Lecture Notes

Optimal and Robust Control

Herbert Werner

TUHH
Hamburg University of Technology

ICS
Institute of Control Systems

Contents

III   APPENDICES   167

Part I

LINEAR QUADRATIC OPTIMAL CONTROL

Introduction

In the first chapter, the linear regulator problem with finite time horizon is introduced. This is a basic problem, and from its solution the solution to other problems - e.g. the tracking problem - can be derived. In order to solve the regulator problem, we employ the principle of optimality and the Hamilton-Jacobi equation, which for the linear regulator problem takes the form of the Riccati equation. In this case, the optimal control law takes the form of linear state feedback, and it is initially assumed that all states are available for feedback. In chapter 6, we turn to the important case of infinite time horizon problems, where the solution is particularly simple. It will be shown that the optimal controller possesses attractive engineering properties such as good robustness margins.

The assumption that all states are available for feedback is removed in chapter 9 and replaced by the more realistic assumption that only a linear combination of state variables - the output variables - are measured. In order to be able to use the results derived in the first part, the state variables have to be estimated and state feedback has to be replaced by state estimate feedback. If the measurement of the output variables is corrupted by noise (which is always the case in real life problems), state estimation involves a trade-off between a fast decay rate of estimation errors and a low noise sensitivity of the closed-loop system. The Kalman Filter is shown to be the "optimal solution" to the estimation problem, optimal in the sense of a probabilistic problem description and minimal variance of the estimation error.

Unfortunately, the attractive robustness properties of the optimal state feedback solution derived in the first part may be lost when state feedback is replaced by state estimate feedback. A technique known as *loop transfer recovery* is presented, that can be used to asymptotically "recover" these properties, at the expense of increased noise sensitivity.

Chapter 1

The Linear Optimal Regulator Problem

Consider the following control problem. The plant to be controlled is represented by a linear state space model

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1.1}$$

$$y(t) = Cx(t). \tag{1.2}$$

The objective is to keep the state variables at desired values, and we can assume without loss of generality that the desired state vector is the zero vector (see Exercise 1.1). Assume that at time $t = t_0$ an external disturbance drives the system into a state $x(t_0) = x_0 \neq 0$, and that the controller is required to bring the system back to the zero state as quickly as possible. One possible criterion that expresses how fast an initial state is reduced to the zero state is the quadratic integral criterion

$$\int_{t_0}^{T} x^T(t)Qx(t)\,dt, \tag{1.3}$$

where Q is a symmetric, positive semi-definite matrix ($Q = Q^T \geq 0$). The term $x^T(t)Qx(t)$ is a measure of the extent to which the state at time t deviates from the zero state, and the weighting matrix Q determines how much weight is attached to each of the components of the state. The integral is a measure for the deviation of x from zero during the interval $[t_0, T]$.

In many cases the state variables themselves do not have any physical significance, and it may be desirable to penalize instead the deviation of a vector

$$z(t) = Dx(t)$$

whose entries are linear combinations of the state variables and do have some significance (a possible choice is $z(t) = y(t)$). It is then easy to see that

$$\int_{t_0}^{T} z^T(t)\tilde{Q}z(t)\,dt, \tag{1.4}$$

is equivalent to (1.3) with $Q = D^T \tilde{Q} D$.

Trying to find an optimal control input $u(t)$ that minimizes (1.3) or (1.4) leads to the problem that the cost can be made arbitrarily small when the control amplitudes are allowed to become infinitely large. A trade-off between fast disturbance rejection and reasonable control effort is possible when a second term, reflecting control energy, is included in the cost function

$$\int_{t_0}^{T} x^T(t)Qx(t) + u^T(t)Ru(t) \, dt, \qquad R = R^T > 0 \tag{1.5}$$

With a control input that makes (1.5) small, the final state $x(T)$ will in general not be the zero state, and with the above formulation of the control objective one has no influence on how close the final state is to zero. For this reason, it is convenient to add a third term to the performance index that penalizes the "distance" of the final state from zero

$$\int_{t_0}^{T} x^T(t)Qx(t) + u^T(t)Ru(t) \, dt + x^T(T)Sx(T), \qquad S = S^T \geq 0. \tag{1.6}$$

The problem of finding a control input $u(t)$, $t_0 \leq t \leq T$ that minimizes (1.6) under the constraint (1.1) is called the *linear optimal regulator problem*. This problem and its solution will be studied in the following chapters. At this point, one comment is in order. The optimal solution to the control problem described here is "optimal" only in the sense of the performance index (1.6) specified by the quantities Q, R and S, and one can view these quantities as convenient "tuning parameters" which can be used in an interactive procedure of trial and error to improve the behaviour of the control system.

Exercises

Problem 1.1

Consider a DC motor with shaft angular velocity ω that is controlled by the input voltage $u(t)$. The system can be described by the scalar differential equation

$$\dot{\omega}(t) = -a\omega(t) + bu(t),$$

where a and b are given constants. Consider the problem of stabilizing the angular velocity at a desired value ω_0. Assume that u_0 is the steady state input that keeps the plant in equilibrium at $\omega = \omega_0$. Formulate the problem of finding a control input that brings the system from an initial state $\omega(t_0) = \omega_1$ back to ω_0 quickly and with reasonable control effort, as a linear optimal regulator problem in the form of (1.6). For this purpose, transform the problem into an equivalent problem with zero equilibrium state by introducing a new state vector $\tilde{\omega} = \omega - \omega_0$.

Chapter 2

The Principle of Optimality

In order to solve the linear optimal regulator problem, we will first study a wider class of optimization problems, and derive a partial differential equation (the *Hamilton-Jacobi equation*) which is satisfied by the optimal performance index.

Consider the following optimal control problem. For the system

$$\dot{x} = f(x, u, t), \qquad x(t_0) = x_0 \tag{2.1}$$

find the optimal control $u^*(t)$, $t \in [t_0, T]$, that minimizes

$$V(x(t_0), u(\cdot), t_0) = \int_{t_0}^{T} l(x(\tau), u(\tau), \tau) \, d\tau + m(x(T)). \tag{2.2}$$

Here f, l and m are sufficiently smooth functions which can be essentially arbitrary (they will later be chosen to match the linear optimal regulator problem). The performance index V depends on initial state $x(t_0)$ and time t_0 and on the control input $u(t)$ for $t_0 \le t \le T$.

To simplify notation, we will write $u[a, b]$ to denote the function $u(\cdot)$ restricted to the interval $[a, b]$, and we sometimes write x_T for $x(T)$. Now we define the optimal performance index

$$V^*(x(t), t) = \min_{u[t, T]} V(x(t), u(\cdot), t). \tag{2.3}$$

Notice that V^* is independent of the control input, because it is defined to be the cost incurred by the optimal control input u^*. Notice also that we have replaced the initial time t_0 with t, because we want to study the optimal cost for different initial times.

To solve the optimal control problem, we will use the principle of optimality:

> "An optimal trajectory has the property that at an intermediate point, no matter how it was reached, the rest of the trajectory must coincide with an optimal trajectory as computed from this intermediate point as the initial point".
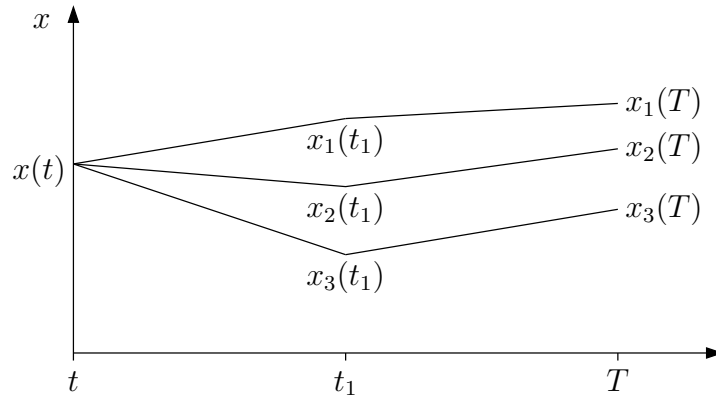
Figure 2.1: Principle of optimality

This may seem almost self-evident, but it is nevertheless a very powerful idea. The principle of optimality was first formulated in 1697 by James Bernoulli and restated by Euler in 1744, but then largely ignored until it was revived in the 1930s and formulated in the above form by Richard Bellman in 1957.

We will now discuss how this idea can be applied to the optimal control problem. First we cut the time interval $[t, T]$ in two subintervals, and consider the two control inputs $u[t, t_1]$ and $u[t_1, T]$. Since $u[t, T]$ is the concatenation of $u[t, t_1]$ and $u[t_1, T]$, minimizing over $u[t, T]$ is equivalent to minimizing over $u[t, t_1]$ and $u[t_1, T]$. Thus

$$V^*(x(t), t) = \min_{u[t,T]} \int_t^T l(x, u, \tau)\, d\tau + m(x_T)$$

$$= \min_{u[t,t_1]} \min_{u[t_1,T]} \left(\int_t^{t_1} l(x, u, \tau)\, d\tau + \int_{t_1}^T l(x, u, \tau)\, d\tau + m(x_T) \right).$$

The inner minimum depends on $u[t, t_1]$, but the first summand is independent of $u[t_1, T]$, so the cost can be rearranged as

$$V^*(x(t), t) = \min_{u[t,t_1]} \left(\int_t^{t_1} l(x, u, \tau)\, d\tau + \min_{u[t_1,T]} \left(\int_{t_1}^T l(x, u, \tau)\, d\tau + m(x_T) \right) \right)$$

or

$$V^*(x(t), t) = \min_{u[t,t_1]} \left(\int_t^{t_1} l(x, u, \tau)\, d\tau + V^*(x(t_1), t_1) \right). \tag{2.4}$$

The minimization required in (2.4) is illustrated in Fig. 2.1. All trajectories from t_1 to T are optimal. The value of the cost function $V(x(t), t)$ for the whole interval $[t, T]$ is obtained by adding the cost of the trajectory form t to t_1, to the optimal cost from t_1 to T. Both depend on $u[t, t_1]$, which must be chosen to minimize the sum.

Chapter 3

The Hamilton-Jacobi Equation

To solve the optimal control problem posed in chapter 1, the principle of optimality is applied with setting $t_1 = t + \Delta t$ and taking the limit as $\Delta t \to 0$. This leads to a partial differential equation for the optimal performance index, the Hamilton-Jacobi equation.

Replacing t_1 by $t + \Delta t$ in (2.4) gives

$$V^*(x(t), t) = \min_{u[t,t+\Delta t]} \left(\int_t^{t+\Delta t} l(x, u, \tau)\, d\tau + V^*(x(t + \Delta t), t + \Delta t) \right). \qquad (3.1)$$

The first term on the right hand side can be written as

$$\int_t^{t+\Delta t} l(x, u, \tau)\, d\tau = \Delta t\, l(x(t + \alpha \Delta t), u(t + \alpha \Delta t), t + \alpha \Delta t)$$

with α some constant between 0 and 1. The second term can be expanded into a Taylor series

$$V^*(x(t + \Delta t), t + \Delta t) =$$

$$V^*(x(t), t) + \left[\frac{\partial V^*}{\partial x}(x(t), t) \right]^T \dot{x}(t)\Delta t + \frac{\partial V^*}{\partial t}(x(t), t)\Delta t + O(\Delta t^2).$$

Substituting these expressions in (3.1) and writing l_α for the $l(\dots)$ term at time $t + \alpha \Delta t$, we obtain

$$V^*(x(t), t) =$$

$$\min_{u[t,t+\Delta t]} \left(\Delta t\, l_\alpha + V^*(x(t), t) + \left[\frac{\partial V^*}{\partial x}(x(t), t) \right]^T \dot{x}(t)\Delta t + \frac{\partial V^*}{\partial t}(x(t), t)\Delta t + O(\Delta t^2) \right).$$

The term $V^*(x(t), t)$ can now be eliminated on both sides. Moreover, the fourth summand is independent of $u[t, t+\Delta t]$ and can be pulled out of the minimization; thus, after dividing by Δt, we obtain

$$\frac{\partial V^*}{\partial t}(x(t), t) = - \min_{u[t,t+\Delta t]} \left(l_\alpha + \left[\frac{\partial V^*}{\partial x} \right]^T f(x, u, t) + O(\Delta t) \right).$$

Finally, after taking the limit as $\Delta t \to 0$ we have

$$\frac{\partial V^*}{\partial t}(x(t), t) = -\min_{u(t)} \left(l(x, u, t) + \left[\frac{\partial V^*}{\partial x}\right]^T f(x, u, t) \right). \qquad (3.2)$$

Here, f and l are known functions of their arguments x, u, t, whereas the function V^* is unknown. Equation (3.2) is already a form of the Hamilton-Jacobi equation, even though it does not yet look like a partial differential equation.

To bring (3.2) in the form of a first order partial differential equation, we define $\bar{u}$ to be the control input that minimizes the right hand side. The minimizing input depends on the values of $x(t)$, $\partial V^*/\partial x$ and t, so we write $\bar{u}(x, \partial V^*/\partial x, t)$. Note that the input $\bar{u}$ that minimizes the right hand side of (3.2), minimizes at the same time the performance index $V(x(t), u(\cdot), t)$, because from the expanded form of V, only terms independent of u have been removed.

With this definition of $\bar{u}$, (3.2) becomes

$$\frac{\partial V^*}{\partial t}(x(t), t) = -l(x, \bar{u}(x, \frac{\partial V^*}{\partial x}, t), t) - \left[\frac{\partial V^*}{\partial x}\right]^T f(x, \bar{u}(x, \frac{\partial V^*}{\partial x}, t), t). \qquad (3.3)$$

This is a first order partial differential equation with independent variable V^* and dependent variables x and t, because l, f and $\bar{u}$ are known functions of their arguments. From the definition (2.2) of the performance index follows the boundary condition

$$V^*(x(T), T) = m(x(T)). \qquad (3.4)$$

Equations (3.3) and (3.4) together form the Hamilton-Jacobi equation.

Let us summarize what we have achieved so far. In order to solve the optimal control problem (2.1), (2.2), we used the principle of optimality to derive a partial differential equation for the optimal cost V^*, the Hamilton-Jacobi equation. The Hamilton-Jacobi equation contains the control input $\bar{u}$ that minimizes the right hand side of (3.2), and $\bar{u}$ is a function of x, $\frac{\partial V^*}{\partial x}$ and t. The optimal control problem can then be solved in three steps:

1. find the minimizing $\bar{u}$ as a function of x, $\frac{\partial V^*}{\partial x}$ and t

2. substitute this expression in (3.3) and solve the Hamilton-Jacobi equation for V^*

3. compute the gradient $\frac{\partial V^*}{\partial x}$ and the optimal control $u^*(t) = \bar{u}(x(t), \frac{\partial V^*}{\partial x}, t)$.

An important observation at this point is that the optimal control input at time t is given as a function of the state vector; this means that the optimal control law has the form of

state feedback. The optimal control law can be obtained via the solution to the Hamilton-Jacobi equation: if such a solution exists, it is the optimal performance index. A problem with this approach is that a solution to the Hamilton-Jacobi equation may not exist, and that not every optimal performance index satisfies the Hamilton-Jacobi equation, so the equation represents only a sufficient condition for the optimal performance index. Moreover, it is in general difficult to solve this partial differential equation. The situation is different however when the above approach is applied to the linear regulator problem: in that case the Hamilton-Jacobi equation represents a necessary and sufficient condition, and the solution can be obtained via the solution to a nonlinear differential equation, the Riccati Differential Equation. This is shown in the following chapter.

Exercises

Problem 3.1

Consider the system

$$\dot{x} = u$$

and the cost function

$$V(x(0), u(\cdot), 0) = \int_0^T \left(x^2 + u^2 + \frac{1}{2} x^4 \right) dt.$$

Write down the Hamilton-Jacobi equation for this problem.

Chapter 4

The Matrix Riccati Equation

In this chapter, we shall use the results of the previous chapter to derive the solution to the linear optimal regulator problem posed in chapter 1. This will be done in two steps:

1. We show that if $V^*(x(t), t)$ exists, it must be of the form $x^T(t)P(t)x(t)$, where $P(t)$ is a positive semidefinite, symmetric matrix

2. Using the above result, we show that if $V^*(x(t), t)$ exists, it satisfies the so called Riccati differential equation.

Finally we outline a proof of existence of V^*, and find the optimal control law in terms of the solution to the Riccati differential equation.

