ & I_5\otimes P_2 & \\ & & I_5\otimes P_3 \end{bmatrix}, \quad \text{with} \quad P_1, P_3 \in \mathbb{R}^{2\times 2},\, P_2 \in \mathbb{R}^{4\times 4}, \\
\bar{Q} &= \begin{bmatrix} I_5\otimes Q_1 & \\ & I_5\otimes Q_2 \end{bmatrix}, \quad \text{with} \quad Q_1, Q_2 \in \mathbb{R}^{2\times 2}
\end{aligned}
$$

and $\bar{R}$ and $\bar{S}$ accordingly.

iv) The transformed main condition only contains block diagonal matrices. The out multiplied condition can be permuted to get a block diagonal LMI, with the blocks only differing in λ. Thus to fulfill the block diagonal LMI it suffices to fulfill one block for all possible values of λ. Since we use the normalized Laplacian here, we know that $\lambda \in [0, 2]$. Thus the main condition results in

$$
\left[*\right]^T
\left[\begin{array}{c:c:c}
\begin{bmatrix} 0 & \begin{bmatrix} P_1 & & \\ & P_2 & \\ & & P_3 \end{bmatrix} \end{bmatrix} & & \\
\hdashline
\begin{bmatrix} \begin{bmatrix} P_1 & & \\ & P_2 & \\ & & P_3 \end{bmatrix} & 0 \end{bmatrix} & \begin{bmatrix} Q_1 & S_1 \\ & Q_2 & & S_2 \end{bmatrix} & \\
& \begin{bmatrix} S_1^T & R_1 \\ & S_2^T & & R_2 \end{bmatrix} & \\
\hdashline
& & -\frac{1}{\gamma^2} I \\
& & I
\end{array}\right]
\left[\begin{array}{c:c:c}
I & 0 & 0 \\
\begin{bmatrix} A+\lambda B_u F_0 & 0 & 0 \\ \lambda(I_2 \otimes B_s) & I_2 \otimes A_s & 0 \\ \lambda(I_2 \otimes B_k) F_0 & 0 & I_2 \otimes A_k \end{bmatrix} & \begin{bmatrix} B_p & B_u \\ 0 & 0 \\ 0 & I_2 \otimes B_k \end{bmatrix} & \begin{bmatrix} B_u F_0 \\ I_2 \otimes B_s \\ (I_2 \otimes B_k) F_0 \end{bmatrix} \\
\hdashline
0 & I & 0 \\
\hdashline
\begin{bmatrix} C_q & 0 & 0 \\ \lambda F_1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ \lambda F_1 \end{bmatrix} \\
\hdashline
0 & 0 & I \\
\hdashline
\begin{bmatrix} 0 & I_2 \otimes C_s & 0 \\ 0 & 0 & I_2 \otimes C_k \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & I_2 \otimes D_k \end{bmatrix} & \begin{bmatrix} \lambda(I_2 \otimes D_s) \\ (I_2 \otimes D_k) F_0 \end{bmatrix}
\end{array}\right],
$$

$$
\forall \lambda \in [0,2]\,,
$$

$$
\begin{bmatrix} * \\ * \end{bmatrix}^\top
\begin{bmatrix} \begin{bmatrix} Q_1 & & \\ & Q_2 \end{bmatrix} & \begin{bmatrix} S_1 & & \\ & S_2 \end{bmatrix} \\ \begin{bmatrix} S_1^T & & \\ & S_2^T \end{bmatrix} & \begin{bmatrix} R_1 & & \\ & R_2 \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \begin{bmatrix} \Theta & & \\ & \Theta & \\ & & I \end{bmatrix} \end{bmatrix} > 0,\ \forall \bar{\Theta} \in \bar{\mathbf{\Theta}}\,.
$$

f) The congruence transformation T to eliminate products of controller and Lyapunov matrix in the main condition of the size of one agent derived in the previous task, which we will call $\tilde{M}\tilde{PsiM} > 0$ in the following, is given by

$$
T = \begin{bmatrix}
P_1^{-1} & & & & & \\
& P_2^{-1} & & & & \\
& & P_3^{-1} & & & \\
& & & I_2 & & \\
& & & & I_2 & \\
& & & & & P_1^{-1}
\end{bmatrix}
$$

resulting in a transformed system with the transformed Lyapunov variables $Y_1 = P_1^{-1}$, $Y_2 = P_2^{-1}$, $Y_3 = P_3^{-1}$ and the new controller variables $D_0 = F_0 Y_1$ and $D_1 = F_1 Y_1$.

g) The congruence transformation of the previous task let to the transformed conditions

$$
\left[*\right]^T
\left[\begin{array}{c:c:c}
\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} & & \\
\hdashline
\begin{bmatrix} \begin{bmatrix} Q_1 & & \\ & Q_2 \end{bmatrix} & \begin{bmatrix} S_1 & & \\ & S_2 \end{bmatrix} \\ \begin{bmatrix} S_1^T & & \\ & S_2^T \end{bmatrix} & \begin{bmatrix} R_1 & & \\ & R_2 \end{bmatrix} \end{bmatrix} & & \\
\hdashline
& -\frac{1}{\gamma^2} I & \\
& & I
\end{array}\right]
\left[\begin{array}{c:c:c}
I & 0 & 0 \\
\begin{bmatrix} AY_1 + \lambda B_u D_0 & 0 & 0 \\ \lambda(I_2 \otimes B_s) Y_1 & (I_2 \otimes A_s) Y_2 & 0 \\ \lambda(I_2 \otimes B_k) D_0 & 0 & (I_2 \otimes A_k) Y_3 \end{bmatrix} & \begin{bmatrix} B_p & B_u \\ 0 & 0 \\ 0 & I_2 \otimes B_k \end{bmatrix} & \begin{bmatrix} B_u D_0 \\ (I_2 \otimes B_s) Y_1 \\ (I_2 \otimes B_k) D_0 \end{bmatrix} \\
\hdashline
0 & I & 0 \\
\hdashline
\begin{bmatrix} C_q Y_1 & 0 & 0 \\ \lambda D_1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ \lambda D_1 \end{bmatrix} \\
\hdashline
0 & 0 & I \\
\hdashline
\begin{bmatrix} 0 & (I_2 \otimes C_s) Y_2 & 0 \\ 0 & 0 & (I_2 \otimes C_k) Y_3 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & I_2 \otimes D_k \end{bmatrix} & \begin{bmatrix} \lambda(I_2 \otimes D_s) Y_1 \\ (I_2 \otimes D_k) D_0 \end{bmatrix}
\end{array}\right],
$$

$$
\forall \lambda \in [0,2]\,,
$$

where the outer factors depend in the following way on the unknowns

$$\left[*\right]^T \left[\begin{array}{cc:cc:c} 0 & I & & & \\ I & 0 & & & \\ \hdashline & & Q & S & \\ & & S^T & R & \\ \hdashline & & & & -\frac{1}{\gamma^2} I \\ & & & & & I \end{array}\right] \left[\begin{array}{ccc} I & 0 & 0 \\ \mathbf{A}(Y_1, Y_2, Y_3, D_0) & \mathbf{B}_p & \mathbf{B}_w(Y_1, D_0) \\ \hdashline 0 & I & 0 \\ \mathbf{C}_q(Y_1, D_1) & 0 & \mathbf{D}_{qw}(D_1) \\ \hdashline 0 & 0 & I \\ \mathbf{C}_z(Y_2, Y_3) & \mathbf{D}_{zp} & \mathbf{D}_{zw}(Y_1, D_0) \end{array}\right], \quad \forall \lambda \in [0, 2]\,.$$

Although we could eliminate the products of Lyapunov matrix and controller matrices by the variable transformation, there are obviously still products of the multiplier matrices with Y_1 and D_1. This can be eliminated by the use of the dual version given by

$$\left[*\right]^T \left[\begin{array}{cc:cc:c} 0 & I & & & \\ I & 0 & & & \\ \hdashline & & \tilde{Q} & \tilde{S} & \\ & & \tilde{S}^T & \tilde{R} & \\ \hdashline & & & & -\gamma^2 I \\ & & & & & I \end{array}\right] \left[\begin{array}{ccc} -\mathbf{A}(Y_1, Y_2, Y_3, D_0)^T & -\mathbf{C}_q(Y_1, D_1)^T & -\mathbf{C}_z(Y_2, Y_3)^T \\ I & 0 & 0 \\ \hdashline -\mathbf{B}_p^T & 0 & -\mathbf{D}_{zp}^T \\ 0 & I & 0 \\ \hdashline -\mathbf{B}_w(Y_1, D_0)^T & -\mathbf{D}_{qw}(D_1)^T & -\mathbf{D}_{zw}(Y_1, D_0)^T \\ 0 & 0 & I \end{array}\right],$$

$$\forall \lambda \in [0, 2]\,,$$

where $\begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}^{-1}$. It is also necessary to use the dual multiplier condition as well, otherwise it would not be linear, because of the inverse appearing in the dual form. The dual multiplier condition is considered in the following task.

g) i) If we out multiply

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \begin{bmatrix} q & s \\ s & r \end{bmatrix} \begin{bmatrix} \delta \\ 1 \end{bmatrix} > 0, \ \forall \delta \in \boldsymbol{\delta}\,. \tag{A.40}$$

we get the quadratic inequality

$$f(\delta) = \delta^2 q + 2\delta s + r > 0$$

If q is negative, the parable open downwards, otherwise upwards. Only if it opens downwards and thus $q < 0$, $f(\delta_{\min}) > 0$ and $f(\delta_{\max}) > 0$ assures that $f(\delta) > 0$ for all $\delta \in [\delta_{\min}, \delta_{\max}]$ as illustrated in Figure A.28.

ii) The dual multiplier condition is The multiplier condition

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} \begin{bmatrix} I \\ -\Delta^T \end{bmatrix} < 0, \ \forall \Delta \in \boldsymbol{\Delta}$$

with $R > 0$.

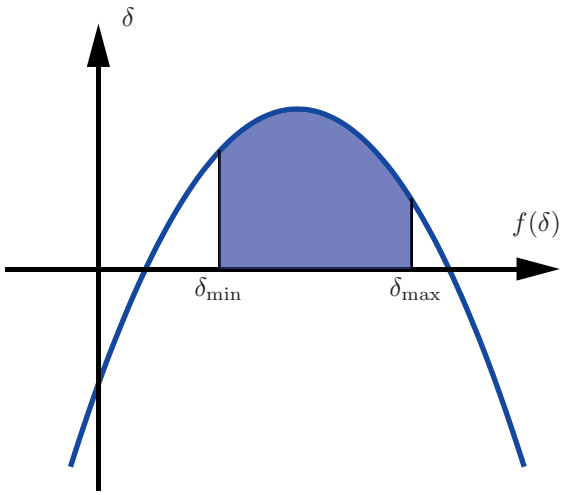


Figure A.28: Multiplier condition in the scalar case

Problem 6.7 *(LPV State Feedback Formation Control for Non-Holonomic Agents)*

a) The dynamics of the handle point can be obtained by finding the geometric position of (x_d, y_d), based on the coordinate system $(\mathbf{x}, \mathbf{y})$, and derivate it w.r.t time. Below are the dynamic equations of the handle.

$$\dot{x}_d = v\cos\phi - \omega d\sin\phi \tag{A.41a}$$
$$\dot{y}_d = v\sin\phi + \omega d\cos\phi \tag{A.41b}$$
$$\dot{\phi} = \omega \tag{A.41c}$$

The transformation T represents a time varying a change of coordinates, i.e. the *horizontal* direction of the disk is aligned towards the direction of motion. The transformed equations for the handle point (x_d, y_d), based on the coordinate system $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ are given by

$$\dot{\tilde{x}}_d = v + \omega\tilde{y}_d \tag{A.42a}$$
$$\dot{\tilde{y}}_d = -\omega\tilde{x}_d + \omega d \tag{A.42b}$$
$$\dot{\phi} = \omega \tag{A.42c}$$

A state space model is given by

$$\begin{bmatrix}\dot{x}_d\\ \dot{y}_d\\ \dot{\phi}\end{bmatrix} = \begin{bmatrix}0 & \omega & 0\\ -\omega & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\tilde{x}_d\\ \tilde{y}_d\\ \phi\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & d\\ 0 & 1\end{bmatrix}\begin{bmatrix}v\\ \omega\end{bmatrix} \tag{A.43}$$

An LFT representation of the above state space model requires to pull out the parameter ω. One can do this by hand, which is relatively easy for systems with small number of parameters or by employing Matlab tools which may find the most efficient representation in terms of parameter repetition. An LFT representation is given by:

$$\begin{bmatrix}\dot{x}^k\\ \dot{p}^k\\ \dot{y}^k\end{bmatrix} = \left[\begin{array}{ccc:cc:cc} 0 & 0 & 0 & 1 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 & d\\ 0 & 0 & 0 & 0 & 0 & 0 & 1\\ \hdashline 0 & 5 & 0 & 0 & 0 & 0 & 0\\ -5 & 0 & 0 & 0 & 0 & 0 & 0\\ \hdashline 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right]\begin{bmatrix}x^k\\ q^k\\ u^k\end{bmatrix} \qquad q^k = \Theta^k p^k \qquad \Theta^k = \omega^k I_2 \tag{A.44}$$

The transformation T eliminates the non-desirable trigonometric functions. Handling these as LPV parameters can introduce conservatism and potentially lead to non-feasible LMIs. A drawback of this transformation, is that all agents k are subject

to different orientations, i.e. all agents' positions are based on different coordinate systems.

The benefit of the handle becomes apparent in Eq. A.43. One can see that, if $d = 0$, then the influence of the input ω on $\tilde{y}_d$ is lost. Thus $d \neq 0$ increases the degree of controlability to the system.

b) The transformation matrices are necessary due to the coordinate system on which the communication occurs. It would not make sense to add/substract quantities in different coordinate systems. Thus, it is necessary to perform the communication in the same reference system, by using $\hat{T}$. However, the controller $\hat{F}(\theta)$ assumes that the information it receives it is on the transformed coordinate system, thus $\hat{T}^{-1}$ is necessary after the communication.

Rearrange the block diagram, to obtain $L_T = \hat{T}^{-1} L_{(3)} \hat{T}$. Notice that this matrices do not commute, which would be nice. The matrix T is an orthogonal matrix, i.e. $T^{-1} = T^{\mathrm{T}}$. That yields $L_T = \hat{T}^{\mathrm{T}} L_{(3)} \hat{T}$, which is symmetric as required. Moreover, since the modulus of all eigenvalues of T is always 1, the eigenvalues of L_T are the same of $L_{(3)}$. Since the approach to employ relies on a controller that is robust against the eigenvalues of the communication topology, using $L_T = \hat{T}^{-1} L_{(3)} \hat{T}$ as a transformed version of the original communication topology is still valid. Hence, no changes in the generalized plant matrices are required.

The communication suggested for this problem can be interpreted in two ways: 1) the usual physical communication where agents communicate their global position (e.g. by using a GPS) to other agents. 2) use local position sensors where instead of measuring their own position, agents measure other agents relative position (e.g. by using a camera/laser system). For the latter, communication is directly implied by measuring relative positions.

c) The LMI 6.70a is structured as

$$\left[\begin{array}{ccc|c} & & & 0 \\ W_1^{\mathrm{T}} & & W_2^{\mathrm{T}} & 0 \\ & & & I \end{array}\right] \left[\begin{array}{ccc|c} W_0 & & 0 & 0 \\ & & & \\ 0 & & -\gamma^2 I & 0 \\ & & & \\ 0 & & 0 & I \end{array}\right] \left[\begin{array}{c} W_1 \\ \\ W_2 \\ \\ 0 \quad 0 \quad I \end{array}\right] > 0\,, \tag{A.45}$$

after performing the multiplications it yields

$$(W_1^{\mathrm{T}} W_0 W_1 + \left[\begin{smallmatrix} 0 & 0 \\ 0 & I \end{smallmatrix}\right]) - W_2^{\mathrm{T}} (\gamma^2 I) W_2 > 0. \tag{A.46}$$

Applying the schur complement to above inequality leads to

$$\begin{bmatrix} W_1^{\mathrm{T}} W_0 W_1 + \left[\begin{smallmatrix} 0 & 0 \\ 0 & I \end{smallmatrix}\right] & W_2^{\mathrm{T}} \\ W_2 & \gamma^{-2} I \end{bmatrix} > 0, \tag{A.47}$$

which is now linear in all the decision variables. Notice that one has to maximize the variable $\tilde{\gamma} = \gamma^{-2}$.

To completely remove the LMI 6.70b, additionally to $\tilde{R} > 0$, one can impose the following constraints:

$$\tilde{Q} = -\tilde{R} < 0 \qquad \tilde{R}\Theta = \Theta R \qquad \tilde{S}^{\mathrm{T}} = -\tilde{S} \qquad \tilde{S}\Theta = \Theta S$$

Those constraints simplify the LMI 6.70b as follows (recall that $\Theta^{\mathrm{T}} = \Theta$)

$$\tilde{Q} - \Theta\tilde{S}^{\mathrm{T}} - \tilde{S}\Theta + \Theta\tilde{R}\Theta < 0$$

$$-\tilde{R} + \tilde{S}\Theta - \tilde{S}\Theta + \tilde{R}\Theta\Theta < 0$$

$$-\tilde{R}(I - \Theta^2) < 0$$

Since $|\Theta| \leq 1$, the factor $(I - \Theta^2)$ is always positive definite. Under the mentioned constraints, this LMI is satisfied at all times and, hence there is no need to include in the synthesis. This is certainly very conservative since many structural constraints are imposed on the multipliers, which possibly can lead to infeasibility. However, for large number of parameters this can greatly reduce complexity.

d & e) A set of weighting filters to impose low steady state error and reasonable control effort is given by

$$W_S = \mathrm{diag}\left(\frac{(1/M_{S_i})(s + w_{S_i}M_{S_i})}{s + w_{S_i}A_{S_i}}\right), \quad i = 1, 2, 3 \tag{A.48}$$

$$W_{KS} = \mathrm{diag}\left(\frac{(c_{KS_i}/M_{KS_i})(s + w_{KS_i})}{s + c_{KS_i}w_{S_i}}\right), \quad i = 1, 2 \tag{A.49}$$

$$\begin{array}{cccc} w_{S_{1,2}} = 0.001 & w_{S_3} = 0.0001 & w_{KS_1} = 0.1 & w_{KS_2} = 0.1 \\ A_{S_{1,2}} = 0.01 & A_{S_3} = 0.01 & M_{KS_1} = 0.5 & M_{KS_2} = 0.9 \\ M_{S_{1,2}} = 1 & M_{S_3} = 1 & c_{KS_1} = 1000 & c_{KS_2} = 1000 \end{array} \tag{A.50}$$

After the state feedback gains are synthesized, the simulation can be run. A test with a square formation can be seen in Figure A.29), where the trajectories, initial conditions and the final interaction topology are shown. The formation and orientation error can be seen in Figure A.30.

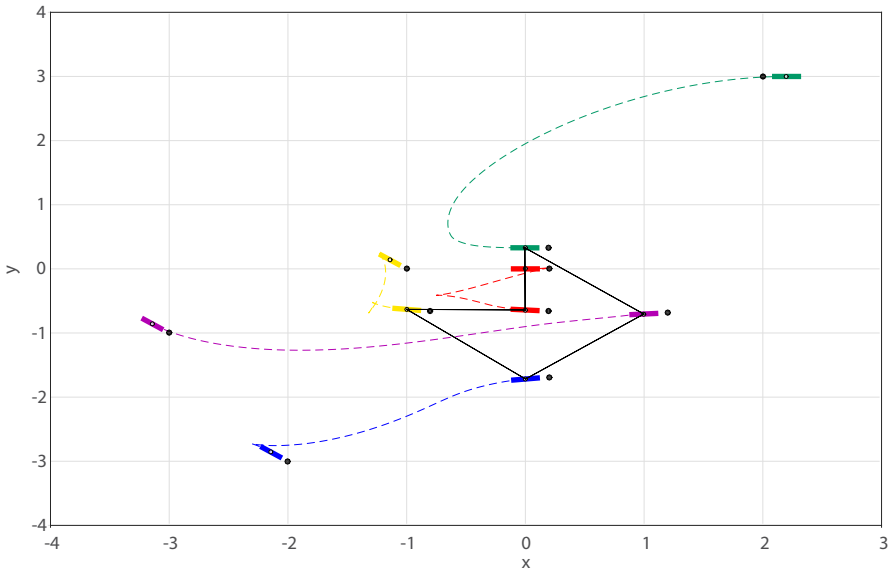


Figure A.29: Formation reference simulation.

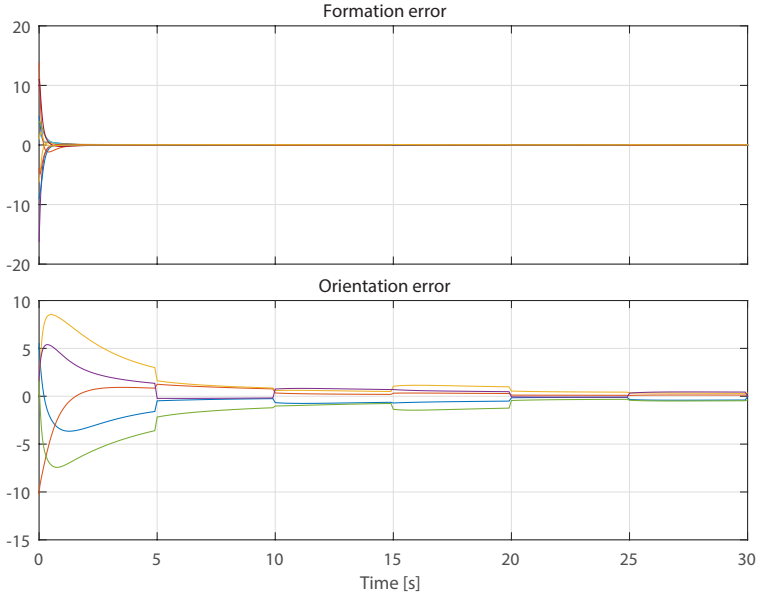


Figure A.30: Formation error.

Solutions — Distributed Control of Spatially Interconnected Systems

Problem 7.1 *(State Space Realization of a PDE)*

a) The central finite difference of the n-th order derivative of function $f(x)$ has the analogous form

$$\frac{d^n f}{dx^n}(x) = \frac{\delta_h^n[f](x)}{h^n} = \frac{1}{h^n}\sum_{i=0}^{n}(-1)^i \begin{pmatrix} n \\ i \end{pmatrix} f(x + (\frac{n}{2} - i)h), \tag{A.51}$$

where h is the distance between two grid points, in our case the distance between two subsystems.

Discretizing (7.30) in space requires the discretization of $\frac{\partial^4 y}{\partial s^4}(t,s)$. The application of (A.51) yields

$$\begin{aligned}\frac{\partial^4 y}{\partial s^4}(t,s) = \frac{1}{h^4}[&y(t,s+2h) - 4y(t,s+h) + 6y(t,s) \\ &- 4y(t,s-h) + y(t,s-2h)]\end{aligned} \tag{A.52}$$

Replace the continuous spatial variable s in (A.52) with its discrete counterpart s, e.g. $s + 2h \to s + 2$

$$\frac{\partial^4 y}{\partial x^4}(t,s) = \frac{1}{H^4}\left[y(t,s+2) - 4y(t,s+1) + 6y(t,s) - 4y(t,s-1) + y(t,s-2)\right]. \tag{A.53}$$

Inserting (A.53) in (7.30) and then rearranging the terms leads to (A.54) discrete in space and continuous in time

$$\begin{aligned}\frac{\partial^2 y}{\partial t^2}(t,s) = &-\alpha[y(t,s+2) - 4y(t,s+1) + 6y(t,s) - 4y(t,s-1) + y(t,s-2)] \\ &+ \beta u(t,s),\end{aligned} \tag{A.54}$$

where $\alpha = \frac{EI}{\rho A h^4}$, and $\beta = \frac{1}{\rho A}$. It describes how the dynamics of subsystem s is influenced by itself and its nearest neighbours, i.e. $s-2$, $s-1$, $s+1$ and $s+2$.