Proof

Step 1

The first step is to show that V^* must be of the form $x^T P x$. This is called a quadratic form, and we will use the following result: the function $V^*(x)$ is a quadratic form if and only if it is continuous in x and

$$V^*(\lambda x) = \lambda^2 V^*(x) \qquad \text{for all real } \lambda \tag{4.1}$$

$$V^*(x_1) + V^*(x_2) = \frac{1}{2}(V^*(x_1 + x_2) + V^*(x_1 - x_2)). \tag{4.2}$$

To show that these two conditions are necessary for V^* to be a quadratic form is left as an exercise; the proof of sufficiency is more involved and will not be discussed here. Now we have to show that V^* satisfies (4.1) and (4.2). Let u_x^* denote the optimal control when the initial state is x. Because the plant model is linear and the performance index quadratic in x, we have

$$V(\lambda x, \lambda u_x^*, t) = \lambda^2 V^*(x, t)$$

and

$$V^*(\lambda x, t) = \lambda^2 V(x, \frac{1}{\lambda} u_{\lambda x}^*, t).$$

Because the optimum is minimal, we also have

$$V^*(\lambda x, t) \leq V(\lambda x, \lambda u_x^*, t)$$

and

$$\lambda^2 V^*(x, t) \leq \lambda^2 V(x, \frac{1}{\lambda} u_{\lambda x}^*, t).$$

Putting the equalities and inequalities together, we obtain

$$V^*(\lambda x, t) \leq \lambda^2 V^*(x, t)$$
$$\lambda^2 V^*(x, t) \leq V^*(\lambda x, t).$$

These two inequalities imply that $V^*(x, t)$ satisfies the first condition (4.1). That it also satisfies the second condition can be shown in a similar manner. Because $V^*(x, t)$ is continuous in x, it follows that it has the form

$$V^*(x(t), t) = x^T(t) P(t) x(t)$$

for some matrix $P(t)$. Positive semidefiniteness follows from the fact that $V^*(x(t), t)$ cannot be negative. To justify the claim that P is symmetric, assume it is not, then one can replace it by the symmetric matrix $1/2(P + P^T)$ without altering the value of V^*.

Step 2

Next we show that the matrix $P(t)$ that appeared in the last step, satisfies a particular matrix differential equation, the Riccati differential equation (RDE). We start with the first form of the Hamilton-Jacobi equation (3.2)

$$\frac{\partial V^*}{\partial t}(x(t), t) = -\min_{u(t)} \left(l(x, u, t) + \left[\frac{\partial V^*}{\partial x} \right]^T f(x, u, t) \right).$$

When applied to the linear regulator problem, we have

$$l(x, u, t) = x^T Q x + u^T R u;$$

and we know that $V^* = x^T P x$ for some $P(t)$, thus

$$\left[\frac{\partial V^*}{\partial x} \right]^T = 2x^T P.$$

Furthermore

$$f(x, u, t) = Ax + Bu,$$

and we have

$$\frac{\partial V^*}{\partial t} = x^T \dot{P} x.$$

Substitution of these expressions in the Hamilton-Jacobi equation yields

$$x^T \dot{P} x = -\min_u (x^T Q x + u^T R u + 2x^T P A x + 2x^T P B u). \tag{4.3}$$

The minimizing control input $\bar{u}$ can be found by setting the partial derivative of the right hand side expression in (4.3) to zero. An alternative way - completing the squares - is shown in Exercise 4.3. The solution is

$$\bar{u}(t) = -R^{-1}B^T P(t)x(t). \tag{4.4}$$

If this is substituted for $\bar{u}$ in (4.3), we obtain

$$x^T \dot{P} x = -x^T(PA + A^T P - PBR^{-1}B^T P + Q)x.$$

Since this equation holds for all x, we have

$$-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q. \tag{4.5}$$

This is the celebrated matrix Riccati Equation (or Riccati differential equation, to distinguish it from the algebraic Riccati equation discussed in chapter 6). A boundary condition for (4.5) follows from the boundary condition (3.4) for the Hamilton-Jacobi equation

$$V^*(x(T), T) = x^T(T)P(T)x(T) = x^T(T)Sx(T)$$

or

$$P(T) = S, \tag{4.6}$$

where S is the penalty on the final state vector introduced in (1.6).

From the discussion in the previous chapter we know that $\bar{u}$ is the optimal control input at time t, thus solving the Riccati equation (4.5) and substituting the solution $P(t)$ in (4.4) yields the optimal controller in the form of linear, time-varying state feedback

$$u^*(t) = F(t)x(t) \tag{4.7}$$
$$F(t) = -R^{-1}B^T P(t). \tag{4.8}$$

The above results have been derived under the assumption that a solution V^* to the minimization problem exists. We now briefly indicate how one can prove the existence of V^*. From the boundary condition (4.6) we know that there exists a matrix $P(t)$ at $t = T$ and (by continuity) for $t < T$ when t is sufficiently close to T. The only way a solution can fail to exist is that there is a finite time $T_1 < T$ at which some entries of P become infinite (in that case T_1 is called a *finite escape time*). However, V^* can never be greater than any V resulting from an arbitrary control input; so we can choose the zero control input and use the fact that the state vector cannot become unbounded in finite time to show that elements of P cannot become infinite in finite time. ∎

Exercises

Problem 4.1

Consider the problem of minimizing

$$V = \int_0^T (x^2 + u^2)dt$$

for the first order system

$$\dot{x} = u.$$

Determine the solution $P(t)$ of the matrix Riccati equation using separation of variables and hence compute the optimal controller for this problem. Plot the time response of $P(t)$.

Problem 4.2

Define the matrix X as

$$X = \begin{bmatrix} M & L \\ L^T & N \end{bmatrix}$$

Show that:

$$X > 0 \quad \text{is equivalent to} \qquad N > 0 \quad \text{and} \quad M - LN^{-1}L^T > 0$$
$$\text{or} \qquad\qquad\qquad M > 0 \quad \text{and} \quad N - L^T M^{-1}L > 0$$

where $M = M^T$ and $N = N^T$, and L is a matrix with appropriate dimension.

Problem 4.3   Completing squares

a)   Consider the function $W(x, u)$ defined as

$$W(x, u) = \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} M & L \\ L^T & N \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}$$

and assume that $N = N^T > 0$. Show that

$$\min_u W(x, u) = x^T(M - LN^{-1}L^T)x$$

and

$$\bar{u} = \arg\min_u W(x, u) = -N^{-1}L^T x.$$

Hint: Find a nonsingular matrix T such that

$$T^T \begin{bmatrix} M & L \\ L^T & N \end{bmatrix} T$$

is block diagonal, and use this matrix to write $W(x, u)$ as a sum of two terms of which one is independent of u and the other one is nonnegative.

b)   Use the above result to show that $\bar{u}$ in (4.4) minimizes the right hand side of (4.3).

Problem 4.4 Riccati Differential Equation

Consider the system

$$\dot{x} = Ax + Bu, \quad x(t_0) = x_0 \neq 0.$$

Prove that:

the optimal control input $u^*(t)$, $t_0 \geq t \geq T$ that minimizes the performance index V in (2.3) is

$$u^*(t) = -R^{-1}B^T P(t)x(t),$$

where $P(t) = P^T(t) \geq 0 \quad \forall t \geq 0$ satisfies

$$-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q$$
$$P(T) = S.$$

Hint: Start with an arbitrary $u(t) = -R^{-1}B^T P(t)x(t) + v(t)$, where the choice of $P(t)$ and $v(t)$ are arbitrary.

Make use of the fact that

$$\int_{t_0}^{T} \frac{d}{dt}(x^T P x)dt = \int_{t_0}^{T} (\dot{x}^T P x + x^T \dot{P} x + x^T P \dot{x})dt$$

Chapter 5

Solution of the Riccati Equation

In this chapter, we show how the Riccati differential equation

$$-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q.$$

with boundary condition

$$P(T) = S,$$

can be solved in cases more general than the simple example at the end of the last chapter. First, we show that the Riccati equation can be solved by solving the linear system

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{5.1}$$

with boundary condition

$$\begin{bmatrix} X(T) \\ Y(T) \end{bmatrix} = \begin{bmatrix} I \\ S \end{bmatrix}, \tag{5.2}$$

where $X(t)$ and $Y(t)$ are square matrices of the same size as A. Having solved the above system for X and Y, one can compute the solution of the Riccati equation as

$$P(t) = Y(t)X^{-1}(t). \tag{5.3}$$

To prove that (5.3) is indeed the solution to the Riccati equation, we compute its derivative

$$\frac{dP}{dt} = \frac{dYX^{-1}}{dt} = Y\frac{dX^{-1}}{dt} + \frac{dY}{dt}X^{-1}. \tag{5.4}$$

Next, differentiate both sides of

$$X(t)X^{-1}(t) = I$$

to get

$$\frac{dX^{-1}}{dt} = -X^{-1}\frac{dX}{dt}X^{-1}.$$

Substituting this and the derivatives of X and Y given in (5.1) in (5.4) gives

$$\frac{dYX^{-1}}{dt} = -A^TYX^{-1} - YX^{-1}A - Q + YX^{-1}BR^{-1}B^TYX^{-1},$$

and it is clear that $P = YX^{-1}$ satisfies the Riccati equation (4.5) and (4.6).

When the matrices A, B, Q and R are constant, one can compute the solution to the Riccati equation in terms of the transition matrix of system (5.1). So far we have made no statement about these matrices being constant or time-varying; in fact the results we have seen so far hold for both cases. From now on, we assume that these matrices are constant.

The matrix

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}$$

is called the *Hamiltonian matrix* and plays an important role in linear quadratic optimisation. It has the property that if λ is an eigenvalue of H, then so is $-\lambda$. To see this, consider the nonsingular transformation matrix

$$J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}.$$

It is straightforward to verify that $-H = JH^TJ^{-1}$. We assume that H has no eigenvalues on the imaginary axis (we will see in chapter 7 under which conditions this is true). Then there exists a nonsingular transformation U such that

$$U^{-1}HU = \begin{bmatrix} \Lambda_s & 0 \\ 0 & \Lambda_u \end{bmatrix}$$

where Λ_s is a matrix whose eigenvalues are the stable eigenvalues of H, and Λ_u is a matrix with the unstable eigenvalues of H. When U is partitioned as

$$U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix},$$

then the columns of

$$\begin{bmatrix} U_{11} \\ U_{21} \end{bmatrix}$$

span the eigenspace corresponding to the stable eigenvalues of H, whereas the columns of

$$\begin{bmatrix} U_{12} \\ U_{22} \end{bmatrix}$$

span the eigenspace corresponding to the unstable eigenvalues. (When the matrices Λ_s and Λ_u are diagonal, the columns of U are the eigenvectors of H.) Applying the transformation

$$\begin{bmatrix} X \\ Y \end{bmatrix} = U \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix} \tag{5.5}$$

to (5.1) yields

$$\begin{bmatrix} \dot{\tilde{X}} \\ \dot{\tilde{Y}} \end{bmatrix} = \begin{bmatrix} \Lambda_s & 0 \\ 0 & \Lambda_u \end{bmatrix} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix}. \tag{5.6}$$

From the decoupled equations, the solution at time T can be computed in terms of the solution at time t as

$$\tilde{X}(T) = e^{\Lambda_s(T-t)}\tilde{X}(t)$$
$$\tilde{Y}(T) = e^{\Lambda_u(T-t)}\tilde{Y}(t).$$

From (5.5) evaluated at $t = T$ we have

$$I = U_{11}\tilde{X}(T) + U_{12}\tilde{Y}(T)$$
$$S = U_{21}\tilde{X}(T) + U_{22}\tilde{Y}(T)$$

which can be used to solve for $\tilde{Y}(T)$: defining

$$G = -(U_{22} - SU_{12})^{-1}(U_{21} - SU_{11})$$

we obtain

$$\tilde{Y}(T) = G\tilde{X}(T).$$

Then again using (5.5), this time evaluated at t, and substituting the decoupled solutions of (5.6) yields

$$X(t) = (U_{11} + U_{12}e^{-\Lambda_u(T-t)}Ge^{\Lambda_s(T-t)})e^{-\Lambda_s(T-t)}\tilde{X}(T)$$
$$Y(t) = (U_{21} + U_{22}e^{-\Lambda_u(T-t)}Ge^{\Lambda_s(T-t)})e^{-\Lambda_s(T-t)}\tilde{X}(T).$$

These expressions can be substituted for $X(t)$ and $Y(t)$ in (5.3) to compute the solution to the Riccati differential equation

$$P(t) = Y(t)X^{-1}(t)$$
$$= (U_{21} + U_{22}e^{-\Lambda_u(T-t)}Ge^{\Lambda_s(T-t)})(U_{11} + U_{12}e^{-\Lambda_u(T-t)}Ge^{\Lambda_s(T-t)})^{-1}. \tag{5.7}$$

Example 5.1

Solve the problem of the exercise 4.1 using (5.7).

Solution: From

$$\dot{x} = u$$

and

$$V(x(0), u(\cdot), 0) = \int_0^T (x^2 + u^2)\, dt.$$

we have the Hamilton matrix

$$H = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}.$$

The eigenvalues are $\lambda_s = -1$ and $\lambda_u = 1$, and the transformation matrix is

$$U = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

With $S = 0$, we have $G = 1$, and (5.7) becomes

$$P(t) = \frac{1 - e^{-2(T-t)}}{1 + e^{-2(T-t)}}$$

as before.

Exercises

Problem 5.1
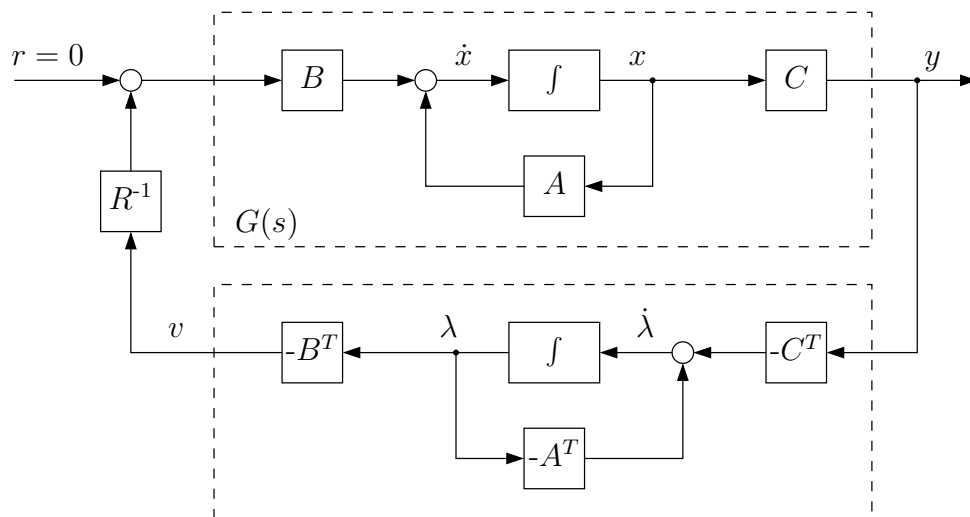
Consider the system shown in Figure 5.1.

Figure 5.1: System Block diagram

a) What is the open-loop transfer function from y to v, and how is it related to $G(s)$?

b) Find a state space model of the closed loop system.

   Hint: choose a state vector of the form: $\begin{bmatrix} x(t) \\ \lambda(t) \end{bmatrix}$.

Problem 5.2 Consider again the speed control problem of Exercise 1.1, with plant equation

$$\dot{\omega}(t) = -a\omega(t) + bu(t)$$

and cost function
$$\int_0^T (\omega^2(t) + \rho u^2(t))\, dt + s\omega^2(T).$$

The numerical values are $a = 0.5 \text{ sec}^{-1}$, $b = 150 \text{ rad}/(\text{Vsec}^2)$ and $T = 1$ sec. Use (5.7) to compute the optimal control law and plot the optimal state feedback gain $f(t)$ and the speed $\omega(t)$ in the interval $0 \text{ sec} \le t \le 1 \text{ sec}$ for the following values of ρ and s

a) $\rho = 10^2, \quad s = 0$

b) $\rho = 10^3, \quad s = 0$

c) $\rho = 10^4, \quad s = 0$

d) $\rho = 10^3, \quad s = 0.19.$

Chapter 6

The Infinite-Time Regulator Problem

The optimal state feedback gain even for a time-invariant plant is time varying because the solution $P(t)$ of the Riccati equation is time-varying. Fig. 6.1 shows the solution of the Riccati equation for the example in Exercise 5.2 with $\rho = 1000$, $s = 0$ and different time horizons T, ranging from $T = 1$ to $T = 10$.
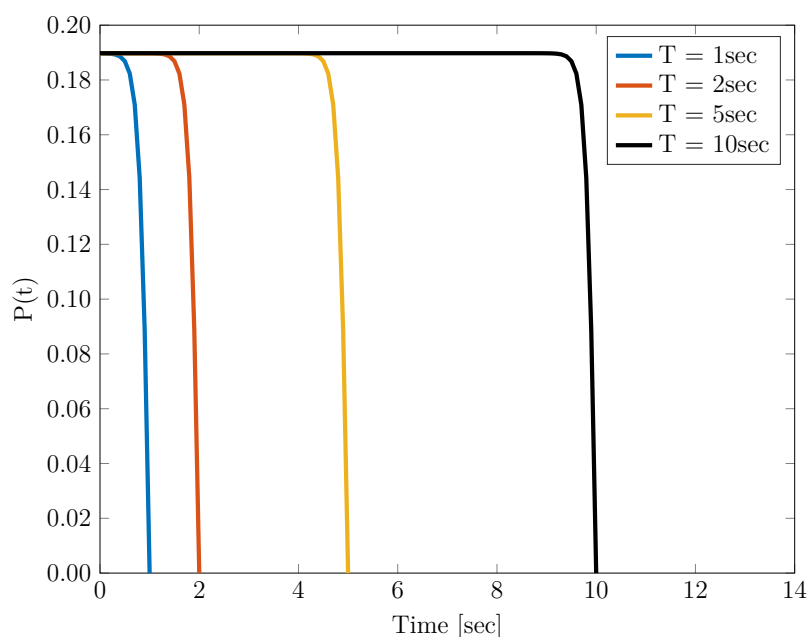

Figure 6.1: Solutions of the Riccati equation for the DC motor example in Exercise 5.2, with $\rho = 1000$, $s = 0$ and different time horizons T

The solution is constant during most of the time interval and changes only when approaching the final time. From a practical point of view, constant feedback gains are much more

suitable for implementation in controllers for steady state operation than time-varying gains; and the fact that the time horizon T is a design parameter that can be chosen by the designer, suggests to take the limit as $T \to \infty$ to obtain constant gain feedback. From (5.7) we have

$$\lim_{T \to \infty} P = U_{21} U_{11}^{-1}, \tag{6.1}$$

i.e. as $T \to \infty$ the solution to the Riccati equation becomes time-invariant, which leads to a constant, steady state optimal state feedback gain

$$F^* = -R^{-1} B^T U_{21} U_{11}^{-1}.$$

With a constant solution P, we have $\dot{P} = 0$, and the Riccati differential equation turns into the *algebraic Riccati equation*

$$0 = PA + A^T P + Q - PBR^{-1}B^T P, \tag{6.2}$$

which is satisfied by P as given by (6.1). From (5.7) it is also clear that a penalty S on the final state has no influence on the solution as $T \to \infty$. Changing the problem description accordingly, we have thus obtained the solution to the *infinite-time regulator problem.* The infinite-time regulator problem is also called the steady state regulator problem, it is: for the system

$$\dot{x}(t) = Ax(t) + Bu(t)$$

and the performance index

$$V(x_0, u, t_0) = \int_0^\infty (x^T Q x + u^T R u) \, dt, \tag{6.3}$$

find the optimal control input $u^*(t)$ that minimizes V. The steady state solution is

$$u^*(t) = -R^{-1} B^T U_{21} U_{11}^{-1} x = F^* x \tag{6.4}$$

where U_{21} and U_{11} are submatrices of the transformation matrix U that decouples the Hamiltonian system into a stable and an unstable subsystem.