According to the definition of the spatial shift operators $\mathbf{S}$

$$\mathbf{S}y(t,s) := y(t,s+1), \tag{A.55}$$

the equation of motion (A.54) can be rewritten as

$$\frac{\partial^2 y}{\partial t^2}(t,s) = -\alpha[\mathbf{S}^2 - 4\mathbf{S} + 6 - 4\mathbf{S}^{-1} + \mathbf{S}^{-2}]y(t,s) + \beta u(t,s). \tag{A.56}$$

b) The state space model of a multi-dimensional system has the form

$$\begin{bmatrix} \dot{x}^t(t,s) \\ w_+(t,s) \\ w_-(t,s) \\ y(t,s) \end{bmatrix} = \left[\begin{array}{ccc|c} A^{tt} & A_+^{ts} & A_-^{ts} & B^t \\ A_+^{st} & A_{++}^{ss} & A_{+-}^{ss} & B_+^s \\ A_-^{st} & A_{-+}^{ss} & A_{--}^{ss} & B_-^s \\ \hline C^t & C_+^s & C_-^s & D \end{array}\right] \begin{bmatrix} x^t(t,s) \\ v_+(t,s) \\ v_-(t,s) \\ u(t,s) \end{bmatrix}. \tag{A.57}$$

Choose the temporal and spatial states as

$$x^t = \begin{bmatrix} y(t,s) \\ \dot{y}(t,s) \end{bmatrix}, \quad \begin{bmatrix} v_+ \\ v_- \end{bmatrix} = \begin{bmatrix} y(t,s-2) \\ y(t,s-1) \\ y(t,s+2) \\ y(t,s+1) \end{bmatrix}.$$

It is not difficult to fill in the matrices A^{tt}, A_+^{ts} A_-^{ts}, and B^t just by recalling the equation of motion (A.54)

$$A^{tt} = \begin{bmatrix} 0 & 1 \\ -6\alpha & 0 \end{bmatrix}, \quad A_+^{ts} = \begin{bmatrix} 0 & 0 \\ -\alpha & 4\alpha \end{bmatrix}, \quad A_-^{ts} = \begin{bmatrix} 0 & 0 \\ -\alpha & 4\alpha \end{bmatrix}, \quad B^t = \begin{bmatrix} 0 \\ \beta \end{bmatrix}$$

To fill in matrices $\left[\begin{array}{ccc|c} A_+^{st} & A_{++}^{ss} & A_{+-}^{ss} & B_+^s \\ A_-^{st} & A_{-+}^{ss} & A_{--}^{ss} & B_-^s \end{array}\right]$, only the equivalences between the shifted spatial states and spatial/temporal states are involved, i.e.

$$\begin{bmatrix} w_+(t,s) \\ w_-(t,s) \end{bmatrix} = \begin{bmatrix} y(t,s-1) \\ y(t,s) \\ y(t,s+1) \\ y(t,s) \end{bmatrix}.$$

Finally, we obtain the state space model defining the dynamics on subsystems

$$\begin{bmatrix} \dot{y}(t,s) \\ \ddot{y}(t,s) \\ y(t,s-1) \\ y(t,s) \\ y(t,s+1) \\ y(t,s) \end{bmatrix} = \left[\begin{array}{cc|cc|cc} 0 & 1 & 0 & 0 & 0 & 0 \\ -6\alpha & 0 & -\alpha & 4\alpha & -\alpha & 4\alpha \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} y(t,s) \\ \dot{y}(t,s) \\ y(t,s-2) \\ y(t,s-1) \\ y(t,s+2) \\ y(t,s+1) \end{bmatrix} + \begin{bmatrix} 0 \\ \beta \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u(t,s) \tag{A.58}$$

$$y(t,s) = \left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} y(t,s) \\ \dot{y}(t,s) \\ y(t,s-2) \\ y(t,s-1) \\ y(t,s+2) \\ y(t,s+1) \end{bmatrix} \tag{A.59}$$

c) The approximation of $\frac{\partial^2 y}{\partial t^2}(t,s)$ takes the form

$$\frac{\partial^2 y}{\partial t^2}(t,s) = \frac{1}{T^2}\left[y(t+T,s) - 2y(t,s) + y(t-T,s)\right]. \tag{A.60}$$

Analogously replace the continuous temporal variable t in (A.60) with its discrete counterpart k

$$\frac{\partial^2 y}{\partial t^2}(k,s) = \frac{1}{T^2}\left[y(k+1,s) - 2y(k,s) + y(k-1,s)\right]. \tag{A.61}$$

Replacing both (A.53) and (A.61) in (7.30) and rearranging the terms leads to a 2D difference equation discrete both in time and space

$$\begin{aligned} y(k,s) =& 2y(k-1,s) - y(k-2,s) - \alpha[y(k-1,s-2) - 4y(k-1,s-1) + 6y(k-1,s) \\ & - 4y(k-1,s+1) + y(k-1,s+2)] + \beta u(k-1,s), \end{aligned} \tag{A.62}$$

where $\alpha = \frac{T^2 EI}{\rho A h^4}$, and $\beta = \frac{T^2}{\rho A}$.

d) Following the same procedure of converting a discrete transfer function into its controller canonical state space form, the matrices $\left[A^{tt}\ A_{+}^{ts}\ A_{-}^{ts}\ B^t\right]$ and C_t can be easily filled up. The rest of the matrices rely again on the equivalence of shifted spatial states and spatial/temporal states. The state space model discrete both in time and space is obtained as

$$\left[\begin{array}{c} x(k-1,s) \\ x(k,s) \\ \hdashline x(k-1,s-1) \\ x(k-1,s) \\ \hdashline x(k-1,s+1) \\ x(k-1,s) \end{array}\right] = \left[\begin{array}{cc:cc:cc} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 2-6\alpha & -\alpha & 4\alpha & -\alpha & 4\alpha \\ \hdashline 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} x(k-2,s) \\ x(k-1,s) \\ \hdashline x(k-1,s-2) \\ x(k-1,s-1) \\ \hdashline x(k-1,s+2) \\ x(k-1,s+1) \end{array}\right] + \left[\begin{array}{c} 0 \\ 1 \\ \hdashline 0 \\ 0 \\ \hdashline 0 \\ 0 \end{array}\right] u(t,s) \tag{A.63}$$

$$y(k,s) = \left[0\ \beta\ 0\ 0\ 0\ 0\right] \left[\begin{array}{c} x(k-2,s) \\ x(k-1,s) \\ \hdashline x(k-1,s-2) \\ x(k-1,s-1) \\ \hdashline x(k-1,s+2) \\ x(k-1,s+1) \end{array}\right] \tag{A.64}$$

Problem 7.2 *(Simulate the Open Loop Response)*

a) Fig. A.31 shows the experimental measurements after an impulse is applied at the middle of the beam. The measured outputs are the sensor voltage.

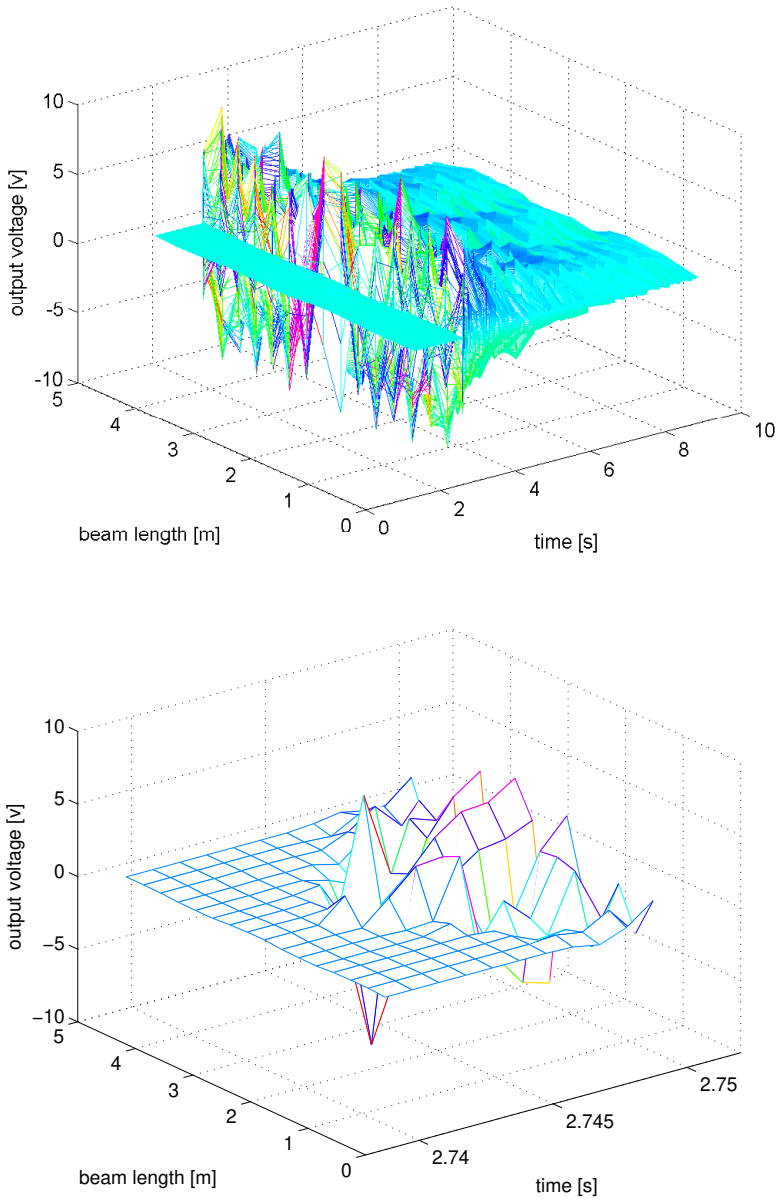


Figure A.31: The experimental impulse response (u.) and a zoomed-in where the vibration starts (b.)

a/b) For an extensive solution to the problem, please consult the MATLAB files.

Problem 7.3 *(Controller Design and Closed-loop Simulation)*

a) The state space representation of the plant discrete both in time and space, taking the disturbance into consideration, has a form

$$\begin{bmatrix} x(k+1,s) \\ w(k,s) \end{bmatrix} = \begin{bmatrix} A_{TT} & A_{TS} \\ A_{ST} & A_{SS} \end{bmatrix} \begin{bmatrix} x(k,s) \\ v(k,s) \end{bmatrix} + \begin{bmatrix} B_T & B_T \\ B_S & B_S \end{bmatrix} \begin{bmatrix} d(k,s) \\ u(k,s) \end{bmatrix} \tag{A.65}$$

$$y = \begin{bmatrix} C_T & C_S \end{bmatrix} \begin{bmatrix} x(k,s) \\ v(k,s) \end{bmatrix} \tag{A.66}$$

The shaping filter for control sensitivity W_K has a state space representation

$$x_1(k+1,s) = A_1 x_1(k,s) + B_1 u(k,s) \tag{A.67}$$

$$z_1(k,s) = C_1 x_1(k,s) + D_1 u(k,s), \tag{A.68}$$

whereas the shaping filter for sensitivity W_S

$$x_2(k+1,s) = A_2 x_2(k,s) + B_2 y(k,s) \tag{A.69}$$

$$z_2(k,s) = C_2 x_2(k,s) + D_2 y(k,s). \tag{A.70}$$

The series interconnection of the control sensitivity W_K and the sensitivity W_S with the plant (A.65)-(A.66) only augments the temporal dynamics of the generalized plant, which takes the form

$$\begin{bmatrix} x(k+1,s) \\ x_1(k+1,s) \\ x_2(k+1,s) \\ w(k,s) \end{bmatrix} = \begin{bmatrix} A_{TT} & 0 & 0 & A_{TS} \\ 0 & A_1 & 0 & 0 \\ B_2C_T & 0 & A_2 & B_2C_S \\ A_{ST} & 0 & 0 & A_{SS} \end{bmatrix} \begin{bmatrix} x(k,s) \\ x_1(k,s) \\ x_2(k,s) \\ v(k,s) \end{bmatrix} + \begin{bmatrix} B_T & B_T \\ 0 & B_1 \\ 0 & 0 \\ B_S & B_S \end{bmatrix} \begin{bmatrix} d(k,s) \\ u(k,s) \end{bmatrix} \tag{A.71}$$

$$\begin{bmatrix} z_1(k,s) \\ z_2(k,s) \\ y(k,s) \end{bmatrix} = \begin{bmatrix} 0 & C_1 & 0 & 0 \\ D_2C_T & 0 & C_2 & D_2C_S \\ C_T & 0 & 0 & C_S \end{bmatrix} \begin{bmatrix} x(k,s) \\ x_1(k,s) \\ x_2(k,s) \\ v(k,s) \end{bmatrix} + \begin{bmatrix} 0 & D_1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d(k,s) \\ u(k,s) \end{bmatrix} \tag{A.72}$$

b) Given the structure of the shaping filters W_S and W_K as in (7.31) and (7.32), by adjusting the tuning knobs, it is possible to realize the required closed-loop bandwidth (e.g. 10 Hz), efficient suppression of the vibration at modes, with reasonable control effort being used.

To construct the generalized plant, the transfer function form of the shaping filters needs to be converted into their state space representations (A.67)-(A.68), (A.69)-(A.70) using standard *ss* command of MATLAB.

Remarks:

- The first argument of the function 'hfMD.m' requires a generalized plant.
- Note that, the function 'hfMD.m' admits by default that the last input of the generalized plant is the output of the controller, and that the last output of the generalized plant is the output of the plant. So when we construct the generalized plant, the last input/output channel has to be $u(k,s) \to y(k,s)$ as in (A.72).
- 'hfMD.m' allows a controller design for either a continuous time or a discrete time plant. The synthesis conditions written in 'hfMD.m' are for continuous time systems. If your plant is discrete in time, let the first element of the vector 'sys.blk' be negative when you define the system matrices. For example, for a system with 2 temporal states, 1 dimension of $v_+(k,s)$ and 1 dimension of $v_-(k,s)$, define sys.blk = [-2,1,1]. Then 'hfMD.m' recognizes a discrete time system, and applies the bilinear transformation to convert the discrete time system to continuous time system, before the controller synthesis process starts.
- To simulate the closed-loop response, you can use the function 'starMD.m' to interconnect the subsystem plant G and the subsystem controller K. Note that, the last input/output channel of the plant G should be defined as the interconnection signals with the controller K, i.e. $u(k,s) \to y(k,s)$ as defined in (A.74). ((A.73)-(A.74) are defined the same way as (A.63) by just adding the disturbance $d(k,s)$ as an extra input)

$$\begin{bmatrix} x(k-1,s) \\ x(k,s) \\ \hdashline x(k-1,s-1) \\ x(k-1,s) \\ \hdashline x(k-1,s+1) \\ x(k-1,s) \end{bmatrix} = \left[\begin{array}{cc:cc:cc} 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 2-6\alpha & -\alpha & 4\alpha & -\alpha & 4\alpha \\ \hdashline 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} x(k-2,s) \\ x(k-1,s) \\ \hdashline x(k-1,s-2) \\ x(k-1,s-1) \\ \hdashline x(k-1,s+2) \\ x(k-1,s+1) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ \hdashline 0 & 0 \\ 0 & 0 \\ \hdashline 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d(k,s) \\ u(k,s) \end{bmatrix} \tag{A.73}$$

$$\begin{bmatrix} u(k,s) \\ y(k,s) \\ y(k,s) \end{bmatrix} = \left[\begin{array}{c:c:c} 0\ 0 & 0\ 0 & 0\ 0 \\ 0\ \beta & 0\ 0 & 0\ 0 \\ \hdashline 0\ \beta & 0\ 0 & 0\ 0 \end{array}\right] \begin{bmatrix} x(k-2,s) \\ x(k-1,s) \\ \hdashline x(k-1,s-2) \\ x(k-1,s-1) \\ \hdashline x(k-1,s+2) \\ x(k-1,s+1) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d(k,s) \\ u(k,s) \end{bmatrix} \tag{A.74}$$

– When using 'csimulateMD.m' or 'dsimulateMD.m' or 'dsimulateMD_ATC.m' to simulate the closed-loop responses, in order to observe the involved control effort and measured output after simulation, it is necessary to output these signals separately, as in (A.74), where the first output is to output the control effort, the second output to output the measured output of the plant, whereas the third output is directed to the controller by using 'starMD.m'.

Fig. A.32 and Fig. A.33 shows the closed-loop response of the beam, where an impulse and a chirp disturbance is injected at subsystem 8, respectively. In Problem 7.2 we have seen that, since the PDE does not include any damping terms, the open-loop beam vibrates forever after the disturbance is introduced. Nevertheless, under the control of designed distributed controllers, the vibration is fast suppressed.

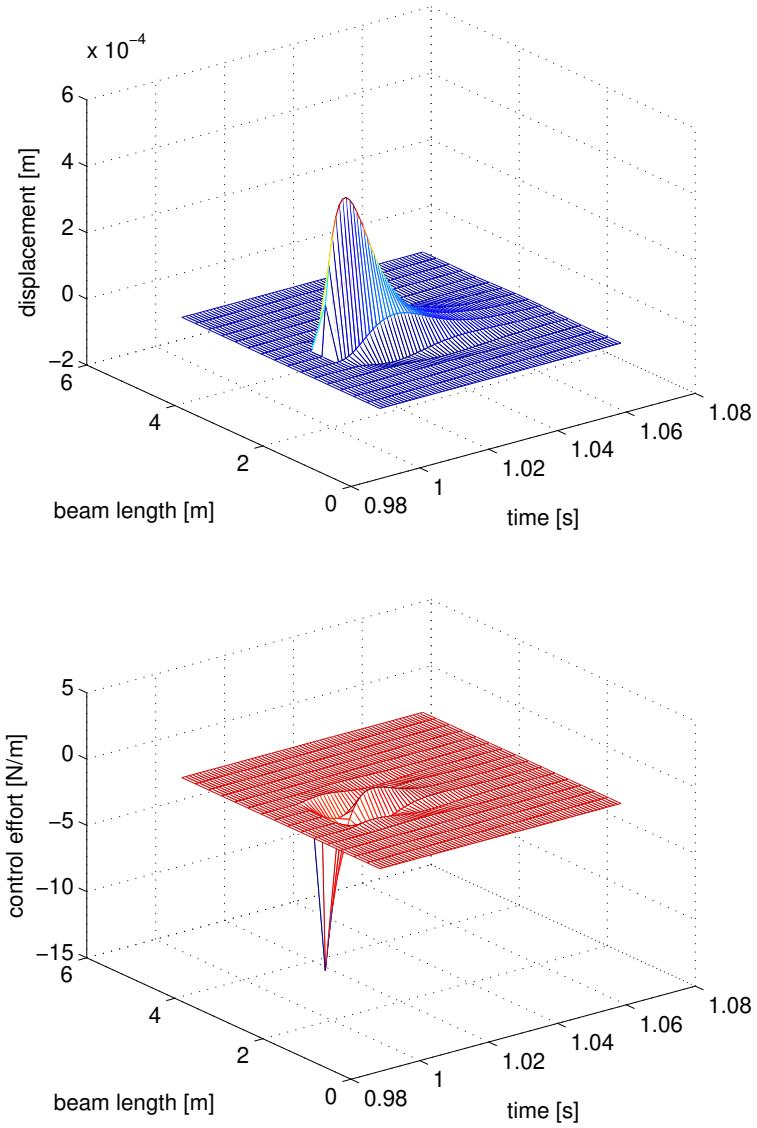


Figure A.32: Closed-loop response of the PDE to an impulse disturbance (u.) and the control effort (b.)

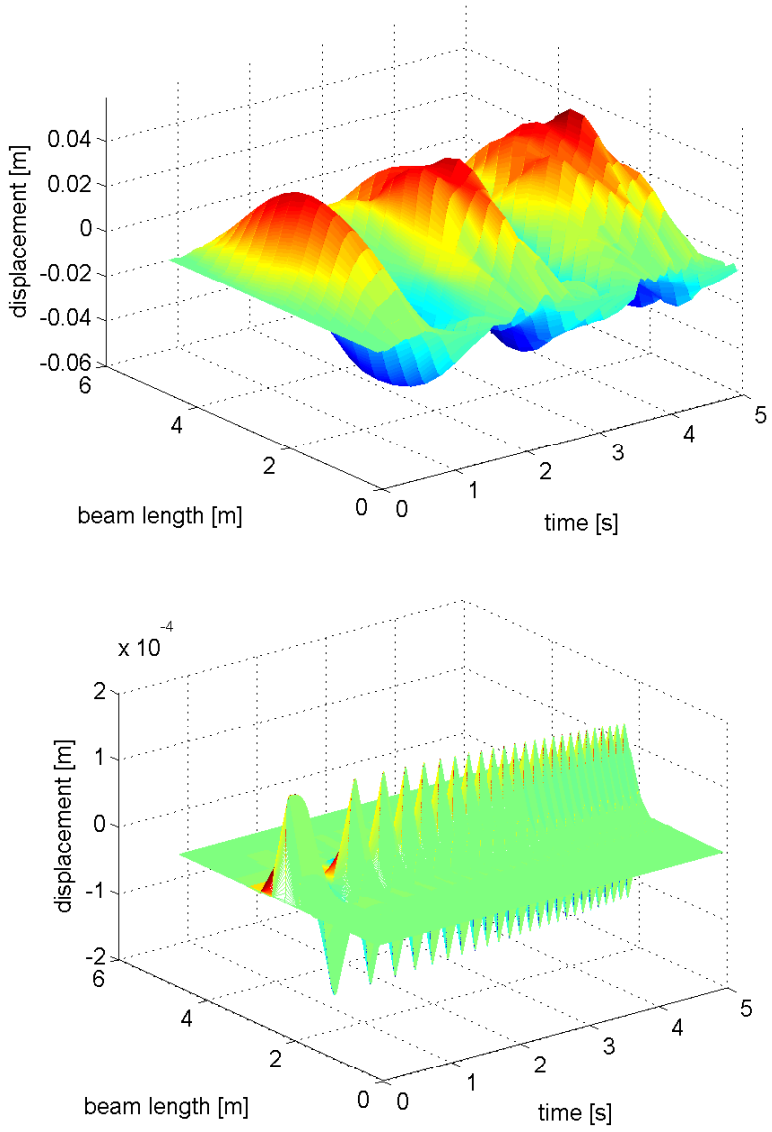
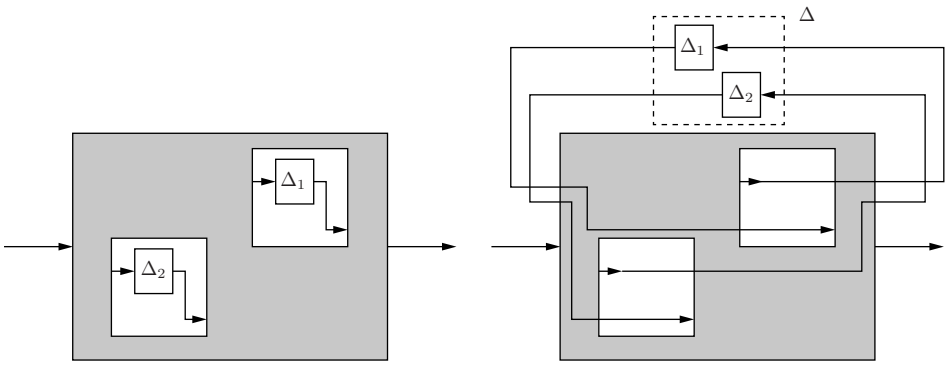


Figure A.33: Open-loop response of the PDE to an chirp signal up to 10 Hz as disturbance (u.) and the closed-loop response (b.)