Compared with the finite-time problem, the steady state regulator problem is easier to solve, and the solution is easier to implement. But with an infinite time horizon, new questions concerning the asymptotic behaviour of the solution have to be addressed. The following example illustrates possible problems.

Example 6.1

Consider the system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad \begin{bmatrix} x_{10} \\ x_{20} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and the performance index

$$V = \int_0^\infty (x_1^2 + x_2^2 + u^2) \, dt.$$

From $\dot{x}_1 = x_1$, we have $x_1 = e^t$ independent of the control input, so the integrand will contain the term e^{2t} no matter what control is chosen, and the performance index will be infinite.

There are two reasons why there exists no solution in the above example. The first one is the fact that the unstable mode x_1 is uncontrollable. The second reason is that this unstable and uncontrollable mode is reflected in the performance index. Two questions must be addressed: a) does a solution to the problem exist, and b) is the solution stable.

Existence of a Solution

A solution P to the algebraic Riccati equation for the infinite time regulator problem exists, if the system (A, B) is controllable. By "existence of a solution", we mean existence of a bounded solution $P < \infty$.

To see that this is true, we note that controllability implies the existence of a state feedback gain F such that $(A + BF)$ is stable. Then the control input $u(t) = Fx(t)$, which is not necessarily optimal, stabilizes the system, and the resulting performance index $V(x_0, u, t_0)$ will be finite. But the optimal performance index cannot be greater than the performance index resulting from any control input, so we have

$$V^*(x_0) \leq V(x_0, u) < \infty.$$

Recalling that $V^*(x_0) = x_0^T P x_0$, we conclude that $x_0^T P x_0 < \infty$; and because this holds for any x_0, it follows that P is bounded.

Stability

Controllability ensures the existence of a bounded solution P to the algebraic Riccati equation, and of a state feedback gain that minimizes the performance index. But it does not imply closed-loop stability, as the following example illustrates.

Example 6.2

Consider the system

$$\dot{x} = x + u$$

and the performance index

$$V = \int_0^\infty u^2 \, dt.$$

The system is controllable, and the optimal control is $u^*(t) \equiv 0$, resulting in $V = 0$ and an unstable closed-loop system.

The instability here is due to the fact that the unstable mode is not reflected in the performance index. A bounded solution P guarantees stability if all modes are reflected

in the performance index. To formulate that requirement, we write the weight matrix for the states as $Q = C^T C$. Thus, the term $x^T Q x$ can be interpreted as the cost resulting from a fictitious output $z = Cx$. Then stability of the closed-loop system is guaranteed if (A, C) is observable.

To show this, we need a result from Lyapunov stability theory: the system $\dot{x} = \bar{A}x$ is stable if there exists a Lyapunov function $V = x^T P x$ such that $P > 0$ and $\dot{V} \leq 0$, and where $\dot{V} \equiv 0$ implies $x(t) \equiv 0$. Now let $\bar{A}$ denote the optimal closed-loop state matrix

$$\bar{A} = A + BF^* = A - BR^{-1}B^T P.$$

We first show (A, C) observable $\Rightarrow P > 0$ by showing that $P \geq 0$ leads to a contradiction. Assume $P \geq 0$, then there exists a nonzero initial state $x_0 \neq 0$ such that

$$x_0^T P x_0 = \int_0^\infty (x^T C^T C x + u^T R u)\, dt = 0.$$

But this can only be true if $Cx(t) = Ce^{At}x_0 \equiv 0$ for $0 \leq t < \infty$, and (A, C) observable then implies that $x_0 = 0$, which contradicts the assumption $x_0 \neq 0$.

Next we prove that $\dot{V} = \frac{d}{dt}(x^T P x) \leq 0$, and that $\dot{V} \equiv 0$ implies $x(t) \equiv 0$. Observe that from the algebraic Riccati equation and from the definition of $\bar{A}$ we have

$$\bar{A}^T P + P\bar{A} = -PBR^{-1}B^T P - C^T C.$$

Substituting the right hand side in

$$\dot{V} = \dot{x}^T P x + x^T P \dot{x} = x^T(\bar{A}^T P + P\bar{A})x$$

gives

$$\dot{V} = -x^T PBR^{-1}B^T P x - x^T C^T C x,$$

so clearly $\dot{V} \leq 0$, and $\dot{V} \equiv 0$ can only be true if $Cx(t) \equiv 0$, which by observability implies $x_0 = 0$ and thus $x(t) \equiv 0$. Invoking the Lyaponov stability result quoted above, this proves that observability of (A, C) guarantees stability of the optimal closed-loop system.

Controllability and observability are only sufficient, not necessary conditions for the existence of a stabilizing solution. One can extend the above results and show the following: a necessary and sufficient condition for the existence of a stabilizing solution is that the *unstable modes* are controllable (i.e. the system is stabilizable) and that there are no unobservable modes on the imaginary axis. If (A, C) has stable but unobservable modes, the solution P will be positive semidefinite (instead of positive definite).

Closed Loop Eigenvalues

The derivation of the solution (5.7) to the Riccati differential equation, which led to the steady state solution (6.1), was based on decoupling the stable and the unstable eigenvalues of the Hamiltonian matrix. Using the algebraic Riccati equation, one can show that the optimal closed-loop eigenvalues are the stable eigenvalues of the Hamiltonian matrix: apply the similarity transformation

$$T^{-1}HT = \tilde{H}, \qquad T = \begin{bmatrix} I & 0 \\ P & I \end{bmatrix}$$

to the Hamiltonian matrix. The result is

$$\tilde{H} = \begin{bmatrix} \bar{A} & -BR^{-1}B^T \\ 0 & -\bar{A}^T \end{bmatrix},$$

so the eigenvalues of H are the eigenvalues of $\bar{A}$ together with those of $-\bar{A}$.

For single input, single output systems, this result makes it possible to solve the infinite time regulator problem as a pole assignment problem: form the Hamiltonian and compute the optimal eigenvalues, and then find the unique state feedback gain that assigns the optimal eigenvalues. For multivariable systems, the optimal eigenvalues alone do not uniquely determine the optimal state feedback gain, and one must consider also the optimal eigenvectors.

Exercise 6.3 illustrates the procedure for a single input, single output system.

Uniqueness of the Solution

We have shown that controllability and observability guarantee the existence of a stable solution to the steady state regulator problem, and that the solution can be obtained via the solution to the algebraic Riccati equation. However, unlike the Riccati differential equation, the algebraic Riccati equation has in general many solutions, as the following example illustrates.

Example 6.3

Consider the scalar system

$$\dot{x} = 3x + u$$

and the performance index

$$V = \int_0^\infty (7x^2 + u^2)\, dt.$$

The algebraic Riccati equation is

$$0 = 6P - P^2 + 7 = (P - 7)(P + 1),$$

and there are two solutions

$$P_1 = 7, \quad P_2 = -1$$

corresponding to two different state feedback gains $f_1 = -7$ and $f_2 = 1$. However, only f_1 leads to a stable closed-loop system $\dot{x} = -4x$, whereas f_2 leads to the unstable system $\dot{x} = 4x$, so here the optimal solution is obviously P_1, which is positive definite.

One can show that under the assumption of controllability and observability, there is *only one stabilizing* solution, and that the stabilizing solution is also the only one that is positive definite. The solution $P = U_{21}U_{11}^{-1}$ is in fact stabilizing when the first half of the columns of U are the stable eigenvectors of H (or a basis for the stable eigenspace). On the other hand, when H is $2n \times 2n$, any combination of n eigenvectors forming the matrices U_{21} and U_{11} leads to a solution P of the algebraic Riccati equation, but the closed-loop eigenvalues are the eigenvalues corresponding to these eigenvectors, so any choice other than the n stable eigenvectors results in an unstable closed-loop system.

Exercises

Problem 6.1

For the plant of Problem 5.2, find the optimal state feedback $u = f\omega$ that minimizes the cost function

$$\int_0^\infty (\omega^2(t) + \rho u^2(t))dt$$

For this purpose, use the MATLAB command

$$f = -\text{lqr}(a, b, 1, \rho)$$

where ρ can be used as a tuning parameter.

a)   Compute the optimal state feedback gain for different values of ρ and compare the resulting responses of the closed-loop system to an initial speed $\omega(0) = 1$;

b)   Plot $\int_0^1 \omega^2 dt$ versus $\int_0^1 u^2 dt$ for different values of ρ in the range $0.1 \leq \rho \leq 10$.

Problem 6.2
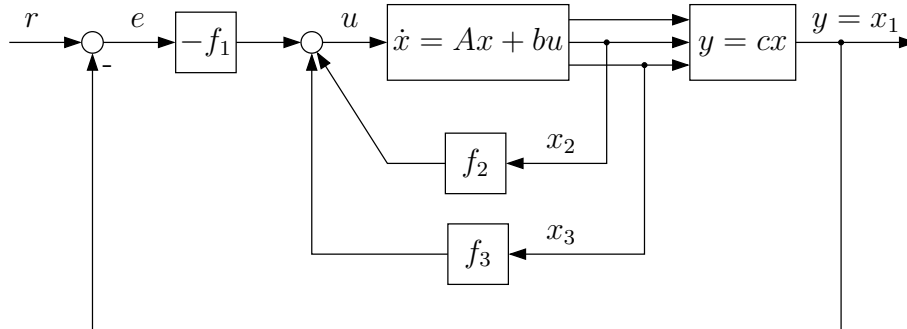
Consider the control system shown in Figure 6.2.

Figure 6.2: System block diagram

A state space model of the plant is

$$\dot{x} = Ax + bu$$
$$y = cx$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -2 & -3 \end{bmatrix}; \qquad b = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \qquad c = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

(a)   The performance index to be minimized is

$$V = \int_0^\infty \left(x^T Q x + u^T R u \right) dt$$

Assume that initially

$$Q = \begin{bmatrix} q_{11} & 0 & 0 \\ 0 & q_{22} & 0 \\ 0 & 0 & q_{33} \end{bmatrix} = I; \qquad R = 1$$

Design an optimal state feedback control law for the above system using MATLAB by tuning Q and R. Assume that the reference input is zero (or $r = 0$) for designing the controller.

(b)   Plot the step response of the closed loop system with the resulting controller, and use the cost weights Q and R to tune the response.

Problem 6.3

Consider the system

$$\dot{x}(t) = Ax(t) + bu(t)$$

where

$$A = \begin{bmatrix} 0 & 1 \\ 5 & 0 \end{bmatrix} \qquad b = \begin{bmatrix} 0 \\ 3 \end{bmatrix}.$$

A state feedback control law $u(t) = fx(t)$ is to be determined that minimizes the performance index

$$V = \int_0^\infty x^T(t)Qx(t) + \rho u^2(t)\, dt$$

with weights

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad \rho = 1.$$

Use the Hamiltonian matrix to compute the optimal closed-loop eigenvalues and the optimal state feedback gain.

Chapter 7

Properties of the Optimal Regulator

In this chapter, we study properties of infinite-time linear optimal regulator systems from an engineering point of view. In the previous chapter it was shown how for a given problem - given in terms of the matrices A, B, Q and R - the optimal state feedback gain can readily be computed numerically. The matrices A and B represent the plant to be controlled, and the matrices Q and R express the design objectives. In practice, however, it is very often not clear what values for Q and R should be chosen to achieve a desired performance; and it turns out that - rather than being a natural part of the problem description - these matrices are in fact design parameters, and that the main use of linear quadratic theory is to provide a convenient parametrization of trial solutions that can be improved by iteratively changing Q and R.

Provided the system is stablizable and detectable (detectable from a fictitious output, in the sense discussed in the previous chapter), the existence of a solution is guaranteed for any choice of $R > 0$ and $Q \geq 0$. Moreover, the optimal closed-loop system is guaranteed to be stable. Another attractive feature of LQ-optimal design is the fact that there is no additional complexity involved when working with multivariable systems compared to single-input single-output systems. We will now explore some further properties of linear optimal regulators.

Kalman's Identity

An important result for studying the properties of linear optimal regulators is the following identity, due to Kalman. Introduce the notation

$$\Phi(j\omega) = (j\omega I - A)^{-1}, \qquad \Phi^H(j\omega) = [(-j\omega I - A)^{-1}]^T,$$

then the following equation - referred to as Kalman's identity - holds true for all ω

$$(I - B^T \Phi^H(j\omega) F^T) R (I - F \Phi(j\omega) B) = R + B^T \Phi^H(j\omega) Q \Phi(j\omega) B. \qquad (7.1)$$

The proof is deferred to the end of this chapter. Because $B^T \Phi^H Q \Phi B \geq 0$, an immediate consequence of Kalman's identity is the following inequality, also referred to as Kalman's inequality

$$(I - B^T \Phi^H (j\omega) F^T) R (I - F \Phi(j\omega) B) \geq R \qquad \forall \omega. \tag{7.2}$$

For single-input, single-output systems, Kalmans's identity simplifies to

$$\rho |1 - f \Phi(j\omega) b|^2 = \rho + b^T \Phi^H (j\omega) Q \Phi(j\omega) b \qquad \forall \omega. \tag{7.3}$$

If in addition the weight matrix Q has rank one and can be written as $Q = c^T c$ (i.e. only one fictitious output $y = cx$ is being penalized), then the inequality (7.2) becomes

$$|1 - f \Phi(j\omega) b| \geq 1 \qquad \forall \omega. \tag{7.4}$$

Optimal State Feedback Loop Gain

We shall study some of the implications of inequality (7.4) for single-input, single-output systems. The importance of this inequality is due to the fact that the term $-f \Phi(j\omega) b$ is the loop gain of the optimal closed-loop system. To see this, draw the state feedback loop as shown in Fig. 7.1, where the negative feedback convention used in the classical control literature is adopted.
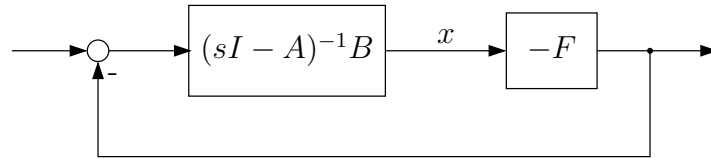

Figure 7.1: Loop gain of the optimal closed-loop system

Because $L(j\omega) = -f \Phi(j\omega) b$ is the loop gain, the left hand side of (7.4) is equal to the distance $|L(j\omega) + 1|$ between a point at frequency ω on the Nyquist plot of the loop gain $L(j\omega)$ and the critical point -1 in the complex plane. Thus, Kalman's inequality states that the Nyquist plot of the loop gain under optimal state feedback never enters a disc of radius 1 around the critical point.

Phase and Gain Margin

Provided the plant is controllable and observable, the optimal closed-loop system is guaranteed stable. Now assume the nominal loop gain $L(j\omega)$ is replaced by $kL(j\omega)$, with k a static gain. Then from the fact that the Nyquist plot of the loop gain never enters a disc of radius 1 around the point -1, it is straightforward to show that the number of encirclements of the critical point cannot change as long as $0.5 < k < \infty$. Thus optimal state feedback guarantees an infinite upper gain margin and a 50% lower gain margin.

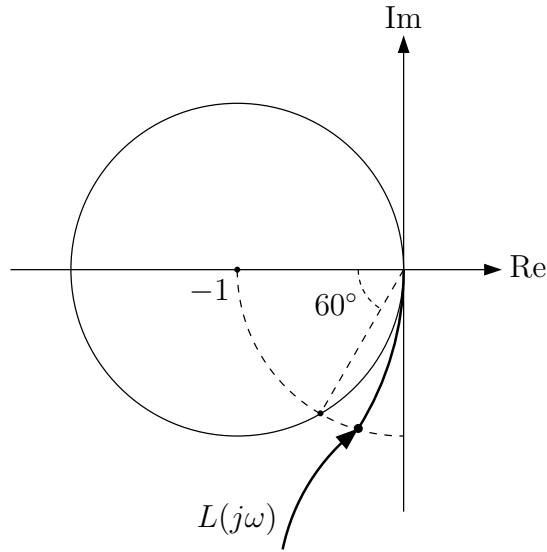To investigate the phase margin, consider the Nyquist plot shown in Fig. 7.2.

Figure 7.2: Phase margin under optimal state feedback

The phase margin is the amount of negative phase shift that must be introduced into the loop to make the Nyquist plot touch the critical point. The point of the Nyquist plot $L(j\omega)$ that will hit the point -1 is the point at the intersection with a circle of radius 1 around the origin. From the Figure, it is clear that the smallest phase shift that could make any Nyquist plot touch the critical point is 60°. Thus, optimal state feedback guarantees a phase margin of at least 60°.

Unfortunately, the above phase and gain margins are guaranteed only when the full state vector is available and used for feedback. In practice, this is normally not the case, and when the state variables are estimated and state feedback is replaced by state estimate feedback, these margins can disappear. Techniques to partially recover these properties (*Loop Transfer Recovery*) will be discussed in chapter 10.

Sensitivity and Complementary Sensitivity

The sensitivity function $S(j\omega)$ is defined as

$$S(j\omega) = \frac{1}{1 + L(j\omega)}.$$

Therefore, inequality (7.4) implies that $|S(j\omega)| \leq 1$ for all frequencies, which is an attractive property as far as disturbance rejection and tracking is concerned.

The complementary sensitivity $T(j\omega)$ is

$$T(j\omega) = 1 - S(j\omega) = \frac{L(j\omega)}{1 + L(j\omega)}.$$

Observing that this is at the same time the closed-loop transfer function, we find from Fig. 7.1

$$T(j\omega) = -f(j\omega I - A - bf)^{-1}b.$$

For rejection of measurement noise, as well as for robustness against unstructured perturbations, the roll-off of $T(j\omega)$ at high frequencies is of interest. Taking the limit as $\omega \to \infty$, we have

$$\lim_{\omega\to\infty} j\omega T(j\omega) = -fb = \frac{1}{\rho} b^T P b,$$

which means that at high frequencies we have

$$|T(j\omega)| \sim \frac{1}{\omega}$$

or a roll-off of 20dB per decade. Therefore, noise performance will be rather poor. However, whereas replacement of state feedback by state estimate feedback can destroy the gain and phase margins, using a dynamic state observer will increase the roll-off and thus improve the noise performance of the closed-loop system.

Spectral Factorisation

It was shown earlier that the optimal closed-loop eigenvalues are the stable eigenvalues of the Hamiltonian matrix. Kalman's identity provides a similar characterisation of the optimal eigenvalues in frequency domain, using spectral factorisation. For single-input, single-output systems, this characterisation becomes particularly simple.

We first state the result. Consider the system

$$\dot{x} = Ax + bu$$
$$y = cx.$$

Here y is again a fictitious output, corresponding to the choice $Q = c^T c$. The performance index is

$$V = \int_0^\infty (y^2 + \rho u^2)\, dt.$$

Introduce the transfer function

$$\frac{b(s)}{a(s)} = c(sI - A)^{-1}b, \tag{7.5}$$

then the optimal closed-loop eigenvalues are the stable roots of the polynomial

$$p(s) = a(-s)a(s) + \frac{1}{\rho} b(s)b(-s). \tag{7.6}$$

To prove this, we use the fact that for two matrices $X \in \mathbb{R}^{m\times n}$ and $Y \in \mathbb{R}^{n\times m}$, we have

$$\det(I_m - XY) = \det(I_n - YX)$$

(this can be shown by considering the determinant of the matrix

$$\begin{bmatrix} I_m & X \\ Y & I_n \end{bmatrix}$$

and making use of the Schur-Complement.) Notice that for column vectors x and y this implies

$$\det\left(I_m - xy^T\right) = 1 - y^T x.$$

Now let $a_c(s)$ denote the closed-loop characteristic polynomial, which can be factored as

$$a_c(s) = \det\left(sI - A - bf\right) = \det\left(sI - A\right)\det\left(I - (sI - A)^{-1}bf\right). \tag{7.7}$$

Using the above determinant identity and the notation $\Phi(s) = (sI - A)^{-1}$, the last factor in (7.7) is

$$\det\left(I - \Phi(s)bf\right) = 1 - f\Phi(s)b.$$

Thus, equation (7.7) can be written as

$$a_c(s) = a(s)(1 - f\Phi(s)b)$$

and we have

$$1 - f\Phi(s)b = \frac{a_c(s)}{a(s)}.$$

Substituting this and (7.5) in the Kalman identity

$$\rho(1 - b^T\Phi^H f^T)(1 - f\Phi b) = \rho + b^T\Phi^H c^T c\Phi b$$

yields

$$\rho\,\frac{a_c(-s)a_c(s)}{a(-s)a(s)} = \rho + \frac{b(-s)b(s)}{a(-s)a(s)}.$$

Rearrangement leads to the result (7.6).

This result provides for single-input, single-output systems a simple way of computing the optimal state feedback gain: from the given data, form the polynomial $p(s)$ in (7.6), find the stable roots, and solve a pole-assignment problem.

Eigenvalues on the Imaginary Axis

At this point, we come back to a question that arose when the eigenvalues of the Hamiltonian matrix were discussed: under what conditions can we be sure that the polynomial $p(s)$ has no roots on the imaginary axis?

Assume $p(s)$ has a purely imaginary eigenvalue, i.e. $p(j\omega_o) = 0$ for some ω_o. Then (7.6) implies

$$p(j\omega_o) = |a(j\omega_o)|^2 + \frac{1}{\rho}|b(j\omega_o)|^2 = 0,$$

which can only be true if

$$a(j\omega_o) = b(j\omega_o) = 0.$$

But then both $a(s)$ and $b(s)$ can be factored as

$$a(s) = \tilde{a}(s)(s - j\omega_o)$$
$$b(s) = \tilde{b}(s)(s - j\omega_o).$$

Thus

$$\frac{a(s)}{b(s)} = \frac{\tilde{a}(s)(s - j\omega_o)}{\tilde{b}(s)(s - j\omega_o)},$$

the transfer function (7.5) must have a pole-zero cancellation. A sufficient condition to exclude this is that the state space representation (A, b, c) is stabilizable and detectable. Thus, stabilizability and detectability guarantee that $p(s)$ has no roots on the imaginary axis (similarly that the Hamiltonian matrix has no eigenvalues on the imaginary axis).

The spectral factorisation result (7.6) provides a characterisation of the optimal closed-loop poles in terms of the open-loop poles and zeros. We use this result now to investigate two limiting cases: the case when the cost of control is very high and the case when it is very low.

High Cost of Control

First we study the optimal solution when the cost of control is infinite, i.e. $\rho \to \infty$. From

$$p(s) = a_c(-s)a_c(s) = a(-s)a(s) + \frac{1}{\rho} b(s)b(-s).$$

it is clear that in this case we have

$$a_c(-s)a_c(s) \quad \to \quad a(-s)a(s).$$

Because all roots of $a_c(s)$ must be in the left half plane, this means that

a)   Stable open-loop roots remain where they are, and

b)   Unstable open-loop roots are reflected about the imaginary axis.

The first point makes intuitively sense: when control is expensive, one should not spend control effort to move stable poles around. For the second point, it is clear that unstable roots must be moved into the left half plane, but it is not immediately clear how far. The spectral factorisation result gives the answer: the optimal strategy is to move an unstable pole to its mirror image in the left half plane. A simple example serves to illustrate this result.

Example 7.1

Consider the scalar system

$$\dot{x} = ax + u, \qquad x(0) = x_0$$
$$y = x$$

and the performance index

$$V = \int_0^\infty (y^2 + \rho u^2)\, dt.$$

The objective is to find the state feedback gain f such that $u = fx$ minimizes V. The closed-loop system is $\dot{x} = (a + f)x$, and the closed-loop state trajectory is therefore

$$x(t) = x_0 e^{(a+f)t}.$$

Substituting this and the control law into the cost function gives

$$V \;=\; (1 + \rho f^2)x_0^2 \int_0^\infty e^{2(a+f)t} dt \;=\; \frac{1 + \rho f^2}{2(a + f)} \, x_0^2 \, e^{2(a+f)t}\Big|_0^\infty.$$

Obviously, if $a + f$ is nonnegative (if the closed-loop system is unstable), the cost is infinite. So we have

$$V = \begin{cases} -\frac{1 + \rho f^2}{2(a+f)}\, x_0^2, & a + f < 0 \\ \infty & else \end{cases}$$

Taking the limit as $\rho \to \infty$, this shows that if the open-loop system is stable ($a < 0$), the choice $f = 0$ leads to a finite cost $V = -1/(2a)$; this is in fact the optimal solution for a stable plant. If the plant is unstable ($a > 0$), we need $f < a$ to stabilize the closed-loop system; this excludes the choice $f = 0$. To find out how far the closed-loop pole must be shifted into the left half plane in order to minimize V, we take the derivative

$$\frac{dV}{df} = \frac{d}{df}\left(\frac{1 + \rho f^2}{2(a + f)}\right) = 0.$$

This leads to

$$f^2 + 2af - \frac{1}{\rho} = 0$$

or

$$f_{1,2} = -a \pm \sqrt{a^2 + \frac{1}{\rho}}.$$

From the two solutions, $f_1 = 0$ leaves the closed-loop system unstable. When $\rho \to \infty$, the stabilizing and thus optimal solution is $f_2 = -2a$. The optimal closed-loop system is therefore

$$\dot{x} = (a - 2a)x = -ax,$$

and the optimal pole location is the mirror image of the unstable pole.

Low Cost of Control

Now we consider the optimal solution in the limiting case $\rho \to 0$. When control is very cheap, it is intuitively clear that the closed-loop poles should be moved far into the left half plane in order to bring the states quickly to zero. In this case we have

$$a_c(-s)a_c(s) \quad \to \quad \frac{1}{\rho} b(-s)b(s).$$

Let $n = \deg a(s)$ and $m = \deg b(s)$, then a condition for a system to be physically realizable is $n > m$. Among the $2n$ roots of $a_c(-s)a_c(s)$ there must be the $2m$ roots of $b(-s)b(s)$. These are the *finite roots*, i.e. roots at finite values of s. The remaining $2(n-m)$ roots are *roots at infinity*. To investigate the behaviour of these infinite roots, we consider equation (7.6) at large values of s. In this case, the polynomials are dominated by the highest power of s and we can ignore the lower order terms. With $a(s) = s^n + a_{n-1}s^{n-1} + \ldots a_0$ and $b(s) = b_m s^m + b_{m-1}s^{m-1} + \ldots b_0$, we can thus use the approximation

$$a(-s)a(s) = (-1)^n s^{2n}, \qquad b(-s)b(s) = (-1)^m b_m^2 s^{2m}.$$

For large values of s, (7.6) can thus be written as

$$(-1)^n s^{2n} + \frac{1}{\rho}(-1)^m b_m^2 s^{2m} = 0$$

or

$$s^{2(n-m)} = (-1)^{m-1-n}\frac{1}{\rho}b_m^2.$$

The magnitude of the solution is

$$|s| = \left(\frac{b_m^2}{\rho}\right)^{\frac{1}{2(n-m)}},$$

and the right hand side is the radius of a circle on which the roots are located. Fig. 7.3 shows some examples; this pole configuration is known in network theory and filter design as *Butterworth configuration.*

Proof of Kalman's Identity

We conclude this chapter with the proof of Kalman's identity

$$(I - B^T \Phi^H(j\omega)F^T)R(I - F\Phi(j\omega)B) = R + B^T\Phi^H(j\omega)Q\Phi(j\omega)B.$$

which was claimed to hold for all ω if $F = -R^{-1}B^T P$ and P is the positive (semi-) definite solution of the algebraic Riccati equation

$$-PA - A^T P + PBR^{-1}B^T P = Q.$$

associated with this problem. Using $F^T R = -PB$ and adding and subtracting $Pj\omega$, we have

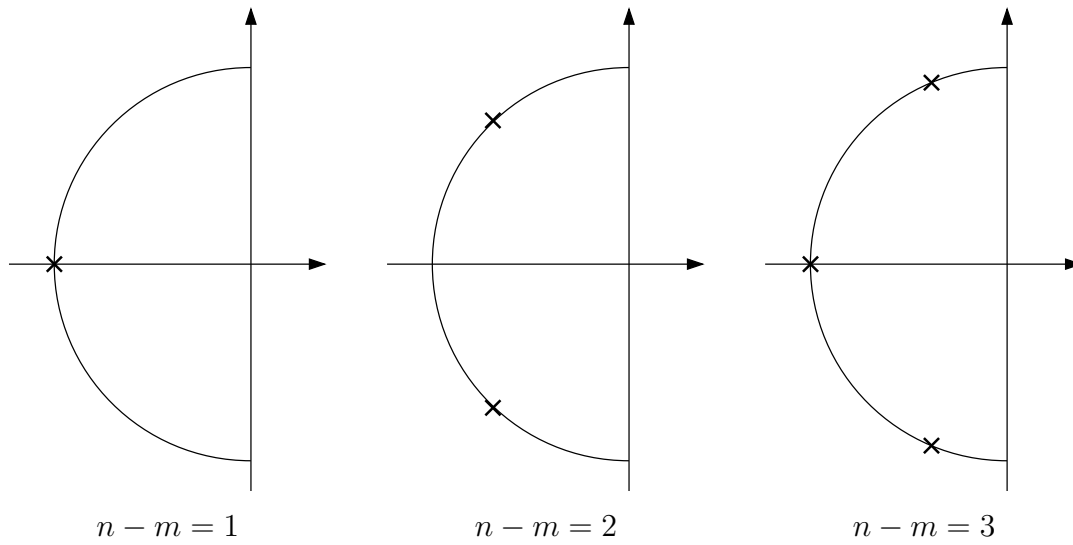$$-PA - A^T P + F^T RF + Pj\omega - Pj\omega = Q$$

Figure 7.3: Butterworth configurations

or

$$P(j\omega I - A) + (-j\omega I - A)^T P + F^T R F = Q.$$

Substituting $\Phi = (j\omega I - A)^{-1}$, and multiplying $B^T \Phi^H$ from left and ΦB from right yields

$$B^T \Phi^H P B + B^T P \Phi B + B^T \Phi^H F^T R F \Phi B = B^T \Phi^H Q \Phi B.$$

Using again $F^T R = -PB$ and adding R on both sides, and introducing $\Psi = F\Phi B$ yields

$$R + B^T \Phi^H Q \Phi B = R - \Psi^H R - R\Psi + \Psi^H R \Psi$$
$$= (I - \Psi^H) R (I - \Psi),$$

which completes the proof. ∎

Exercises

Problem 7.1

Consider the spring-mass system shown in Figure 7.4.

a)   Derive a state space model for the above system (*Hint: Use Newton's laws*) with $x = \begin{bmatrix} q & \dot{q} \end{bmatrix}^T$.

b*)   A performance index V is given by

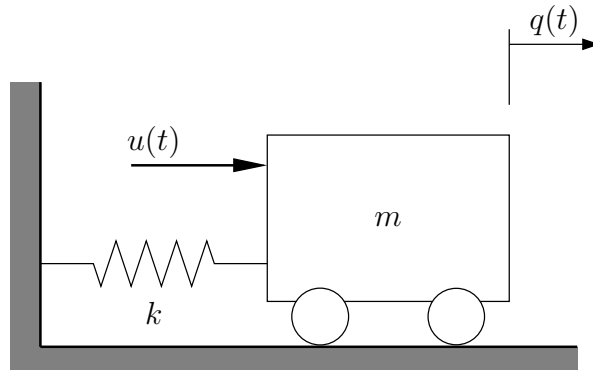$$V = \int_0^\infty [z^2(t) + \rho u^2(t)] dt,$$

Figure 7.4: Mass spring system

where $u(t)$ is the control signal, ρ is the control weighting parameter and $\rho > 0$. The variable of interest $z(t)$ is defined as $z(t) = cx(t)$, $c = \begin{bmatrix} 1 & 0 \end{bmatrix}$.