Appendix B

Linear Fractional Transformations

B.1 Definitions



(a) Absorbed "Deltas" (b) Pulled out "Deltas"

Figure B.1: Visualization of the "Pulling-Out-The-Deltas"-procedure, [13]

Definition B.1 (Upper LFT, [13]) *Let the complex matrix M be partitioned as*

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \in \mathbb{C}^{(p_1+p_2)\times(q_1+q_2)}.$$

Then an upper LFT with respect to $\Delta \in \mathbb{C}^{q_1 \times p_1}$ is defined as the map

$$\Delta \star M : \mathbb{C}^{q_2 \times p_2} \mapsto \mathbb{C}^{p_1 \times q_1}$$

with

$$\Delta \star \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] = M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12},$$

provided that the inverse of $(I - M_{11}\Delta)$ exists.

Definition B.2 (Lower LFT, [13]) *Let the complex matrix M be partitioned as in Def. B.1. Then a lower LFT with respect to $\Delta \in \mathbb{C}^{q_2 \times p_2}$ is defined as the map*

$$\Delta \star M : \mathbb{C}^{q_1 \times p_1} \mapsto \mathbb{C}^{p_2 \times q_2}$$

with

$$\left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] \star \Delta = M_{11} + M_{12}\Delta(I - M_{22}\Delta)^{-1}M_{21},$$

provided that the inverse of $(I - M_{22}\Delta)$ exists.

B.2 Useful Formulas

A multiplication of two LFTs is another LFT

$$\left(\Delta_M \star \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right]\right)\left(\Delta_N \star \left[\begin{array}{c|c} N_{11} & N_{12} \\ \hline N_{21} & N_{22} \end{array}\right]\right) = \left[\begin{array}{cc} \Delta_M & 0 \\ 0 & \Delta_N \end{array}\right] \star \left[\begin{array}{cc|c} M_{11} & M_{12}N_{21} & M_{12}N_{22} \\ 0 & N_{11} & N_{12} \\ \hline M_{21} & M_{22}N_{21} & M_{22}N_{22} \end{array}\right].$$

A useful formula exists in

$$(A + B\Delta)(C + D\Delta)^{-1} = \Delta \star \left[\begin{array}{c|c} -C^{-1}D & C^{-1} \\ \hline B - AC^{-1}D & AC^{-1} \end{array}\right],$$

while it is assumed that the inverse of C exists.

Appendix C

Mathematical Tools

Often, several tools are necessary to derive LMI formulations. Here is a collection of some useful theorems, lemmas, etc.

Theorem C.1 (Full-Block $\mathcal{S}$-Procedure, [16]) *Given*

$$G(\delta) = \Delta \star \left[\begin{array}{c|c} G_{11} & G_{12} \\ \hline G_{21} & G_{22} \end{array}\right] = G_{22} + G_{21}\Delta(I - G_{11}\Delta)^{-1}G_{12},\ \delta \in P_\delta,$$

the matrix inequality

$$G^\top(\delta)QG(\delta) < 0,\ \forall \delta \in P_\delta$$

holds iff $\exists M = M^\top$, *such that*

$$\left[\begin{array}{c} * \\ \hline * \end{array}\right]^\top \left[\begin{array}{c|c} M & 0 \\ \hline 0 & Q \end{array}\right] \left[\begin{array}{cc} G_{11} & G_{12} \\ I & 0 \\ \hline G_{21} & G_{22} \end{array}\right] < 0,\ \left[\begin{array}{c} * \\ * \end{array}\right]^\top M \begin{bmatrix} I \\ \Delta \end{bmatrix} > 0,\ \forall \delta \in P_\delta.$$

Lemma C.1 (Dualization Lemma, [16]) *Assume Q is a nonsingular matrix and $\mathcal{S}$ a subspace with* $\text{in}_0(Q|_{\mathcal{S}})$. *Then* $\text{in}(Q|_{\mathcal{S}}) + \text{in}(Q^{-1}|_{\mathcal{S}}) = \text{in}(Q)$.

Lemma C.2 (Dual Quadratic Inequalities, [16]) *Consider the matrix inequality in X*

$$\begin{bmatrix} I_m \\ A + W^\top XV \end{bmatrix}^\top Q \begin{bmatrix} I_m \\ A + W^\top XV \end{bmatrix} < 0 \tag{C.1}$$

with $A \in \mathbb{R}^{n\times m}$, $\text{in}(Q) = (m, 0, n)$. *Therefore $P = Q^{-1}$ exists. Then, due to Lemma C.1, (C.1) is equivalent to*

$$\begin{bmatrix} -(A + W^\top XV)^\top \\ I_n \end{bmatrix}^\top P \begin{bmatrix} -(A + W^\top XV)^\top \\ I_n \end{bmatrix} > 0 \tag{C.2}$$

Lemma C.3 (Parameter Elimination Lemma, [16]) *Inequality (C.2) is solvable iff*

$$\mathcal{N}_V^\top \begin{bmatrix} I_m \\ A \end{bmatrix}^\top Q \begin{bmatrix} I_m \\ A \end{bmatrix} \mathcal{N}_V < 0, \quad \textit{and} \quad \mathcal{N}_W^\top \begin{bmatrix} -A^\top \\ I_n \end{bmatrix}^\top P \begin{bmatrix} -A^\top \\ I_n \end{bmatrix} \mathcal{N}_W > 0, \tag{C.3}$$

with $Q = P^{-1}$.

Appendix D

Induced $\mathcal{L}_2$-Gain Based State Feedback Synthesis

The following reviews the synthesis of a gain-scheduled state feedback gain using parameter-dependent Lyapunov functions. The derivation is based on [5] and is extended, such that the existence conditions are convexified for the cases when simplifying assumptions on the model are made. More specifically the cases of affine/polytopic and rational/LFT models are considered using tools such as multi-convexity constraints and the Full-Block $\mathcal{S}$-Procedure to derive convex LMI constraints.

Goal

- Synthesize a stabilizing state feedback with three distinct methods to convexify the problem:
  - Gridding,
  - Polytopic approximation and multi-convexity conditions,
  - Using the Full-Block $\mathcal{S}$-Procedure and multi-convexity conditions on the quadratic multiplier conditions.

References *The core methodology worked out in this exercise is contained in [5] and presented here with a slightly more streamlined proof. Results from [16], as well as [11], are adopted to provide a relatively broad overview about the now standard techniques for convexifying matrix inequality based problems.*

Problem Setup Consider the LPV plant

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \left[\begin{array}{c|cc} A(\rho) & B_p(\rho) & B_u(\rho) \\ C_p(\rho) & 0 & D_{pu}(\rho) \\ I & 0 & 0 \end{array}\right] \begin{bmatrix} x \\ w \\ u \end{bmatrix} \tag{D.1}$$

It is the goal to derive conditions for the synthesis of a scheduled state feedback gain $D_K(\rho)$, such that $A_{\mathrm{cl}}(\rho) = A(\rho) + B_u(\rho)D_K(\rho)$ is stable for all $\rho \in [-1, 1]$ and achieves an induced $\mathcal{L}_2$-gain performance of γ.

Dualization Starting from the parameter-dependent Bounded Real Lemma

$$\begin{aligned} P(\rho)A_{\mathrm{cl}}(\rho) + A_{\mathrm{cl}}(\rho)^\top P(\rho) + \dot{P}(\rho) + \ldots \\ \ldots \frac{1}{\gamma}(C_{p,\mathrm{cl}}^\top(\rho)C_{p,\mathrm{cl}}(\rho) + D_{pp,\mathrm{cl}}^\top(\rho)D_{pp,\mathrm{cl}}(\rho)) - \gamma I < 0, \quad \forall \rho \in P_\rho \end{aligned} \tag{D.2}$$

$$P(\rho) > 0, \quad \forall \rho \in P_\rho \tag{D.3}$$

the dual matrix inequality in the unknown $Y(\rho) = P^{-1}(\rho)$ is first derived.

Therefore, rewrite (2.39) as

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \left[\begin{array}{cc|cc} \dot{P}(\rho) & P(\rho) & & \\ P(\rho) & 0 & & \\ \hline & & -\gamma I & 0 \\ & & 0 & \frac{1}{\gamma} I \end{array}\right] \left[\begin{array}{cc} I & 0 \\ A_{\mathrm{cl}}(\rho) & B_{p,\mathrm{cl}}(\rho) \\ \hline 0 & I \\ C_{p,\mathrm{cl}}(\rho) & D_{pp,\mathrm{cl}}(\rho) \end{array}\right] < 0, \quad \forall (\rho, \eta) \in P_\rho \times P_\eta \tag{D.4}$$

where $\eta = \dot{\rho}$ is the rate of change of the scheduling parameter and appears in $\dot{P}(\rho)$. Dualize using Lemma C.2 and receive

$$\begin{bmatrix} * \\ * \end{bmatrix}^\top \left[\begin{array}{cc|cc} 0 & Y(\rho) & & \\ Y(\rho) & -\dot{Y}(\rho) & & \\ \hline & & -\frac{1}{\gamma} I & 0 \\ & & 0 & \gamma I \end{array}\right] \left[\begin{array}{cc} -A_{\mathrm{cl}}^\top(\rho) & -C_{p,\mathrm{cl}}^\top(\rho) \\ I & 0 \\ \hline -B_{p,\mathrm{cl}}^\top(\rho) & -D_{pp,\mathrm{cl}}^\top(\rho) \\ 0 & I \end{array}\right] > 0, \quad \forall (\rho, \eta) \in P_\rho \times P_\eta \tag{D.5}$$

It is easy to check that

$$\begin{bmatrix} \dot{P}(\rho) & P(\rho) \\ P(\rho) & 0 \end{bmatrix} \begin{bmatrix} 0 & Y(\rho) \\ Y(\rho) & -\dot{Y}(\rho) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

due to $\dot{Y}(\rho) = -P^{-1}(\rho)\dot{P}(\rho)P^{-1}(\rho)$.

Parameter-Dependent Existence Conditions In order to derive parameter-dependent LMI conditions for the existence of a state feedback gain that achieves an induced $\mathcal{L}_2$-gain performance γ as an upper bound, the elimination lemma can be used.