By solving the Riccati equation show that the optimal state feedback gain vector f as a function of the tuning parameter ρ is

$$f_1 = -\sqrt{1 + \frac{1}{\rho}} + 1; \qquad f_2 = -\sqrt{2\left(\sqrt{1 + \frac{1}{\rho}} - 1\right)}$$

and the closed-loop eigenvalues are

$$\lambda_{1,2} = -\frac{1}{2}\sqrt{2\left(\sqrt{1 + \frac{1}{\rho}} - 1\right)} \pm j\frac{1}{2}\sqrt{2\left(\sqrt{1 + \frac{1}{\rho}} + 1\right)}$$

c) Discuss the effect of ρ on the feedback gains and the closed-loop eigenvalues?
Hint: Check for $\rho \to 0$ and $\rho \to \infty$.

d) Simulate this system using MATLAB/SIMULINK and plot the locus of the closed loop pole locations as ρ varies over $0 < \rho < \infty$.

Problem 7.2

Consider the system

$$\dot{x} = Ax + bu$$
$$y = cx$$

where

$$A = \begin{bmatrix} 0 & 1 \\ 5 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \quad c = [1 \ 0].$$

The performance index is

$$V = \int_0^\infty (y^2 + u^2)\, dt.$$

Compute the transfer function from u to y and use spectral factorisation to find the optimal closed-loop eigenvalues and the optimal state feedback gain.

Problem 7.3

Consider a single input-system $\dot{x} = Ax + bu$, where

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \qquad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

For what values of q is the state feedback gain vector $f = \begin{bmatrix} -1 & -q \end{bmatrix}$ optimal in the sense that there exist $Q \geq 0$, $\rho > 0$ such that f minimizes

$$V = \int_0^\infty (x^T Q x + \rho u^2) dt$$

Problem 7.4

Consider a nearly vertical rod with an attached pendulum, as shown in Figure 7.5. The input u to the system is a torque. The equations of motion are
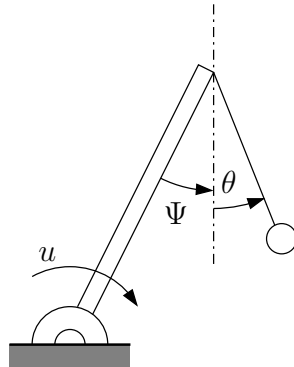

Figure 7.5: Vertical rod with a pendulum

$$\ddot{\psi} - \frac{3}{2}\psi - (\psi + \theta) = u$$
$$\ddot{\theta} + \theta + \ddot{\psi} = 0$$

a)   Derive a state-space model by defining the state variable as

$$x_1 = \psi, \qquad x_2 = \dot{\psi}, \qquad x_3 = \theta, \qquad x_4 = \dot{\theta}$$

b)   Use the spectral factorization theorem and the root locus technique to plot the optimal root loci for regulation of the quantity $y = \psi + \theta$, and select an optimal

pole configuration such that the dominant time constant is $\frac{1}{3}$.

Hint: Use the command `rltool` *in MATLAB.*

c)   Assume that at time $t = 0$, the initial condition is $x(0) = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix}^T$. Design a state-feedback controller using MATLAB so that the mass m will swing out with a time constant of about $\frac{1}{3}$. Also plot the time response of the system for the given initial conditions.

Chapter 8

The Euler-Lagrange Equations

The solution to the linear optimal regulator problem was derived in chapter 1 using the principle of optimality and the Hamilton-Jacobi equation. This approach appears to be the simplest self-contained way to derive the optimal control law, i.e. without having to quote results from elsewhere. An alternative approach to the derivation of the optimal control law is to use the Minimum Principle of Pontryagin together with the Euler-Lagrange equations. In this chapter, a brief outline of this approach is presented.

As in the derivation of the Hamilton-Jacobi equation, we consider a more general class of problems with a nonlinear, time-varying plant and cost function. For simplicity we restrict the discussion to single-input, single-output systems; the model is

$$\dot{x} = f(x, u, t), \qquad x(0) = x_0$$

and the cost function

$$V = \int_0^T l(x, u, t)\, dt + m(x(T)).$$

The objective is to find the control input $u^*(t)$, $0 \le t \le T$ that minimizes V, subject to the constraint $\dot{x} = f(x, u, t)$.

The first step is to adjoin the constraint $\dot{x} = f$ to the cost by rewriting the cost function as

$$V = \int_0^T (l + \psi^T (f - \dot{x}))\, dt + m.$$

Since $f - \dot{x} = 0$, this does not change the cost function. Because f and $\dot{x}$ are column vectors, they need to be multiplied by a row vector ψ^T. This vector is called a *Lagrange multiplier*, it can be freely chosen, for example as a time-varying vector $\psi = \psi(t)$. Next we introduce the *Hamilton function*

$$H = l + \psi^T f.$$

With this definition, the cost function becomes

$$V = \int_0^T (H - \psi^T \dot{x})\, dt + m.$$

The term containing $\dot{x}$ can be eliminated using integration by parts

$$\int_0^T \psi^T \dot{x}\, dt = \psi^T x \big|_0^T - \int_0^T \dot{\psi}^T x\, dt,$$

and substituting this in the cost function yields

$$V = -\ \psi^T x \big|_0^T + \int_0^T (H + \dot{\psi}^T x)\, dt + m(x_T). \tag{8.1}$$

To find the control input that minimizes (8.1), we use the following result from calculus of variations. If u^* is the optimal control input, and x^* the optimal state trajectory, then an infinitely small but arbitrary variation δu will lead to a small variation δx in the state trajectory. The *optimal* solution has the property that the variation of the optimal cost is zero for any δu. Therefore, we apply the condition

$$\delta V = 0 \qquad \forall \delta u$$

to (8.1) in order to find the optimal control input.

From (8.1) we have

$$\delta V = -\ \psi^T \delta x \big|_T + \psi^T \delta x \big|_0 + \int_0^T \left[\left(\frac{\partial H}{\partial x}\right)^T \delta x + \frac{\partial H}{\partial u} \delta u + \dot{\psi}^T \delta x \right] dt + \left(\frac{\partial m}{\partial x}\right)^T \delta x \bigg|_T.$$

Because the initial state is given, the second term on the right hand side is zero. Collecting terms with common factors yields

$$\delta V = \left[\left(\frac{\partial m}{\partial x}\right)^T - \psi^T \right] \delta x \bigg|_T + \int_0^T \left[\left(\left(\frac{\partial H}{\partial x}\right)^T + \dot{\psi}^T \right) \delta x + \frac{\partial H}{\partial u} \delta u \right] dt \tag{8.2}$$

This equation offers a convenient choice for the Lagrange multiplier $\psi(t)$: if we choose

$$\psi(T) = \frac{\partial m}{\partial x}\bigg|_T, \qquad \dot{\psi} = -\frac{\partial H}{\partial x}, \tag{8.3}$$

then (8.2) simplifies to

$$\delta V = \int_0^T \frac{\partial H}{\partial u} \delta u\, dt.$$

Optimality requires $\delta V = 0 \ \forall \delta u(t)$, thus a necessary condition for u to be optimal is

$$\frac{\partial H}{\partial u} = 0. \tag{8.4}$$

Equations (8.3) and (8.4) are known as the *Euler-Lagrange equations.*

Returning to the linear optimal regulator problem, we replace the functions $f(x, u, t)$ and $l(x, u, t)$ in the general problem description by

$$f = Ax + bu, \qquad l = \frac{1}{2}(x^T Q x + \rho u^2).$$

For simplicity we choose $m(x_T) = 0$. The Hamilton function is then

$$H = l + \psi^T f = \frac{1}{2}(x^T Q x + \rho u^2) + \psi^T (Ax + bu).$$

Equations (8.3) are now

$$\dot{\psi} = -Qx - A^T \psi, \qquad \psi(T) = 0. \tag{8.5}$$

For the third equation, we need the derivative of the Hamilton function H

$$\frac{\partial H}{\partial u} = \rho u + b^T \psi = 0.$$

Solving for u yields the optimal control input as a linear function of ψ

$$u^*(t) = -\frac{1}{\rho} b^T \psi(t).$$

Substituting this in the plant equation gives

$$\dot{x} = Ax - b \frac{1}{\rho} b^T \psi,$$

which can be combined with (8.5) into the *Hamiltonian system*

$$\begin{bmatrix} \dot{x} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} A & -b \frac{1}{\rho} b^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} x \\ \psi \end{bmatrix}$$

The optimal control law can be obtained from the solution to the Hamiltonian system with boundary conditions $x(0) = x_0$, $\psi(T) = 0$, but unlike the Hamilton-Jacobi equation, the Euler-Lagrange approach does not show directly that the optimal control law has the form of linear state feedback. This must be shown separately, in fact it can be shown that as $T \to \infty$, and under the usual controllability and observability assumptions there exists a constant (but so far unknown) matrix P that satisfies

$$\psi(t) = Px(t).$$

Substituting Px for ψ in the control law leads to linear state feedback

$$u^*(t) = -\frac{1}{\rho} b^T P x(t).$$

To find the matrix P, use this control in the plant equation and multiply from the left by P to get

$$P\dot{x} = PAx - Pb \frac{1}{\rho} b^T Px.$$

Finally, equating this with

$$P\dot{x} = \dot{\psi} = -Qx - A^T Px$$

shows that P is the solution to the algebraic Riccati equation.

Chapter 9

State Estimate Feedback

We now remove the assumption on which the optimal control law discussed in the previous chapters is based, namely that the full state is available for feedback. In practical problems, this is rarely the case; instead of the state variables, only estimates can be used for feedback. In this chapter, we will study the estimation problem and the solution to the optimal regulator problem when state feedback is replaced by state estimate feedback. In the next chapter we will investigate some properties of the closed-loop system when optimal state feedback is replaced by state estimate feedback.

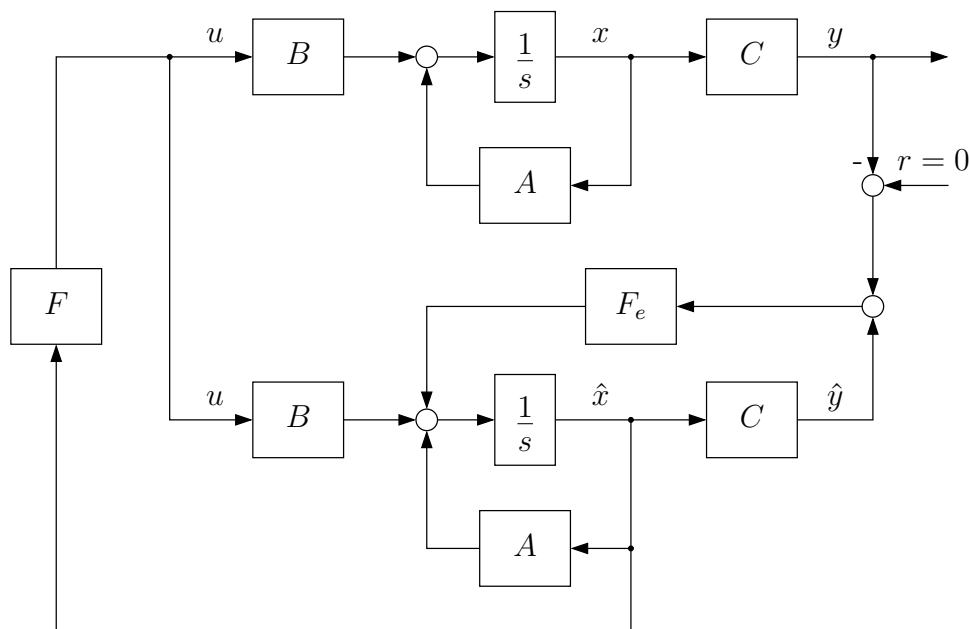We assume that the estimator has the structure of an identity observer as shown in Fig. 9.1.

Figure 9.1: State estimate feedback

Subtracting the estimator equation

$$\dot{\hat{x}} = A\hat{x} + Bu - F_eC(x - \hat{x})$$

from the plant equation

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

gives

$$\dot{\epsilon} = (A + F_eC)\epsilon.$$

The state feedback gain F and the estimator gain F_e can be designed to place the closed-loop plant eigenvalues (the eigenvalues of $A + BF$) and the estimator eigenvalues (the eigenvalues of $A + F_eC$) independently in desired locations. The estimator eigenvalues should be faster than the closed-loop plant eigenvalues, but an upper limit on the estimator bandwidth is determined by the level of measurement noise present in the output channels: increasing the estimator bandwidth leads to increased amplification of high frequency noise.

Whereas in the previous chapters the regulator problem was studied in a purely deterministic framework, the study of the estimation problem makes it necessary to take noise effects into account, and a stochastic problem formulation is more appropriate.

Stochastic Plant Model

We now extend the plant model to include the effect of stochastic disturbances. Consider the system

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t)$$
$$y(t) = Cx(t) + v(t), \tag{9.1}$$

where $w(t)$ is *process noise* and $v(t)$ is *measurement noise*. Both noise processes are assumed to be white, zero mean, Gaussian and uncorrelated, and satisfy

$$E[w(t)w^T(t + \tau)] = Q_e\delta(\tau), \qquad E[v(t)v^T(t + \tau)] = R_e\delta(\tau).$$

A block diagram of the closed-loop with the stochastic plant model is shown in Fig. 9.2.

Optimal Estimation Problem

Let $\varepsilon(t) = x(t) - \hat{x}(t)$ denote the estimation error, and let q be a weighting vector such that $q^T\varepsilon$ is a linear combination of the errors where the elements of q reflect the relative weight placed on the estimation of individual state variables. With the above definitions, we pose the estimation problem as follows. Given the estimator structure in Fig. 9.1, find the estimator gain F_e that minimizes the stochastic cost function

$$V_e = \lim_{t \to \infty} E[\varepsilon^T q q^T \varepsilon]. \tag{9.2}$$
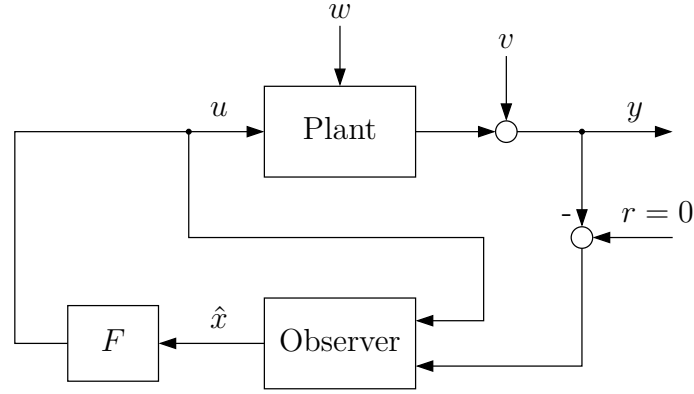
Figure 9.2: Closed-loop system with stochastic plant model

Here the limit is taken as $t \to \infty$ because we are interested in the steady state. Subtracting the estimator equation

$$\dot{\hat{x}} = A\hat{x} + Bu + F_e(\hat{y} - y)$$

from the state equation in (9.1) yields

$$\dot{\varepsilon} = (A + F_e C)\varepsilon + \xi,$$

where ξ is the white, zero mean, Gaussian noise process

$$\xi = w + F_e v, \qquad E[\xi(t)\xi^T(t + \tau)] = (Q_e + F_e R_e F_e^T)\delta(\tau).$$

Stochastic-Deterministic Dualism

The estimation problem as formulated above is a stochastic optimization problem. We will not solve it directly, but we will show that this problem has the same structure as a deterministic problem the solution of which we know: the linear regulator problem. When a stochastic problem has the same structure and can be solved by the same methods as an equivalent deterministic problem, we speak of a stochastic-deterministic dualism.

To establish this dualism, we first consider a simple deterministic and a simple stochastic problem.

Deterministic Problem

Given the autonomous plant model

$$\dot{x} = Ax, \qquad x(0) = x_0,$$

assume we are interested in the value of the cost function

$$V = \int_0^\infty x^T Q x \, dt, \qquad Q = Q^T > 0.$$

One can show that the value of the cost function is

$$V = x_0^T P x_0 \tag{9.3}$$

where P is the positive definite matrix satisfying

$$PA + A^T P + Q = 0. \tag{9.4}$$

This can be shown in the same way as the following result, see Exercise 9.1.

Stochastic Problem

Now consider the process

$$\dot{x} = Ax + \xi, \tag{9.5}$$

where ξ is a white, zero mean, Gaussian noise process that satisfies $E[\xi(t)\xi^T(t + \tau)] = Q_s\delta(\tau)$, and assume we want to find the value of the stochastic cost function

$$V_s = \lim_{t\to\infty} E[x^T q q^T x],$$

where q is a weighting vector as before.

We will show that the solution to this problem is

$$V_s = q^T P_s q, \tag{9.6}$$

where P_s is a positive definite matrix satisfying

$$P_s A^T + A P_s + Q_s = 0. \tag{9.7}$$

Notice that this problem and its solution have the same structure as the deterministic problem; A is replaced by A^T, x_0 by q, Q by Q_s and P by P_s.

To prove that (9.6) is indeed the value of the cost, we start with the solution to the state equation (9.5)

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}\xi(\tau) \, d\tau.$$

Using the properties of ξ, we have

$$
\begin{aligned}
E[xx^T] &= E\left[e^{At}x_0 x_0^T e^{A^T t} + \int_0^t \int_0^t e^{A(t-\tau_1)}\xi(\tau_1)\xi^T(\tau_2)e^{A^T(t-\tau_2)} d\tau_1 d\tau_2\right] \\
&= e^{At}x_0 x_0^T e^{A^T t} + \int_0^t \int_0^t e^{A(t-\tau_1)}Q_s\delta(\tau_1 - \tau_2)e^{A^T(t-\tau_2)} d\tau_1 d\tau_2 \\
&= e^{At}x_0 x_0^T e^{A^T t} + \int_0^t e^{A(t-\tau)}Q_s e^{A^T(t-\tau)} d\tau \\
&= e^{At}x_0 x_0^T e^{A^T t} + \int_0^t e^{A\tau}Q_s e^{A^T\tau} d\tau.
\end{aligned}
$$

Assuming that A is stable and taking the limit as $t \to \infty$

$$\lim_{t \to \infty} E[xx^T] = 0 + \int_0^\infty e^{A\tau} Q_s e^{A^T \tau} d\tau$$

where in the last term the variable of integration has been changed. Multiplying the above equation from left and right by q^T and q respectively yields

$$\lim_{t \to \infty} E[q^T xx^T q] = \lim_{t \to \infty} E[x^T qq^T x] = q^T \int_0^\infty e^{A\tau} Q_s e^{A^T \tau} d\tau q,$$

because $q^T xx^T q$ is scalar. The left hand side is the stochastic cost function V_s, and the above equation can be written as

$$V_s = q^T P_s q$$

when the positive definite matrix P_s is defined as

$$P_s = \int_0^\infty e^{A\tau} Q_s e^{A^T \tau} d\tau.$$

Thus we have shown that the value of the stochastic cost is given by (9.6). It remains to prove that P_s is the solution to (9.7). Substituting the above expression for P_s in (9.7) yields

$$\int_0^\infty \left(e^{A\tau} Q_s e^{A^T \tau} A^T + A e^{A\tau} Q_s e^{A^T \tau} \right) d\tau + Q_s = 0.$$

But this is the same as

$$\int_0^\infty \frac{d}{d\tau} \left(e^{A\tau} Q_s e^{A^T \tau} \right) d\tau + Q_s = 0$$

or

$$e^{A\tau} Q_s e^{A^T \tau} \Big|_0^\infty + Q_s = 0 - Q_s + Q_s = 0,$$

which proves that P_s satisfies (9.7). (That P_s is the *unique* solution follows from the fact that a matrix equation $MP + PN + Q = 0$ has a unique solution P if and only if M and $-N$ have no common eigenvalues.)

Solution to the Optimal Estimation Problem

We return to the optimal estimation problem

$$\min_{F_e} \lim_{t \to \infty} E[\varepsilon^T qq^T \varepsilon]$$

where the estimation error is governed by

$$\dot{\varepsilon} = (A + F_e C)\varepsilon + \xi, \qquad E[\xi(t)\xi^T(t+\tau)] = (Q_e + F_e R_e F_e^T)\delta(\tau).$$

Applying the above result with the replacements

$$A \to A + F_e C, \qquad Q_s \to Q_e + F_e R_e F_e^T,$$

the value of the stochastic cost function V_e is

$$V_e = q^T P_e q,$$

where P_e is the positive definite solution to

$$P_e(A + F_e C)^T + (A + F_e C)P_e + Q_e + F_e R_e F_e^T = 0.$$

We will find the optimal estimator gain F_e by comparing the above with the solution to the linear regulator problem.

Linear Regulator Problem

Because we know that the optimal solution to the linear regulator problem has the form of linear state feedback $u = Fx$, we can pose the linear regulator problem in the following form: given the closed-loop system

$$\dot{x} = (A + BF)x, \qquad x(0) = x_0,$$

find the state feedback gain F that minimizes

$$V = \int_0^\infty (x^T Q x + u^T R u)\, dt = \int_0^\infty x^T(Q + F^T R F)x\, dt.$$

The optimal solution was shown to be

$$F = -R^{-1}B^T P,$$

where P is the positive semidefinite solution to the algebraic Riccati equation

$$PA + A^T P - PBR^{-1}B^T P + Q = 0,$$

and the value of the optimal cost function is

$$V = x_0^T P x_0.$$

With the substitution $PB = -F^T R$, it is straightforward to show that the Riccati equation can also be written as

$$P(A + BF) + (A + BF)^T P + Q + F^T R F = 0.$$

Comparing this with the optimal estimation problem, we find that both problems are equivalent with the replacements

$$Q \to Q_e, \qquad R \to R_e, \qquad A \to A^T, \qquad B \to C^T,$$
$$x_0 \to q, \qquad F \to F_e^T, \qquad P \to P_e.$$

Thus, we can establish the following result by duality, i.e. by making the above replacements in the solution to the regulator problem: the optimal estimator gain is

$$F_e = -P_e C^T R_e^{-1}, \tag{9.8}$$

where P_e is the positive definite solution to

$$P_e A^T + A P_e - P_e C^T R_e^{-1} C P_e + Q_e = 0. \tag{9.9}$$

This equation is known as the filter algebraic Riccati equation (FARE).

The following example illustrates the result.

Example 9.1

Consider the first order system

$$\dot{x} = ax + w$$
$$y = x + v,$$

where w and v are white, zero mean, Gaussian noise processes that satisfy

$$E[w(t)w(t+\tau)] = q_e \delta(\tau), \qquad E[v(t)v(t+\tau)] = r_e \delta(\tau).$$

For the estimator

$$\dot{\hat{x}} = a\hat{x} + f_e(\hat{x} - y) = (a + f_e)\hat{x} - f_e y,$$

find the optimal estimator gain f_e.

Solution: the FARE is

$$2ap_e - \frac{1}{r_e}p_e^2 + q_e = 0$$

or

$$p_e^2 - 2ar_e p_e - r_e q_e = 0$$

with positive solution

$$p_e = ar_e + \sqrt{a^2 r_e^2 + r_e q_e}.$$

The optimal estimator gain is therefore

$$f_e = -\frac{p_e}{r_e} = -a - \sqrt{a^2 + \frac{q_e}{r_e}}.$$

Substituting the optimal gain in the estimator equation yields

$$\dot{\hat{x}} = -\sqrt{a^2 + \frac{q_e}{r_e}}\,\hat{x} - f_e y.$$

Note that the solution depends only on the ratio q_e/r_e of the intensities of process and measurement noise. It is instructive to consider the two limiting cases $q_e/r_e \to 0$ and $q_e/r_e \to \infty$.

The case $q_e/r_e \to 0$ corresponds to a situation with very large measurement noise. In this case, the estimator equation becomes

$$\dot{\hat{x}} = -|a|\hat{x} - f_e y,$$

because the square root is positive no matter whether a is positive or negative. For the estimator gain, we have to distinguish between these cases, thus

$$f_e = \begin{cases} 0, & a < 0 \\ -2a, & a \geq 0 \end{cases}$$

If measurement noise is very large and the plant is stable, the optimal solution is to use no measurements at all. If the plant is unstable, this is not possible because it would result in an unstable estimator. In this case the optimal solution is to place the estimator pole at the stable mirror image of the unstable plant pole; this is the dual to the result on high cost on control for the regulator problem.

The assumption $q_e/r_e \to \infty$ corresponds to a situation with no measurement noise. In this case, we have $f_e \to -\infty$, i.e. the optimal solution is to make the estimator dynamics infinitely fast.

A final comment on the noise matrices Q_e and R_e is in order. These parameters play the same role as the weight matrices Q and R in the dual regulator problem. Just as in real life problems these weight matrices are not uniquely defined by the problem description, but are rather used as convenient tuning parameters, the noise intensities are in many cases not precisely known, and the matrices Q_e and R_e are used as tuning parameters in the same way. In the above example it is easy to see that the parameter q_e/r_e can be used to tune the bandwidth of the estimator; if the bandwidth is too low, then q_e/r_e can be increased and vice versa.

LQG Control

Now that we have established the solution to the optimal estimation problem, we turn to the problem of finding an optimal controller for the plant (9.1) when the full state is not available for feedback and state estimate feedback must be used. First we must clarify what we mean by optimal in this case. The closed-loop system in Fig. 9.2 is redrawn in Fig. 9.3 in a form that clearly shows which parts of the loop form the controller. We let $K(s)$ denote the transfer function from the state error $r - y$ to u, and define the optimal control problem as the problem of finding the controller $K(s)$ that minimizes the stochastic cost function

$$V_{LQG} = \lim_{t \to \infty} E[x^T Q x + u^T R u]. \tag{9.10}$$

This problem is known as the LQG (Linear Quadratic Gaussian) control problem. The LQG problem represents all aspects of the optimal control problem we have discussed

so far: the cost function has the structure of the cost for the linear regulator problem with two terms penalizing state error and control energy respectively; it reflects the fact that the state is not available and state estimate feedback must be used instead, and the presence of process and measurement noise is taken into account by the stochastic plant model (9.1).
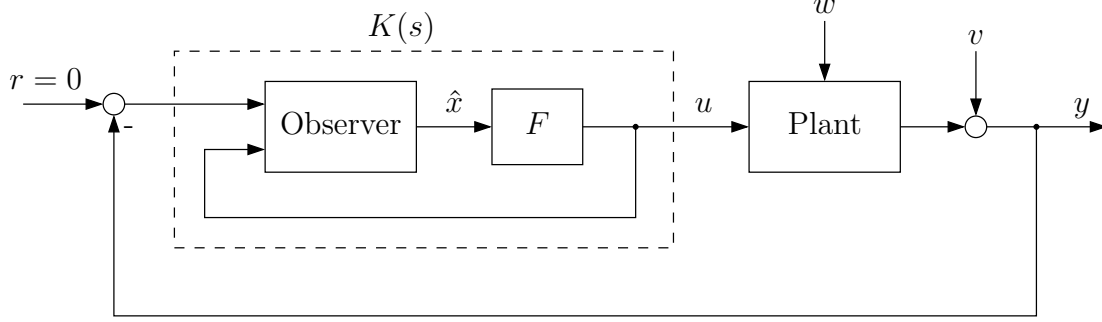

Figure 9.3: Closed-loop system with LQG controller $K(s)$

The Separation Principle

The solution to the LQG problem is based on the *separation principle*: the controller $K(s)$ that minimizes (9.10) is obtained by combining the optimal state feedback gain F and the optimal estimator gain F_e. Even though it may seem natural that the optimal controller should be the combination of optimal state feedback with optimal estimation, it is non-trivial to prove this fact. Here, we give only a brief outline of the proof.

Proof of Separation Principle

The proof is based on the following two properties of the optimal estimator:

1) State estimate and estimation error are uncorrelated, i.e. $E[\hat{x}\varepsilon^T] = 0$, and

2) the output error $y - C\hat{x}$ is white and zero mean, its covariance is
$E[(y - C\hat{x})(y - C\hat{x})^T] = R_e$ (this can be proved by using Kalman's identity).

Because $x = \hat{x} + \varepsilon$, we can use the first property above and rewrite the LQG cost as

$$V_{LQG} = \lim_{t \to \infty} E[\hat{x}^T Q \hat{x} + u^T R u + \varepsilon^T Q \varepsilon].$$

Thus, the LQG cost can be split into two terms that can be minimized independently: a term that represents the cost of estimation

$$V_e = \lim_{t \to \infty} E[\varepsilon^T Q \varepsilon],$$

and a term that represents the cost of control

$$V_c = \lim_{t \to \infty} E[\hat{x}^T Q \hat{x} + u^T R u].$$

The cost V_e is independent of the state feedback gain F, it depends only on the estimator gain F_e. The estimator gain that minimizes V_e is clearly the solution to the optimal estimation problem $F_e = -P_e C^T R_e^{-1}$, where P_e is the solution to the FARE. The cost V_c is independent of the estimator gain and depends only on the state feedback gain F; from property (2) of the optimal estimator we know that the state estimate $\hat{x}$ is the state x perturbed by an additive white, zero mean noise process. It can be shown that under this condition, the solution to the problem of minimizing V_c is the same as that to minimizing the deterministic cost function for the linear regulator problem: it is $F = -R^{-1} B^T P$, where P is the solution to the ARE. When a stochastic problem can be solved by replacing the stochastic variables with deterministic ones, we say that the *certainty equivalence principle* holds. ∎

In summary, the controller that minimizes the LQG cost (the *LQG controller*) is obtained by solving two Riccati equations: the ARE to find the optimal state feedback gain, and the FARE to find the optimal estimator gain.

Exercises

Problem 9.1

Given the autonomous plant model

$$\dot{x} = Ax, \qquad \text{A is stable} \qquad x(0) = x_0$$

and the cost function

$$V = \int_0^\infty x^T Q x \, dt.$$

Prove that the value of the cost function is given by

$$V = x_0^T P x_0, \tag{9.11}$$

where P is the positive definite matrix satisfying

$$PA + A^T P + Q = 0. \tag{9.12}$$

(That is, show that (9.3) and (9.4) are indeed the solution to the deterministic problem.)

Problem 9.2 LQG Control

Consider the system

$$\dot{x}(t) = x(t) + u(t) + w(t),$$
$$y(t) = x(t) + v(t)$$

where the process noise $w(t)$ and the measurement noise $v(t)$ are white, zero mean, Gaussian and uncorrelated, and satisfy

$$E[w(t)w(t+\tau)] = q_e\delta(\tau) = 1\delta(\tau), \qquad E[v(t)v(t+\tau)] = r_e\delta(\tau).$$

A controller $K(s)$ that minimizes

$$V_{LQG} = \lim_{t\to\infty} E[x^2(t) + \rho u^2(t)].$$

can be constructed from an optimal state feedback gain f and an optimal estimator gain f_e.

a) Draw the block diagram of the controller $K(s)$ and give its transfer function.

b) Find the optimal estimator gain f_e as a function of r_e.

c) The optimal state feedback gain was found to be $f = -3$. Find the corresponding value of ρ.

d) When $r_e = 0.125$, compare the error dynamics with the plant dynamics. Replace r_e by a value $\tilde{r}_e$ such that the error dynamics is at least five times faster than the plant.

e) The LQG cost can be written as

$$V_{LQG} = \lim_{t\to\infty} E[\hat{x}^2(t) + \rho u^2(t) + \varepsilon^2(t)]$$

where $\dot{\varepsilon}(t) = \alpha_e \varepsilon(t) + \xi(t)$. With your result from (d), what is α_e? Express the process $\xi(t)$ in terms of $w(t)$ and $v(t)$ and compute $E[\xi(t)\xi(t+\tau)]$.

Problem 9.3

Consider again the LQG loop of Problem 9.2. What is the transfer function from d to y with the designed controller (see Figure 9.4)? How does that transfer function change if the disturbance d enters directly at plant input without being feedback to the estimator (see Figure 9.5)?
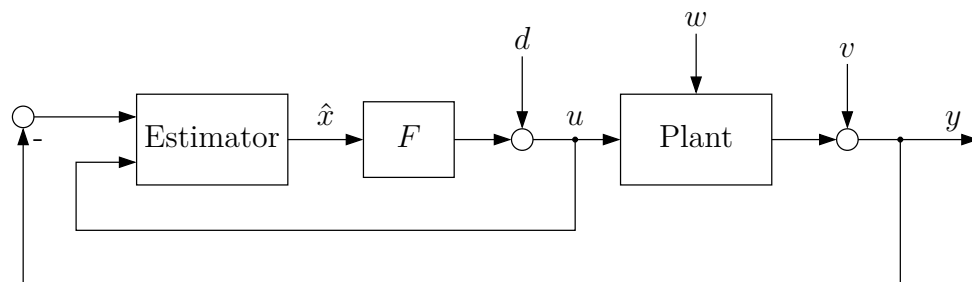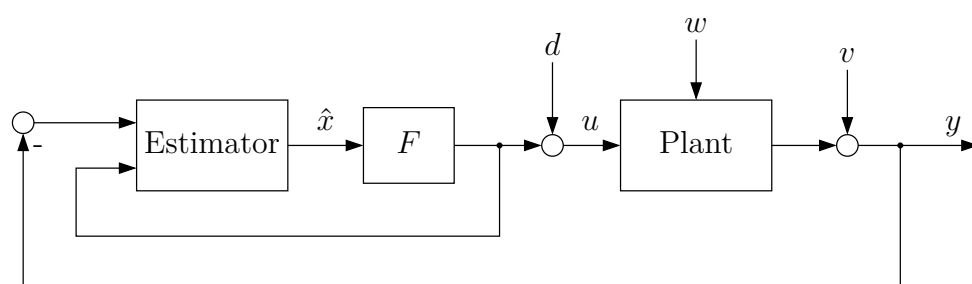
Figure 9.4: LQG Loop

Figure 9.5: Disturbance at plant input

Chapter 10

Loss of Robustness and Loop Transfer Recovery

In chapter 7, we studied the robustness properties of the closed-loop system with optimal state feedback. It turned out that the optimal closed-loop system has attractive robustness margins. In this chapter we will review these results for the state estimate feedback case. We first present an example which shows that the robustness margins for optimal state feedback control can completely disappear when state estimate feedback is used; and we will introduce a design technique that allows to partially recover these properties by manipulating the optimal estimator.

The following well known example was published in 1978 in the *IEEE Transactions on Automatic Control* by John Doyle.