With

$$\begin{bmatrix} A_{\mathrm{cl}}(\rho) & B_{p,\mathrm{cl}}(\rho) \\ C_{p,\mathrm{cl}}(\rho) & D_{pp,\mathrm{cl}}(\rho) \end{bmatrix} = \begin{bmatrix} A(\rho) & B_p(\rho) \\ C_p(\rho) & 0 \end{bmatrix} + \begin{bmatrix} B_u(\rho) \\ D_{pu}(\rho) \end{bmatrix} D_K(\rho) \begin{bmatrix} I & 0 \end{bmatrix} \tag{D.6}$$

the controller gain can be eliminated by pre- and postmultiplication of (D.5), $\mathcal{N}(\rho) = \ker(\begin{bmatrix} B_u^\top(\rho) & D_{pu}^\top(\rho) \end{bmatrix})$. For simplicity, assume that both matrices $B_u(\rho)$ and $D_{pu}(\rho)$ are not parameter dependent, i.e. $\mathcal{N} = \ker(\begin{bmatrix} B_u^\top & D_{pu}^\top \end{bmatrix})$. which yields the existence condition

$$\mathcal{N}^\top \begin{bmatrix} * \\ * \end{bmatrix}^\top \left[\begin{array}{cc:cc} 0 & Y(\rho) & & \\ Y(\rho) & -\dot{Y}(\rho) & & \\ \hdashline & & -\frac{1}{\gamma}I & 0 \\ & & 0 & \gamma I \end{array}\right] \left[\begin{array}{cc} -A^\top(\rho) & -C_p^\top(\rho) \\ I & 0 \\ \hdashline -B_p^\top(\rho) & 0 \\ 0 & I \end{array}\right] \mathcal{N} > 0, \qquad \forall (\rho, \eta) \in P_\rho \times P_\eta \tag{D.7}$$

Controller Construction Assume that the plant is properly defined and scaled such that we can write

$$D_{pu} = \begin{bmatrix} 0 \\ I \end{bmatrix}, \quad C_p = \begin{bmatrix} C_{p_1} \\ C_{p_2} \end{bmatrix}.$$

Under the assumption that a solution $Y(\rho) = P^{-1}(\rho)$ for the dual problem derived above has already been found we use the the closed-loop plant factorization

$$\begin{bmatrix} A_{\mathrm{cl}}(\rho) & B_{p,\mathrm{cl}}(\rho) \\ C_{p,\mathrm{cl}}(\rho) & D_{pp,\mathrm{cl}}(\rho) \end{bmatrix} = \begin{bmatrix} A(\rho) & B_{\mathrm{p}}(\rho) \\ C_{\mathrm{p}}(\rho) & 0 \end{bmatrix} + \begin{bmatrix} B_u(\rho) \\ D_{pu}(\rho) \end{bmatrix} D_K(\rho) \begin{bmatrix} I & 0 \end{bmatrix} \tag{D.8}$$

to show that the state feedback gain

$$D_K(\rho) = -(\gamma B_u^\top P(\rho) + C_{p_2}), \tag{D.9}$$

is a solution to the state feedback problem by comparing above results.

First, we are making use of the assumptions

$$D_{pu} = \begin{bmatrix} 0 \\ I \end{bmatrix}, \quad C_p = \begin{bmatrix} C_{p_1} \\ C_{p_2} \end{bmatrix}$$

to first find an explicit expression for $\mathcal{N}$. We will then prove the formula for the gain (D.9) by comparing (D.5) and (D.7).

With

$$\mathcal{N} = \ker(\begin{bmatrix} B_u^\top & D_{pu}^\top \end{bmatrix}) = \begin{bmatrix} I & 0 \\ 0 & I \\ -B_u^\top & 0 \end{bmatrix},$$

observe that (D.7) becomes

$$\begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:ccc} 0 & Y(\rho) & & & \\ Y(\rho) & -\dot{Y}(\rho) & & & \\ \hdashline & & -\frac{1}{\gamma}I & 0 & \\ & & 0 & \gamma I & \end{array}\right] \left[\begin{array}{cc} -(A(\rho)-B_uC_{p_2})^\top & -C_{p_1}^\top \\ I & 0 \\ \hdashline -B_p^\top(\rho) & 0 \\ -B_u^\top & 0 \end{array}\right] > 0, \quad \forall(\rho,\eta) \in P_\rho \times P_\eta \ , \tag{D.10}$$

while (D.5) can be written as

$$\begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:ccc} 0 & Y(\rho) & & & \\ Y(\rho) & -\dot{Y}(\rho) & & & \\ \hdashline & & -\frac{1}{\gamma}I & 0 & 0 \\ & & 0 & \gamma I & 0 \\ & & 0 & 0 & \gamma I \end{array}\right] \left[\begin{array}{ccc} -(A(\rho)+B_uD_K(\rho))^\top & -C_{p_1}^\top & -(C_{p_2}+D_K(\rho))^\top \\ I & 0 & 0 \\ \hdashline -B_p^\top(\rho) & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{array}\right] > 0. \tag{D.11}$$

Inserting (D.9) yields

$$\begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:ccc} 0 & Y(\rho) & & & \\ Y(\rho) & -\dot{Y}(\rho) & & & \\ \hdashline & & -\frac{1}{\gamma}I & 0 & 0 \\ & & 0 & \gamma I & 0 \\ & & 0 & 0 & \gamma I \end{array}\right] \left[\begin{array}{ccc} -(A(\rho)-B_uC_{p_2})^\top + \dots & -C_{p_1}^\top & -\gamma P(\rho)B_u \\ \dots \gamma P(\rho)B_uB_u^\top & & \\ I & 0 & 0 \\ \hdashline -B_p^\top(\rho) & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{array}\right] > 0.$$

Since in (D.10) we have a 2×2 block matrix, we will apply the Schur complement to the larger matrix inequality. We therefore expand (D.11) and obtain

$$\begin{bmatrix} -Y(\rho)(A(\rho)-B_uC_{p_2})^\top - (A(\rho)-B_uC_{p_2})Y(\rho) - \dot{Y}(\rho) + 2\gamma B_uB_u^\top + \frac{1}{\gamma}B_uB_u^\top & -YC_{p_1}^\top & -\gamma B_u \\ * & \gamma I & 0 \\ * & * & \gamma I \end{bmatrix},$$

which after a Schur complement with respect to the (3,3)-block yields exactly (D.10).

Convex LMI Conditions: Gridding We now derive convex LMI conditions to solve for $Y(\rho)$ while minimizing γ via gridding. Write (D.7) as

$$\begin{aligned} \mathcal{N}^\top \begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:c} 0 & Y(\rho) & \\ Y(\rho) & -\dot{Y}(\rho) & \\ \hdashline & & \gamma I \end{array}\right] \left[\begin{array}{cc} -A^\top(\rho) & -C_p^\top(\rho) \\ I & 0 \\ \hdashline 0 & I \end{array}\right] \mathcal{N} - \dots \\ \mathcal{N}^\top \begin{bmatrix} * & * \end{bmatrix}^\top \frac{1}{\gamma} \begin{bmatrix} -B_p^\top(\rho) & 0 \end{bmatrix} \mathcal{N} > 0, \quad \forall(\rho,\eta) \in P_\rho \times P_\eta \end{aligned} \tag{D.12}$$

and apply the Schur complement. This yields

$$\begin{bmatrix} \mathcal{N}^\top \begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:c} 0 & Y(\rho) & \\ Y(\rho) & -\dot{Y}(\rho) & \\ \hdashline & & \gamma I \end{array}\right] \begin{bmatrix} -A^\top(\rho) & -C_p^\top(\rho) \\ \multicolumn{2}{c}{\dashrule} \\ I & 0 \\ \multicolumn{2}{c}{\dashrule} \\ 0 & I \end{bmatrix} \mathcal{N} & \mathcal{N}^\top \begin{bmatrix} -B_p(\rho) \\ \multicolumn{1}{c}{\dashrule} \\ 0 \end{bmatrix} \\ \begin{bmatrix} -B_p^\top(\rho) & 0 \end{bmatrix} \mathcal{N} & \gamma I \end{bmatrix} > 0,$$

$$\begin{bmatrix} \mathcal{N}^\top & 0 \\ 0 & I \end{bmatrix} \left[\begin{array}{cc} \begin{bmatrix} * \\ - \\ * \end{bmatrix}^\top \left[\begin{array}{cc:c} 0 & Y(\rho) & \\ Y(\rho) & -\dot{Y}(\rho) & \\ \hdashline & & \gamma I \end{array}\right] \begin{bmatrix} -A^\top(\rho) & -C_p^\top(\rho) \\ \multicolumn{2}{c}{\dashrule} \\ I & 0 \\ \multicolumn{2}{c}{\dashrule} \\ 0 & I \end{bmatrix} & \begin{bmatrix} -B_p(\rho) \\ \multicolumn{1}{c}{\dashrule} \\ 0 \end{bmatrix} \\ \begin{bmatrix} -B_p^\top(\rho) & 0 \end{bmatrix} & \gamma I \end{array}\right] \begin{bmatrix} \mathcal{N} & 0 \\ 0 & I \end{bmatrix} > 0,$$

which eventually leads to

$$\begin{bmatrix} \mathcal{N}^\top & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} -\dot{Y}(\rho) - Y(\rho)A^\top(\rho) - A(\rho)Y(\rho) & -Y(\rho)C_p^\top(\rho) & -B_p(\rho) \\ * & \gamma I & 0 \\ * & * & \gamma I \end{bmatrix} \begin{bmatrix} \mathcal{N} & 0 \\ 0 & I \end{bmatrix} > 0, \quad \text{(D.13)}$$

When inserting $\dot{Y}(\rho) = \sum_{i=1}^{n_\rho} \eta_i \frac{\partial Y(\rho)}{\partial \rho_i}$, it becomes apparent that the inequality is affine in the parameters' rate of change. It therefore suffices to grid the the parameters ρ and consider only the extremal values of η.

Convex LMI Conditions: Polytopic Approach We now derive convex LMI conditions to solve for $Y(\rho)$ while minimizing γ via a polytopic multi-convexity approach, assuming that both Lyapunov function and LPV plant are depending affinely on some parameters θ. We take care of the special assumptions on the plant now and observe that only $A(\theta)$ is parameter-dependent. Including the multi-convexity condition

$$\frac{\partial^2}{\partial \theta^2} \left(Y(\theta)A^\top(\theta) + A(\theta)Y(\theta) \right) \geq 0, \forall \theta \in P_\theta$$

allows to solve

$$\begin{bmatrix} \mathcal{N}^\top & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} -\sum_{i=1}^{n_\theta} \eta_i \frac{\partial Y(\theta)}{\partial \theta_i} - Y(\theta)A^\top(\theta) - A(\theta)Y(\theta) & -Y(\theta)C_p^\top & -B_p \\ * & \gamma I & 0 \\ * & * & \gamma I \end{bmatrix} \begin{bmatrix} \mathcal{N} & 0 \\ 0 & I \end{bmatrix} > 0,$$
$$\forall (\theta, \nu) \in P_\theta \times P_\nu \quad \text{(D.14)}$$

in the vertices that span the convex hull of both the ranges of parameter values and the parameters' rate of change. Note that also the multi-convexity condition has to be evaluated in all the vertices of the parameter range.

Convex LMI Conditions: Full-Block $\mathcal{S}$-Procedure In order to derive convex LMI conditions to solve for $Y(\rho)$ while minimizing γ via the Full-Block $\mathcal{S}$-Procedure and multi-convexity conditions, use a quadratic factorization $Y(\rho) = T_Y^\top(\rho)\mathcal{Y}T_Y(\rho)$ for the

Lyapunov function. Here we assume that the dependeny of both Lyapunov and LPV plant can be described as LFT with some parameters δ.

It is the goal to write inequality (D.7) in the form $G(\delta)^\top Q G(\delta) > 0$, in order to straightforwardly apply the Full-Block $\mathcal{S}$-Procedure. This will result in a convex parameter-independent matrix inequality and a quadratic matrix inequality for the newly introduced multiplier. We will first derive an LFT representation for $G(\delta)$.

With $Y(\delta) = T_Y^\top(\delta)\mathcal{Y}T_Y(\delta)$ note that

$$\begin{bmatrix} 0 & Y(\delta) \\ Y(\delta) & -\dot{Y}(\delta) \end{bmatrix} = \begin{bmatrix} * \end{bmatrix}^\top \begin{bmatrix} 0 & \mathcal{Y} \\ \mathcal{Y} & 0 \end{bmatrix} \begin{bmatrix} T_Y(\delta) & -\dot{T}_Y(\delta) \\ 0 & T_Y(\delta) \end{bmatrix}.$$

From an LFT representation

$$T_Y(\delta) = \Delta_Y \star \left[\begin{array}{c|c} Y_{11} & Y_{12} \\ \hline Y_{21} & Y_{22} \end{array}\right],$$

one can get

$$\begin{bmatrix} T_Y(\delta) & -\dot{T}_Y(\delta) & 0 & 0 \\ 0 & T_Y(\delta) & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} = \begin{bmatrix} \dot{\Delta}_Y & & \\ & \Delta_Y & \\ & & \Delta_Y \end{bmatrix} \star \left[\begin{array}{ccc|ccc|c} 0 & 0 & Y_{11} & 0 & Y_{12} & 0 & 0 \\ Y_{11} & Y_{11} & 0 & -Y_{12} & 0 & 0 & 0 \\ 0 & 0 & Y_{11} & 0 & Y_{12} & 0 & 0 \\ \hline -Y_{21} & -Y_{21} & 0 & Y_{22} & 0 & 0 & 0 \\ 0 & 0 & Y_{21} & 0 & Y_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{array}\right]$$

$$= \tilde{\Delta}_Y \star \left[\begin{array}{c|c} \tilde{Y}_{11} & \tilde{Y}_{12} \\ \hline \tilde{Y}_{21} & \tilde{Y}_{22} \end{array}\right].$$

It is rather tedious to verify the above. A helpful identity can be

$$\dot{T}_Y(\delta) = Y_{21}(I - \Delta_Y Y_{11})^{-1}\dot{\Delta}_Y(I - Y_{11}\Delta_Y)^{-1}Y_{12}.$$

Together with the LFT representation

$$\begin{bmatrix} -A^\top(\delta) & -C_p^\top(\delta) \\ I & 0 \\ \hline -B_p^\top(\delta) & 0 \\ 0 & I \end{bmatrix} \mathcal{N} = \tilde{\Delta}_A \star \left[\begin{array}{c|c} \tilde{A}_{11} & \tilde{A}_{12} \\ \hline \tilde{A}_{21} & \tilde{A}_{22} \end{array}\right],$$

we can now find

$$G(\delta) = \begin{bmatrix} \tilde{\Delta}_Y & \\ & \tilde{\Delta}_A \end{bmatrix} \star \left[\begin{array}{cc|c} \tilde{Y}_{11} & \tilde{Y}_{12}\tilde{A}_{21} & \tilde{Y}_{12}\tilde{A}_{22} \\ 0 & \tilde{A}_{11} & \tilde{A}_{12} \\ \hline \tilde{Y}_{12} & \tilde{Y}_{22}\tilde{Y}_{21} & \tilde{Y}_{22}\tilde{A}_{22} \end{array}\right],$$

which can be constructed using the above formulae and should be reduced using the LFT tools provided by MATLAB, in order to find a small-in-size LFR representation

$$G(\delta) = \Delta \star \left[\begin{array}{c|c} G_{11} & G_{12} \\ \hline G_{21} & G_{22} \end{array}\right].$$

Inequality (D.7) can then be turned into the two inequalities

$$\left[\begin{array}{c} * \\ \hline * \end{array}\right]^\top \left[\begin{array}{c|cccc} M & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \mathcal{Y} & 0 & 0 \\ 0 & \mathcal{Y} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\gamma} I & 0 \\ 0 & 0 & 0 & 0 & \gamma I \end{array}\right] \left[\begin{array}{cc} G_{11} & G_{12} \\ I & 0 \\ \hline G_{21} & G_{22} \end{array}\right] > 0, \tag{D.15}$$

$$\left[\begin{array}{c} * \\ \hline * \end{array}\right]^\top M \left[\begin{array}{c} I \\ \hline \Delta \end{array}\right] < 0, \ \forall (\delta, \eta) \in P_\delta \times P_\eta. \tag{D.16}$$

Condition (D.16) is quadratic in Δ, but can be solved in the vertices of a convex hull including $P_\delta \times P_\eta$, if the multi-convexity conditions

$$M_{11} < 0, \quad M_{22} > 0, \quad \text{with } M = \left[\begin{array}{cc} M_{11} & M_{12} \\ M_{21} & M_{22} \end{array}\right]$$

are imposed, where the M_{ij} are sized conformably with Δ, for $i, j = 1, 2$. Note that a Schur complement is necessary to turn the above (sigmonial) inequality into an inequality linear in γ.

The important contraint $Y(\delta) > 0$ should not be forgotten either. For this purpose, recall that we factorized $Y(\delta) = T_Y^\top(\delta) \mathcal{Y} T_Y(\delta) > 0$, which is already in the form the Full-Block $\mathcal{S}$-Procedure can be applied to. Thus, with

$$T_Y(\delta) = \Delta_Y \star \left[\begin{array}{c|c} Y_{11} & Y_{12} \\ \hline Y_{21} & Y_{22} \end{array}\right],$$

we have that $Y(\delta) > 0$ if and only if there exists a multiplier N, such that

$$\left[\begin{array}{c} * \\ \hline * \end{array}\right]^\top \left[\begin{array}{c|c} N & 0 \\ \hline 0 & \mathcal{Y} \end{array}\right] \left[\begin{array}{cc} Y_{11} & Y_{12} \\ I & 0 \\ \hline Y_{21} & Y_{22} \end{array}\right] > 0$$

and

$$\left[\begin{array}{c} * \\ \hline * \end{array}\right]^\top N \left[\begin{array}{c} I \\ \hline \Delta_Y \end{array}\right] < 0, \ \forall (\delta) \in P_\delta.$$

Again, we consider solving the last inequality in the vertices of a convex hull including P_δ. We therefore require the multi-convexity conditions

$$N_{11} < 0, \quad N_{22} > 0, \quad \text{with } N = \left[\begin{array}{cc} N_{11} & N_{12} \\ N_{21} & N_{22} \end{array}\right],$$

where the N_{ij} are sized conformably with Δ_Y, for $i, j = 1, 2$.

Appendix E

Advanced Approaches

Appendix F

Graph Theory

Here some additional definitions and theorems useful for graph theory are given.

Definition F.1 (Incidence Matrix, [19]) *Given is a directed Graph $\mathcal{D} = (\mathcal{V}, \mathcal{E})$. If the directed edge $e_k = (v_i, v_j) \in \mathcal{E}$, then v_i is said to be the tail (where the arrow starts) and v_j is the head (where the arrow ends). The $n \times m$ incidence matrix $D(\mathcal{D})$ is then defined as*

$$D_{ik}(\mathcal{D}) = \begin{cases} -1 & \text{if } v_i \text{ is the tail of } e_k \\ 1 & \text{if } v_i \text{ is the head of } e_k \\ 0 & \text{otherwise.} \end{cases}$$

Lemma F.1 (Laplacian of Undirected Graphs, [19]) *Given is an undirected Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Assume an arbitrarily orientation to every edge in the edge set $\mathcal{E}$, the resulting oriented graph $\mathcal{G}^o$ is directed. The Laplacian of $\mathcal{G}$ can then be constructed by*

$$L(\mathcal{G}) = D(\mathcal{G}^o)D(\mathcal{G}^o)^T.$$

The Laplacian of $\mathcal{G}$ is symmetric and positive semidefinite, with its eigenvalues

$$0 = \lambda_1(\mathcal{G}) \leq \lambda_2(\mathcal{G}) \leq \ldots \leq \lambda_n(\mathcal{G}).$$

Theorem F.1 (Connectivity of an undirected graph, [19]) *The undirected graph $\mathcal{G}$ is connected, if and only if $\lambda_2(\mathcal{G}) > 0$.*

Proof: The eigenvector corresponding to $\lambda_1(\mathcal{G}) = 0$ is $\mathbf{1}$. Suppose there exist another vector $z \perp \mathbf{1}$, which is an eigenvector corresponding to the eigenvalue 0, such that

$$z^T L(\mathcal{G}) z = 0 \quad \Longleftrightarrow \quad z^T D(\mathcal{G}) D(\mathcal{G})^T z = 0 \quad \Longleftrightarrow \quad D(\mathcal{G})^T z = 0.$$

Thus we consider the If $(v_i, v_j) \in \mathcal{E}$, we get $z_j - z_i = 0$. However since $\mathcal{G}$ is connected, this implies $z_j = z_i$ for all $i, j \in \{1, \ldots, n\}$ and thus $z \parallel \mathbf{1}$. Thus the geometric multiplicity of the eigenvalue $\lambda_1(\mathcal{G}) = 0$ is one. Since $L(\mathcal{G})$ is symmetric the algebraic multiplicity is equal to the geometric and thus also one, what finishes the proof. ■

Theorem F.2 (Rank of a Directed Graph, [19]) *Given a directed graph $\mathcal{D}$. There exist at least one vertex with a path to all other vertices, if and only if* $\operatorname{rank} L(\mathcal{D}) = n - 1$.

Proof: The statement $\operatorname{rank} L(\mathcal{D}) = n - 1$ is equivalent to showing that the algebraic multiplicity of $\lambda_1 = 0$ is equal to one. Let us denote the the characteristic polynomial of $L(\mathcal{D})$ therefore, which is

$$p_{\mathcal{D}} = \lambda^n + \alpha_{n-1}\lambda^{n-1} + \cdots + \alpha_1\lambda + \alpha_0,$$

with $\alpha_0 = 0$ since zero is an eigenvalue of $L(\mathcal{D})$. The algebraic multiplicity of that zero eigenvalue is one if and only if $\alpha_1 \neq 0$. We know that

$$\alpha_1 = \sum_{v \in \mathcal{V}} \det L_v(\mathcal{D}),$$

where $L_v(\mathcal{D})$ is the matrix obtained from $L(\mathcal{D})$ by deleting the vth row and vth column. From Theorem F.3 it is known that $L_v(\mathcal{D}) \neq 0$ if there is is at least one possible path from v to all other vertices. Thus $\alpha_1 > 0$ if an only if there exists at least one $v \in \mathcal{V}$ with a path to all other vertives. This concludes the proof. ■

The following Theorem is a simplified version of Theorem 2.12 given in [19], given there without proof. The proof can be found in [33].

Theorem F.3 *Let v be an arbitrary vertex of a directed graph $\mathcal{D}$, then*

$$\det L_v(\mathcal{D}) = t(\mathcal{D}),$$

where $L_v(\mathcal{D})$ is the matrix obtained from $L(\mathcal{D})$ by deleting the vth row and vth column and $t(\mathcal{D})$ is the number of combinations of all possible cycle-free paths from v to all other vertices. Here cycle-free means that no vertex is allowed to appear more than once in one possible path.

Example F.1 *The content of Theorem F.3 is shown exemplarily for the directed graph in Fig. F.1 with the Laplacian*

$$L(\mathcal{D}) = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}.$$

There are 3 possibilites to reach vertices 2 and 3 starting from vertex 1 as shown in Fig. F.1. This is confirmed by

$$\det L_1(\mathcal{D}) = \det \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = 3.$$

There are no possibilities to reach the remaining vertices from vertex 2 or 3, which is confirmed by

$$\det L_2(\mathcal{D}) = \det L_3(\mathcal{D}) = \det \begin{bmatrix} 0 & 0 \\ -1 & 2 \end{bmatrix} = 0.$$

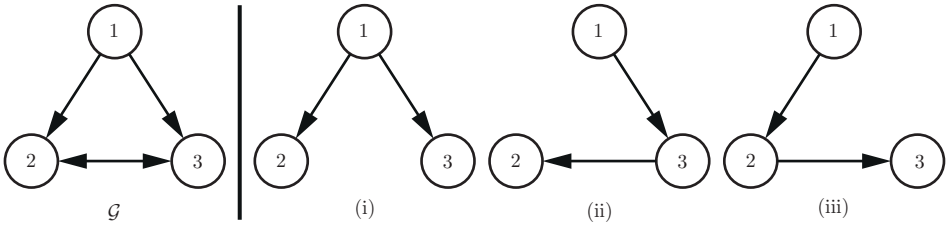


Figure F.1: Graph $\mathcal{D}$ and all possibilities (i), (ii) and (iii) to reach all other vertices starting from vertex 1

Appendix G

Quadrocopter

Quad-rotor helicopters (short quadrocopters) are a suitable and popular research platform for the application and testing of concepts for autonomous vehicles, such as e.g. formation control. Here the dynamic behavior of such a quad-rotor helicopter is sketched based on [29]:

A quadrocopter is a vertical take-off and landing (VTOL) aerial vehicle, the basic configuration is shown in Fig. G.1: Four rotors are mounted on a cross-shaped body, such that each of them exerts a thrust force directing upward perpendicular to the body plane. These forces add up to the total thrust force

$$F_{Th} = F_1 + F_2 + F_3 + F_4 \tag{G.1}$$

acting on the quadrocopter. The attitude of the body is described by the yaw angle ψ around the z-axis, the pitch angle θ around the y-axis and the roll angle ϕ around the x-axis. As a special property of the quadrocopter, within the x-y-plane the configuration does not imply any predominant flight direction. However, here we define the positive x-direction as "forward", to which also the naming of the attitude angles refers to.