Example 10.1

Consider an LQG design problem with the following data. The system matrices are

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad c = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

the cost weights are

$$Q = \gamma \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \qquad \rho = 1,$$

and the noise intensities

$$Q_e = \sigma \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \qquad r_e = 1.$$

The optimal state feedback and estimator gains can be computed as

$$f = \alpha \begin{bmatrix} 1 & 1 \end{bmatrix}, \qquad \alpha = -2 - \sqrt{4 + \gamma}$$

$$f_e = \beta \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad \beta = -2 - \sqrt{4 + \sigma}$$

To investigate the robustness of the optimal closed-loop system, we introduce a gain perturbation k. Assume the plant equation is

$$\dot{x} = Ax + kbu + w,$$

where $k = 1$ represents the nominal model for which the controller has been designed, and values $k \neq 1$ represent perturbations. With the closed-loop plant equation

$$\dot{x} = Ax + kbf\hat{x} + w,$$

and the estimator equation

$$\dot{\hat{x}} = (A + f_e c)\hat{x} + bf\hat{x} - f_e cx,$$

the closed-loop system matrix is

$$\bar{A} = \begin{bmatrix} A & kbf \\ -f_e c & A + bf + f_e c \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & k\alpha & k\alpha \\ -\beta & 0 & 1+\beta & 1 \\ -\beta & 0 & \alpha+\beta & 1+\alpha \end{bmatrix}.$$

A necessary condition for closed-loop stability is that all coefficients of the closed-loop characteristic polynomial

$$\det(sI - \bar{A}) = s^4 + \bar{a}_3 s^3 + \bar{a}_2 s^2 + \bar{a}_1 s + \bar{a}_0$$

are positive. We check only the absolute coefficient $\bar{a}_0$, it is

$$\bar{a}_0 = \det \bar{A} = 1 + (1 - k)\alpha\beta.$$

A necessary condition for stability is therefore

$$\bar{a}_0 > 0 \quad \Rightarrow \quad k < 1 + \frac{1}{\alpha\beta}.$$

The last inequality shows that there is no infinite gain margin in this case, in fact for large values of α or β the gain margin can become arbitrarily small.

Loop Transfer Function

The fact - illustrated by the above example (10.1) - that under state estimate feedback the robustness margins can become arbitrarily small, gave rise to research into methods to improve the robustness of LQG controllers. Here we present a method known as *LQG Loop Transfer Recovery*, which consists in manipulating the optimal estimator in order to recover the loop transfer function of state feedback.
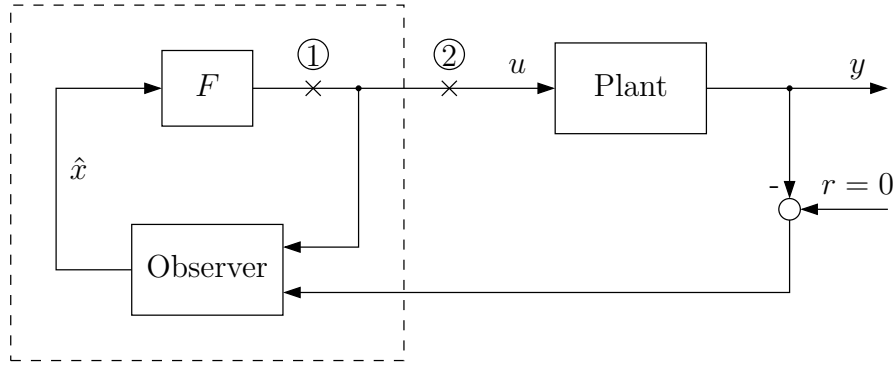
Figure 10.1: Loop break points

It was shown in chapter 7 that optimal state feedback has attractive robustness properties because the loop gain, or loop transfer function, satisfies Kalman's identity. To study the closed-loop properties of LQG control, we first need to find the loop transfer function for the state estimate feedback case. The closed-loop system is shown in Fig. 10.1. In the state feedback case, the loop gain was obtained by cutting the loop at plant input. In Fig. 10.1, two possible loop break points are marked by 1 and 2, and it makes a crucial difference at which of these points the loop is cut. When cut at point 1, the inputs to plant and observer are identical. A transfer function describes a system in steady state and initial states are ignored, therefore in this case we have $\hat{x} \equiv x$, and the loop gain is

$$L_1(s) = F(sI - A)^{-1}B.$$

However, it is not the loop gain seen from point 1 that determines the robustness properties of the control system, because the loop break point is inside the controller. The loop break point is the point where a disturbance or perturbation is expected to enter the loop, and it is unlikely that a perturbation is injected inside the controller. For a realistic assessment of the closed-loop robustness properties, it is necessary to cut the loop outside the controller, at point 2 just before the plant input.
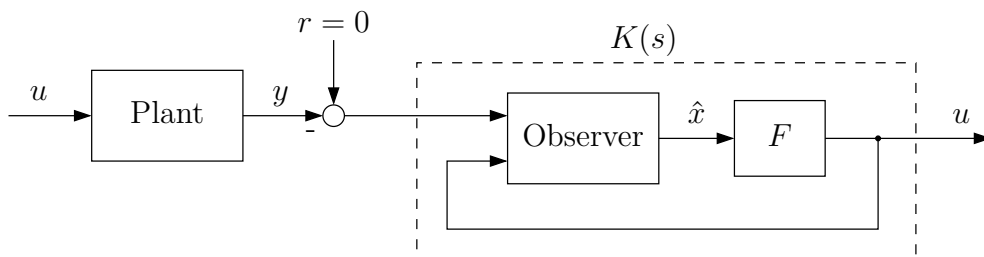

Figure 10.2: Loop broken at point 2

Fig. 10.2 shows the loop cut open at point 2; the loop transfer function $L_2(s)$ is the

negative product of plant transfer function and controller transfer function

$$L_2(s) = -K(s)C(sI - A)^{-1}B.$$

A block diagram of the controller is shown in Fig. 10.3. The state space representation of the controller is

$$\dot{\hat{x}} = (A + BF + F_eC)\hat{x} + F_e(-y)$$
$$u = F\hat{x}$$

and the controller transfer function (from $-y$ to u) is therefore

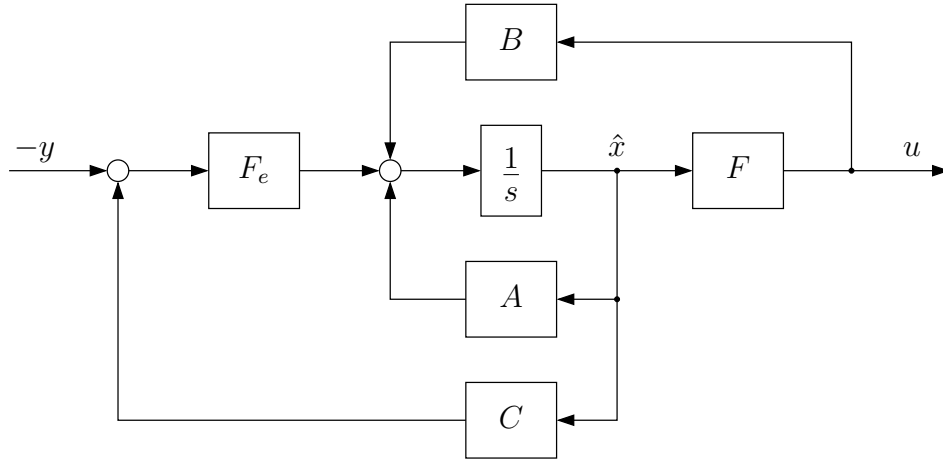$$K(s) = F(sI - A - BF - F_eC)^{-1}F_e.$$

Figure 10.3: LQG controller

Loop Transfer Recovery

We present now a technique to recover the optimal state feedback loop gain $L_1(s)$ with an LQG controller. For simplicity, we consider the single-input, single-output case. If an LQG design turns out to show poor robustness margins, then the idea is to pretend that there is white noise entering the loop at point 2, and to redesign the optimal estimator such as to cope with this fictitious noise. (Loosely speaking, the fictitious white noise at plant input can be viewed as representing plant uncertainty.) Therefore, we make the replacement

$$Q_e \quad \rightarrow \quad Q_e + \sigma bb^T$$

for the true process noise covariance, where σ is a tuning parameter that determines the intensity of the fictitious white noise process.

For the case when the estimator is designed for the fictitious noise intensity, we will prove the following: provided the plant is minimum phase, we have

$$\lim_{\sigma \to \infty} L_2(s) = f(sI - A)^{-1}b = L_1(s), \tag{10.1}$$

which means the optimal state feedback loop gain can be asymptotically recovered by letting the fictitious noise intensity approach infinity.

Proof The proof is based on the following asymptotic result for the optimal estimator gain: as $\sigma \to \infty$, we have

$$f_e = \pm \sqrt{\frac{\sigma}{r_e}} b, \tag{10.2}$$

where the sign is determined by the requirement that the estimator is stable. We will prove this fact later and first use it to show the main result (10.1). At this point we make the observation that as σ becomes large, the estimator gain will become large and thus the closed-loop system becomes sensitive against measurement noise. This indicates that LTR design involves a trade-off between closed-loop robustness and noise sensitivity.

We now prove (10.1). We will make repeated use of the matrix identity

$$(I - XY)^{-1}X = X(I - YX)^{-1},$$

which is particularly useful when X and Y are vectors x and y^T. Using the notation

$$\Phi(s) = (sI - A)^{-1}, \qquad \Phi_{cl}(s) = (sI - A - bf)^{-1},$$

the controller transfer function can be written as

$$\begin{aligned}
K(s) &= f(\Phi_{cl}^{-1} - f_e c)^{-1} f_e \\
&= f[\Phi_{cl}^{-1}(I - \Phi_{cl} f_e c)]^{-1} f_e \\
&= f(I - \Phi_{cl} f_e c)^{-1} \Phi_{cl} f_e \\
&= f\Phi_{cl} f_e (1 - c\Phi_{cl} f_e)^{-1}.
\end{aligned}$$

Note that the denominator in the last expression is a scalar. Taking the limit as $\sigma \to \infty$ and using (10.2), the controller transfer function becomes

$$K(s) = \frac{f\Phi_{cl}(s) f_e}{1 - c\Phi_{cl}(s) f_e} = -\frac{f\Phi_{cl}(s)b}{c\Phi_{cl}(s)b}.$$

The common factor $\Phi_{cl}(s)b$ of numerator and denominator can be written as

$$\Phi_{cl}(s)b = (\Phi^{-1} - bf)^{-1}b = (I - \Phi bf)^{-1}\Phi b = \Phi b(1 - f\Phi b)^{-1}.$$

Thus we have

$$K(s) = -\frac{f\Phi(s)b}{c\Phi(s)b}.$$

Substituting the last expression for $K(s)$ in the loop gain leads to

$$L_2(s) = -K(s)c\Phi(s)b = \frac{f\Phi(s)b}{c\Phi(s)b}c\Phi(s)b = f\Phi(s)b$$

or

$$L_2(s) = L_1(s) = f\Phi(s)b.$$

This shows that as $\sigma \to \infty$, the controller cancels the plant zeros and replaces them with the zeros of $f\Phi(s)b$; this explains why the plant must have no unstable zeros.

To complete the proof, we have to show that (10.2) holds. We will show this by duality. Setting $Q = \gamma c^T c$ in Kalman's identity and considering the regulator problem, we first prove that

$$\gamma \to \infty \quad \Rightarrow \quad f = \pm\sqrt{\frac{\gamma}{\rho}}\,c. \tag{10.3}$$

We have

$$\rho|1 - f\Phi b|^2 = \rho + \gamma|c\Phi b|^2.$$

Dividing by ρ and taking the limit as $\gamma \to \infty$, we obtain

$$|1 - f\Phi b| = \sqrt{\frac{\gamma}{\rho}}|c\Phi b|.$$

From this equation, we can draw two conclusions. The first one is that as $\gamma \to \infty$, the state feedback gain f must also become infinitely large, in fact it can be expressed as $f = \sqrt{\gamma}\tilde{f}$, where $\tilde{f}$ is some vector that does not depend on γ. The second observation here is that because the zeros of $1 - f\Phi(s)b$ are the closed-loop poles under state feedback, by root locus arguments they approach the zeros of $f\Phi(s)b$ as $\gamma \to \infty$. The fact that the closed-loop poles are stable then implies that the zeros of $f\Phi(s)b$ are also stable.

If f is infinitely large, we can make a further simplification to get

$$|f\Phi(s)b| = \sqrt{\frac{\gamma}{\rho}}|c\Phi(s)b|.$$

This equality forces the zeros of $f\Phi(s)b$ and $c\Phi(s)b$ to be either equal or reflections across the imaginary axis. The claim (10.3) holds if all zeros are equal. We know that the zeros of $f\Phi(s)b$ are stable; and because we assumed that the same is true for the plant transfer function $c\Phi(s)b$ we conclude that (10.3) holds. Now making the replacements

$$\gamma \to \sigma, \qquad \rho \to r_e, \qquad f \to fe^T, \qquad c \to b^T$$

shows that (10.2) holds by duality. This completes the proof of (10.1). ∎

Bibliography - Linear Quadratic Optimal Control

[1] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems*. Wiley-interscience New York, 1972, vol. 1.

[2] B. D. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Courier Corporation, 2007.

[3] P. Dorato, V. Cerone, and C. Abdallah, *Linear Quadratic Control: An Introduction*. Simon & Schuster, Inc., 1994.

[4] T. Kailath, *Linear Systems*. Prentice-Hall Englewood Cliffs, NJ, 1980, vol. 156.

Part II

OPTIMAL AND ROBUST CONTROL

Introduction

This course introduces modern design techniques for optimal and robust multivariable control systems. The state space methods for optimal controller design developed in the 1960s and 1970s (LQR/LQG) were found to suffer from being sensitive to modelling errors and parameter uncertainty. In the 1980s research activities turned to a new approach, where design objectives are achieved by minimizing the H_2 norm or H_∞ norm of suitable closed-loop transfer functions. LQG control can be interpreted as a special case of H_2 optimal control, while constraints on the H_∞ norm can be used to include robustness against modelling uncertainty into the controller design.

The new approach turned out to be closely related to the familiar LQG methods - the computation of both H_2 and H_∞ optimal controllers involves the solution of two algebraic Riccati equations. Moreover, H_2 optimal control can be shown to be a special case of H_∞ optimal control.

Conditions on the H_2 norm and the H_∞ norm represent different aspects of the design. The H_2 norm can be used to achieve an optimal trade-off between regulation and control effort, and to deal with noisy measurements. Constraints on the H_∞ norm on the other hand can guarantee closed-loop stability for a given range of uncertain parameters. However, neither H_2 nor H_∞ norm constraints give direct control of the closed-loop pole locations and thus of dynamic properties like the damping ratio.

In many practical applications it would be desirable to combine the three design aspects mentioned above: for example, one might want to design a controller that achieves a given damping ratio and speed of response (a condition on the pole locations), while at the same time guaranteeing robust stability in the face of model uncertainty (a H_∞ constraint). And of all controllers that satisfy these conditions one would like to pick the best one in the sense of a quadratic performance index (H_2 norm). Efficient methods for such a design have been developed in the 1990s. Instead of solving Riccati equations, one can express H_2 and H_∞ constraints as linear matrix inequalities (LMI). The advantage over the Riccati approach is that one can combine different constraints into a single convex optimization problem, including constraints on admissible pole regions which can also be expressed as LMI constraints.

A major problem with modern H_2 and H_∞ optimal control is the fact that the controllers

have the same dynamic order as the plant. If the plant to be controlled is of high dynamic order, the optimal design results in controllers that are difficult to implement. Moreover, the high order may cause numerical problems. For practical applications, it is therefore necessary to approximate the plant by a simplified, reduced order model. This can be done for example by using the method of balanced truncation or balanced residualization.

This course provides an introduction to the topics outlined above. First, the notion of a generalized plant as a way of expressing design specifications is introduced and a review of signal and system norms and of the singular value decomposition is given. Next, the role of the sensitivity and complementary sensitivity function in H_∞ optimal design is discussed and illustrated in a case study. It is shown how one can express constraints on the H_2 and H_∞ norm and pole region constraints as LMI constraints, and how one can compute controllers that satisfy such constraints on the closed-loop system. Finally the design of robust controllers is discussed. It is shown how parameter uncertainty can be included in the plant model and taken into account when designing the controller. Again, the approach is illustrated with a case study.

The emphasis in this course is on practical application, and theoretical aspects are discussed in so far as their understanding is important for using the design tools. On completion, students should be able to apply the relevant MATLAB and Simulink tools to solve optimal and robust design problems. The course includes two design exercises: one on mixed sensitivity design and one on robust controller design.