A horizontal movement can be achieved by tilting the quadrocopter, such that the thrust vector gets a component in the x-y-plane and horizontally accelerates the quadrocopter. With respect to a fixed coordinate system (aligned such that $\psi = 0$), the forces on the center of gravity are

$$\begin{aligned} F_x &= -F_{Th} sin(\theta) \\ F_y &= F_{Th} cos(\theta) sin(\phi) \\ F_z &= F_{Th} cos(\theta) cos(\phi) - mg. \end{aligned} \tag{G.2}$$

Torques around the axes result from differences of the rotor speeds:

$$\begin{aligned} \tau_\phi &= b(F_4 - F_2) \\ \tau_\theta &= b(F_1 - F_3) \\ \tau_\psi &= \tau_{M1} + \tau_{M3} - \tau_{M2} - \tau_{M4}, \end{aligned} \tag{G.3}$$

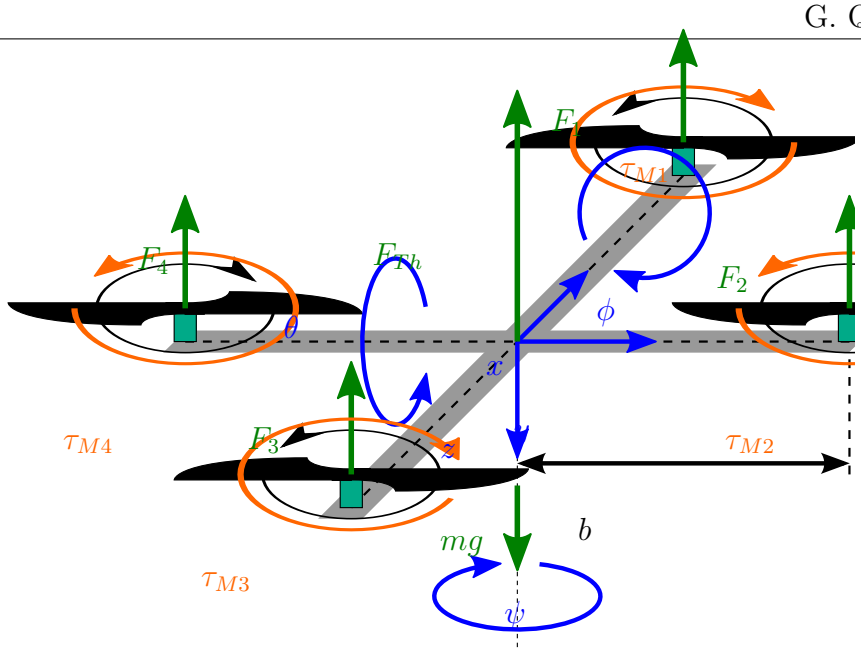


Figure G.1: Configuration of a quadrocopter

where τ_{Mi} denotes the moment around the z-axis exerted by rotor i on the body, b is the distance between rotor and center of gravity. Note that the directions of rotation are chosen such that $T_\psi = 0$ for equal speed of all rotors.

Using Newton's law $F = m\ddot{x}$, we can write

$$\begin{aligned} m\ddot{x} &= -F_{Th} \sin\theta \\ m\ddot{y} &= F_{Th} \cos\theta \sin\phi \\ m\ddot{z} &= F_{Th} \cos\theta \cos\phi - mg \\ \ddot{\psi} &= \tau_\psi \\ \ddot{\theta} &= \tau_\theta \\ \ddot{\phi} &= \tau_\phi, \end{aligned}$$

where $m = 0.64kg$ denotes the mass and $g = 9.81$ the gravity constant. Choosing a state vector

$$\xi = \begin{bmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} & \psi & \dot{\psi} & \theta & \dot{\theta} & \phi & \dot{\phi} \end{bmatrix}^T \tag{G.4}$$

and an input vector

$$u = \begin{bmatrix} F_{Th} - mg & \tau_\psi & \tau_\theta & \tau_\phi \end{bmatrix}^T, \tag{G.5}$$

we obtain a non-linear state space model $\dot{\xi} = f(\xi, u)$ as

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \\ \dot{\xi}_4 \\ \dot{\xi}_5 \\ \dot{\xi}_6 \\ \dot{\xi}_7 \\ \dot{\xi}_8 \\ \dot{\xi}_9 \\ \dot{\xi}_{10} \\ \dot{\xi}_{11} \\ \dot{\xi}_{12} \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \frac{-u_1}{m} \sin \xi_9 - g \sin \xi_9 \\ \xi_4 \\ \frac{u_1}{m} \cos \xi_9 \sin \xi_{11} + g \cos \xi_9 \sin \xi_{11} \\ \xi_6 \\ \frac{u_1}{m} \cos \xi_9 \cos \xi_{11} + g \cos \xi_9 \cos \xi_{11} - g \\ \xi_8 \\ u_2 \\ \xi_{10} \\ u_3 \\ \xi_{12} \\ u_4 \end{bmatrix}. \tag{G.6}$$

A linear state space model can be derived by Taylor series linearization around the equilibrium $\xi = 0$ with $f(0, 0) = 0$:

$$\begin{aligned} \dot{\xi} &= A\xi + Bu \\ y &= C\xi \end{aligned} \tag{G.7}$$

with

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{G.8}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{1}{m} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$$

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

Bibliography

[1] W. J. Rugh and J. S. Shamma, "A survey of research on gain-scheduling", *Automatica*, vol. 36, pp. 1401–1425, 2000.

[2] A. Eichler, C. Hoffmann and H. Werner, "Robust Stability Analysis of Interconnected Systems with Uncertain Time-Varying Time Delays via IQCs", in *52nd IEEE Conference on Decision and Control*, 2013.

[3] M. Vidyasagar, *Nonlinear Systems Analysis*, 2nd, ser. Classics in Applied Mathematics. 3600 University City Science Center, Philadelphia, PA 19104-2688: SIAM, 2002, ISBN: 0-89871-526-1.

[4] P. Apkarian, G. Becker, P. Gahinet and H. Kajiwara, *LMI Techniques in Control Engineering from Theory to Practice: Workshop Notes*. 1996.

[5] F. Wu, "Control of Linear Parameter Varying Systems", Ph.D. dissertation, University of Berkeley, California and USA, 1995.

[6] D. Stöckel, "Entwicklung eines Entwurfsverfahrens für die qualitative Regelung dynamischer Systeme", Ph.D. dissertation, 1994.

[7] J. Theis, C. Radisch and H. Werner, "Self-Scheduled Control of a Gyroscope", in *19th IFAC World Congress*, 2014, pp. 6129–6134.

[8] P. Apkarian and P. Gahinet, "A Convex Characterization of Gain-Scheduled H∞ Controllers", *IEEE Transactions on Automatic Control*, pp. 853–864, 1995.

[9] P. Apkarian and R. J. Adams, "Advanced gain-scheduling techniques for uncertain systems", *IEEE Transactions on Control Systems Technology*, vol. 6, no. 1, pp. 21–32, 1998.

[10] C. W. Scherer, "LPV Control and Full Block Multipliers", *Automatica*, vol. 37, no. 3, pp. 361–375, 2001. (visited on 03/04/2014).

[11] F. Wu and K. Dong, "Gain-Scheduling Control of LFT Systems Using Parameter-Dependent Lyapunov Functions", *Automatica*, vol. 42, no. 1, pp. 39–50, 2006.

[12] P. Apkarian, P. Gahinet and G. Becker, "Self-Scheduled H∞ Control of Linear Parameter-varying Systems: a Design Example", *Automatica*, vol. 31, no. 9, pp. 1251–1261, 1995.

[13] K. Zhou, J. Doyle and K. Glover, *Robust and Optimal Control*. Prentice-Hall, N.J., USA, 1996, ISBN: 0-13-456567-3.

[14] G. Kaiser, Q. Liu, C. Hoffmann, M. Korte and H. Werner, "Torque vectoring for an electric vehicle using an LPV drive controller and a torque and slip limiter", in *51st IEEE Conference on Decision and Control*, 2012, pp. 5016–5021.

[15] C. W. Scherer, "Robust Mixed Control and LPV Control with Full Block Scalings", in *Advances in Linear Matrix Inequality Methods in Control*, L. El Ghaoui and S.-I. Niculescu, Eds., SIAM, 2000, ISBN: 0-89871-438-9.

[16] ——, "Theory of Robust Control", Delft, Netherlands, 2001.

[17] A. Kwiatkowski and H. Werner, "PCA-Based Parameter Set Mappings for LPV Models With Fewer Parameters and Less Overbounding", *IEEE Transactions on Control Systems Technology*, vol. 16, no. 4, pp. 781–788, 2008.

[18] S. M. Hashemi, H. S. Abbas and H. Werner, "Low-Complexity Linear Parameter-Varying Modeling and Control of a Robotic Manipulator", *Control Engineering Practice*, vol. 20, pp. 248–257, 2012.

[19] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton and Oxford: Princeton University Press, 2010.

[20] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations", *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[21] A. Popov and H. Werner, "Robust Stability of a Multi-Agent System under Arbitrary and Time-Varying Communication Topologies and Communication Delays", *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2343–2347, 2012.

[22] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control - Analysis and Design*, 2nd. John Wiley & Sons, Ltd, 2005, ISBN: 978-0-470-01168-3.

[23] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays", *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[24] W. Ren and R. W. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*, ser. Communications and Control Engineering. London: Springer-Verlag London Limited, 2008, ISBN: 9781848000155.

[25] P. Massioni and M. Verhaegen, "Distributed Control for Identical Dynamically Coupled Systems: A Decomposition Approach", *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 124–135, 2009.

[26] A. Eichler and H. Werner, "Convergence bounds for discrete-time second-order multi-agent-systems", in *European Control Conference*, 2013.

[27] P. Massioni and M. Verhaegen, "A full block S-procedure application to distributed control", in *American Control Conference*, 2010.

[28] C. Hoffmann, A. Eichler and H. Werner, "Distributed Control of Linear Parameter-Varying Decomposable Systems", in *American Control Conference*, 2013, pp. 2386–2391.

[29] D. Lara, A. Sanchez, R. Lozano and P. Castillo, "Real-Time Embedded Control System for VTOL Aircrafts: Application to stabilize a quad-rotor helicopter", in *International Conference on Control Applications*, 2006.

[30] U. Pilz, A. Popov and H. Werner, "An Information Flow Filter Approach to Cooperative Vehicle Control", in *18th World Congress*, 2011, pp. 7432–7437.

[31] H. Werner, "Linear Matrix Inequalities for Applications in Control Engineering", *IEE Control Theory and Applications*, p. 3, 2003.

[32] R. D'Andrea and G. E. Dullerud, "Distributed Control Design for Spatially Interconnected Systems", *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1478–1495, 2003.

[33] W. T. Tutte, *Graph Theory As I Have Known It*, ser. Oxford Lecture Series in Mathematics and Its Applications,11. Oxford: OUP Oxford, 2012, ISBN: 978-0199660551.