Chapter 11

The Concept of a Generalized Plant - A Review of LQG Control

In modern control, almost any design problem is represented in the form shown in Figure 11.1. Here, P is referred to as the *generalized plant* and K is the controller. The signal vector w represents external inputs to the control system, such as command inputs, disturbances and noise, u is the control input, v contains the measured plant signals that are used for feedback control, and z is a fictitious output vector that is used to express design specifications. We will model the dynamic systems P and K both as transfer function matrices $P(s)$ and $K(s)$ and in the form of state space realizations.
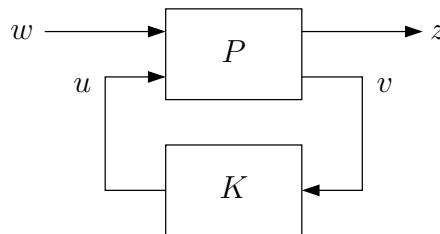

Figure 11.1: Control loop with generalized plant

To illustrate the concept of a generalized plant, we will express the problem of designing an LQG controller in the form of Figure 11.1. The control system is shown in Figure 11.2. The plant with transfer function model $G(s)$ is corrupted by process noise w_x and measurement noise w_y. We consider a state space realization of G

$$\dot{x} = Ax + Bu + w_x$$
$$y = Cx + w_y \tag{11.1}$$

where w_x and w_y are white noise processes

$$\mathrm{E}\left[w_x(t)w_x^T(t+\tau)\right] = Q_e\delta(\tau), \quad \mathrm{E}\left[w_y(t)w_y^T(t+\tau)\right] = R_e\delta(\tau), \quad \mathrm{E}\left[w_x(t)w_y^T(t+\tau)\right] = 0 \tag{11.2}$$
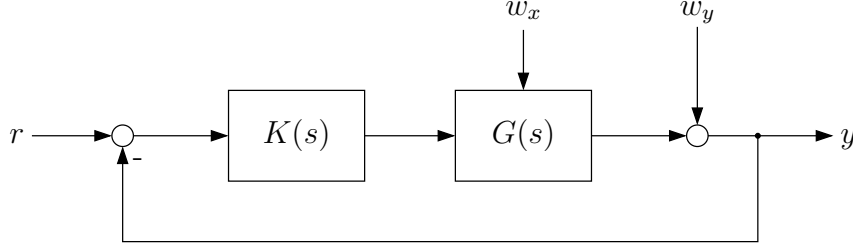

Figure 11.2: LQG control loop

Initially, we consider a regulation problem with zero reference input $r = 0$. The objective is to find a controller $K(s)$ that minimizes the LQG performance index

$$V = \lim_{T\to\infty} \mathrm{E}\left[\frac{1}{T}\int_0^T (x^T Q x + u^T R u)\,dt\right] \tag{11.3}$$

To express this design problem in the form of Figure 11.1, we start with a state space realization of P. With two inputs w and u, and two outputs z and v, we have

$$\dot{x} = A_p x + B_w w + B_u u$$
$$z = C_z x + D_{zw} w + D_{zu} u$$
$$v = C_v x + D_{vw} w + D_{vu} u \tag{11.4}$$

The first step is to relate the inputs and outputs of the generalized plant to the LQG problem under consideration. We take the control input u of the generalized plant to be the control input of the LQG problem, and the measured output v of the generalized plant to be the control error $e = -y$ in the LQG problem. Accordingly, we relate the plant model (A, B, C) in (11.1) to the subsystem (A_p, B_u, C_v) of the generalized plant model (11.4) by defining

$$A_p = A, \quad B_u = B, \quad C_v = -C, \quad D_{vu} = 0$$

where the definition of D_{vu} reflects the fact that the plant in (11.1) is strictly proper.

To express the cost function (11.3) in terms of the generalized plant, choose

$$C_z = \begin{bmatrix} Q^{1/2} \\ 0 \end{bmatrix}, \qquad D_{zu} = \begin{bmatrix} 0 \\ R^{1/2} \end{bmatrix}$$

Assume first that $w = 0$, then the squared integral of the fictitious output z has the same form as the integral in the LQG cost

$$\int_0^\infty z^T z\,dt = \int_0^\infty (x^T Q x + u^T R u)\,dt$$

We have not yet taken the noise processes into account. The external input w in (11.4) can be used to represent process and measurement noise as follows. Assume that w is a white noise process satisfying

$$\mathrm{E}\left[w(t)w^T(t+\tau)\right] = \delta(\tau)I$$

and choose

$$B_w = \begin{bmatrix} Q_e^{1/2} & 0 \end{bmatrix}, \qquad D_{vw} = \begin{bmatrix} 0 & R_e^{1/2} \end{bmatrix}, \qquad D_{zw} = 0$$

We then have

$$w_x = B_w w = \begin{bmatrix} Q_e^{1/2} & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = Q_e^{1/2} w_1$$

and

$$w_y = D_{vw} w = \begin{bmatrix} 0 & R_e^{1/2} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = R_e^{1/2} w_2$$

where w is partitioned to be compatible with the dimensions of Q_e and R_e. It is now easy to see that minimizing

$$\lim_{T \to \infty} \mathrm{E}\left[\frac{1}{T} \int_0^T z^T(t)z(t)\, dt\right]$$

is equivalent to minimizing the LQG performance index V in (11.3).

For the transfer function matrix $G(s)$ of a system

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

we will use the following notation in terms of its state space realization

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$$

Thus, the transfer function of a generalized plant that represents the LQG problem is

$$P(s) = \left[\begin{array}{c|c} A_p & B_p \\ \hline C_p & D_p \end{array}\right] = \left[\begin{array}{c|cc} A & \begin{bmatrix} Q_e^{1/2} & 0 \end{bmatrix} & B \\ \hline \begin{bmatrix} Q^{1/2} \\ 0 \end{bmatrix} & 0 & \begin{bmatrix} 0 \\ R^{1/2} \end{bmatrix} \\ -C & \begin{bmatrix} 0 & -R_e^{1/2} \end{bmatrix} & 0 \end{array}\right]$$

where 0 stands for zero matrix blocks of appropriate dimensions.

So far, the LQG problem was considered as a regulator problem, i.e. we assumed that $r = 0$ in Figure 11.2. How a reference input can be introduced into the above model to make it equivalent to the control system shown in Figure 11.2 is indicated in Figure 11.3. The external input w and the fictitious output z of the generalized plant are

$$w = \begin{bmatrix} r \\ w_1 \\ w2 \end{bmatrix} \qquad \text{and} \qquad z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$
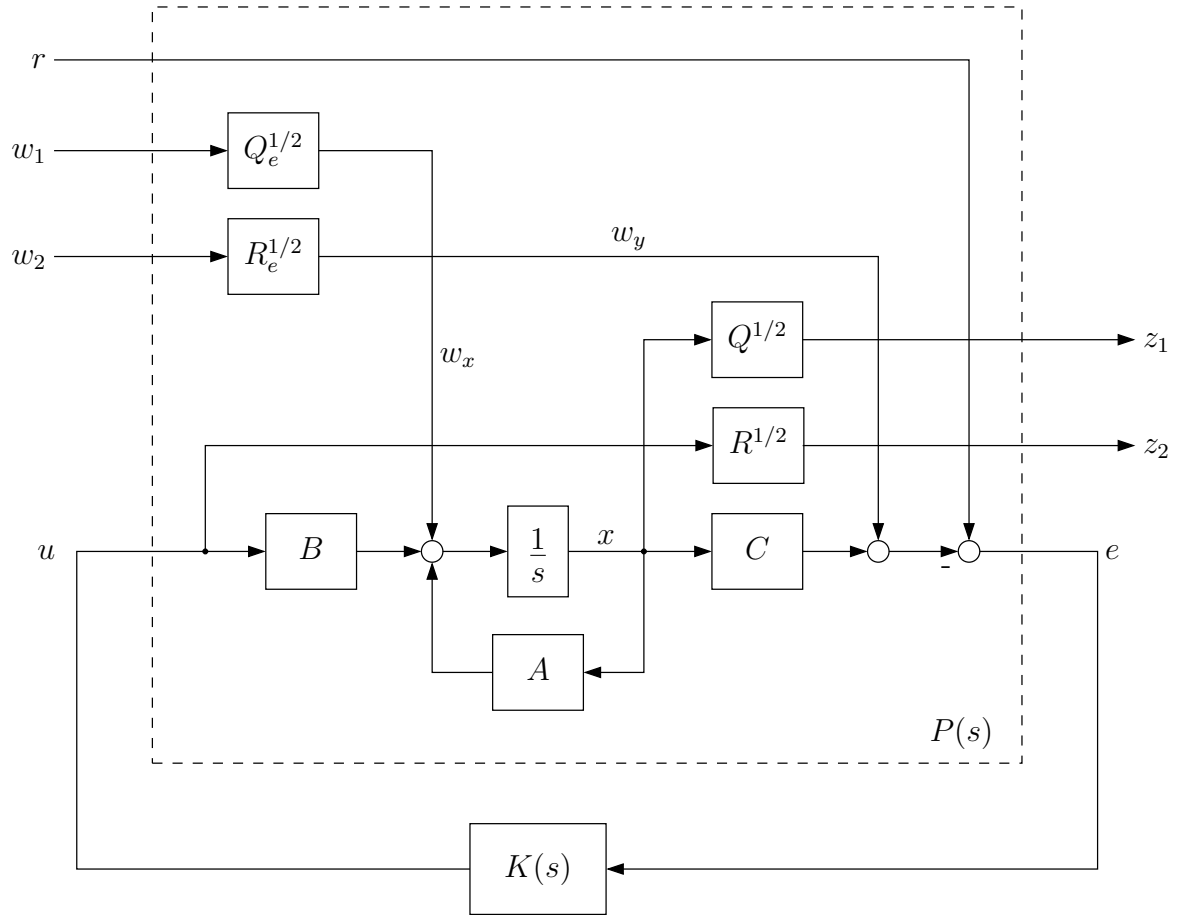
Figure 11.3: LQG problem represented in a generalized plant configuration

The input vector u of P is the same as that in Figure 11.2, and the measured output v is equal to the control error e.

Example 11.1

The following state space model represents a linearized model of the vertical-plane dynamics of an aircraft. It has three inputs, three outputs and five states. The inputs are

u_1 spoiler angle (in 0.1 deg)
u_2 forward acceleration (in m s^{-2})
u_3 elevator angle (in deg)

The states are
x_1 relative altitude (in m)
x_2 forward speed (in m s^{-1})
x_3 pitch angle (in deg)
x_4 pitch rate (in deg s^{-1})
x_5 vertical speed (in m s^{-1})

The matrices A, B, C and D are

$$A = \begin{bmatrix} 0 & 0 & 1.132 & 0 & -1 \\ 0 & -0.0538 & -0.1712 & 0 & 0.0705 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0485 & 0 & -0.8556 & -1.013 \\ 0 & -0.2909 & 0 & 1.0532 & -0.6859 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ -0.12 & 1 & 0 \\ 0 & 0 & 0 \\ 4.419 & 0 & -1.665 \\ 1.575 & 0 & -0.0732 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \qquad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Exercises

Problem 11.1

The function `aircraft_model()` generates the aircraft model from Example 11.1 and the Simulink model `aircraft_sim.mdl` provides a closed-loop for simulations.

a) Design an LQG controller for the aircraft. For this purpose, construct a state space representation of the generalised plant in Figure 11.3 and use the MATLAB function `h2syn` to design the LQG controller. The usage is

    ```
    K = h2syn( Gplant, nmeas, ncont )
    ```

    where the input argument `Gplant` is the generalized plant, `nmeas` is the number of measured outputs and `ncont` is the number of controller inputs. The returned output `K` is the LQG controller.

b) Design a controller for tracking a reference step input $r(t) = \begin{bmatrix} \sigma(t) & 0 & 0 \end{bmatrix}^T$ ("altitude acquire") subject to the constraint $|u_3| \le 20$ on the elevator angle.

Chapter 12

Vector Norms and Induced Norms

In the previous chapter it was shown how the LQG problem can be transformed into an equivalent problem in terms of the generalised plant in Figure 11.3. The objective of finding the controller $K(s)$ in Figure 11.2 that minimises the performance index V in (11.3), is then equivalent to finding the controller $K(s)$ in Figure 11.3 that minimises

$$\lim_{T \to \infty} \mathrm{E} \left[\frac{1}{T} \int_0^T z^T(t) z(t) \, dt \right]$$

where the input w is a white noise process with $\mathrm{E}\left[w(t)w^T(t+\tau)\right] = \delta(\tau)I$.

When a design problem is expressed in terms of a generalised plant, the objective is usually to minimise the "size" of the fictitious output signal z, under the condition that a specified type of signal is applied to the external input w. This leads to the concept of the *signal norm* as a measure of the "size" of a signal, and the *system norm* as a measure of the system "gain" (in terms of a given signal norm).

Vector Norms

To introduce signal and system norms, we start with a review of vector norms and induced norms. Consider a vector space $\mathcal{X}$. A norm $\|x\|$ is function mapping a vector x into a real number, that satisfies the following four properties for any $x, y \in \mathcal{X}$

1) $\|x\| \geq 0$

2) $\|x\| = 0 \quad \Leftrightarrow \quad x = 0$

3) $\|\alpha x\| = \alpha \|x\|$ for any scalar $\alpha > 0$

4) $\|x + y\| \leq \|x\| + \|y\|$

We first consider finite dimensional real or complex vector spaces, $\mathcal{X} = \mathbb{R}^n$ or $\mathcal{X} = \mathbb{C}^n$. Let $x = [x_1 \ x_2 \ \ldots \ x_n]^T$ be a vector with $x_i \in \mathbb{R}$, $i = 1, \ldots, n$ or $x_i \in \mathbb{C}$, $i = 1, \ldots, n$. A frequently used norm on x is the vector-p-norm

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}}$$

where p is a positive integer. Of practical importance are the three cases

$$\|x\|_1 = \sum_{i=1}^n |x_i|$$

$$\|x\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$$

$$\|x\|_\infty = \max_i |x_i|$$

The vector-2-norm can also be written as

$$\|x\|_2 = \sqrt{x^T x} \quad \text{if} \quad x \in \mathbb{R}^n \qquad \text{or} \qquad \|x\|_2 = \sqrt{x^H x} \quad \text{if} \quad x \in \mathbb{C}^n$$

where A^H denotes the *Hermitian* of a matrix A: $A^H = \bar{A}^T$ where $\bar{A}$ is the complex conjugate of A. In this course we will only use the 2-norm for vectors, and we will usually drop the subscript and write $\|x\|$ for the vector-2-norm of x.

Signal Norms

Now consider the space of scalar, real or complex valued signals $x(t)$. The p-norm of a signal is defined in a way similar to the vector p-norm as

$$\|x(t)\|_p = \left(\int_{-\infty}^\infty |x(\tau)|^p \, d\tau \right)^{\frac{1}{p}}$$

Here we are interested in the signal-2-norm

$$\|x(t)\|_2 = \sqrt{\int_{-\infty}^\infty |x(\tau)|^2 \, d\tau}$$

We will usually drop the subscript on the signal-2-norm. Note that taking the magnitude in the vector-2 and signal-2-norm is only required for complex vectors.

There are signals for which the signal-2-norm as defined above does not exist. As an example, consider the signals shown in Figure 12.1: the signal $x_1(t)$ has a finite 2-norm, whereas the 2-norm of the sinusoidal signal $x_2(t)$ is infinite. These two types of signals are referred to as *energy signals* and *power signals*, respectively. Energy signals have a finite 2-norm. For power signals, the 2-norm does not exist, but we can define the *root mean square* (rms) value as

$$\|x(t)\|_{\text{rms}} = \lim_{T \to \infty} \sqrt{\frac{1}{2T} \int_{-T}^T |x(\tau)|^2 \, d\tau}$$
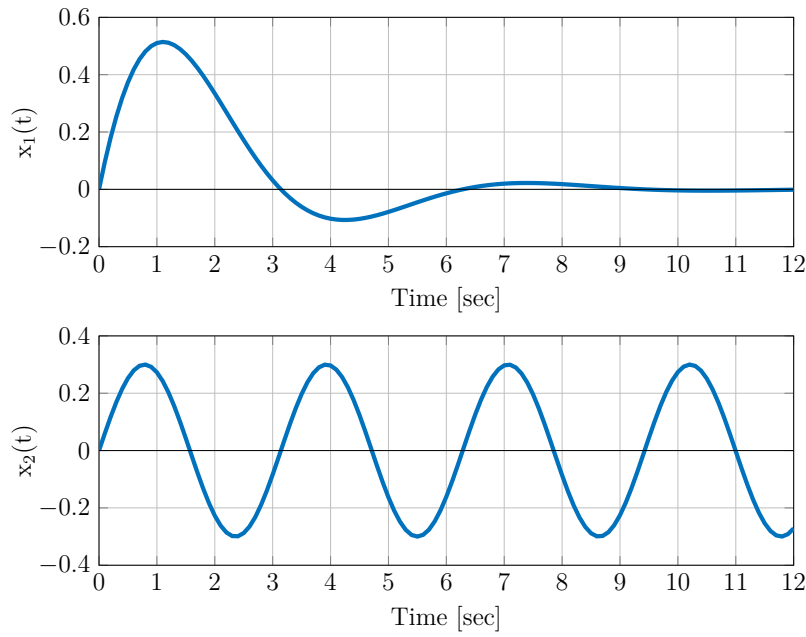
Figure 12.1: Energy signal $x_1(t)$ and power signal $x_2(t)$

Note that this is strictly speaking not a norm, because it does not satisfy property (2) of a norm.

Now consider a real or complex valued signal vector $x(t) = [x_1(t) \quad x_2(t) \quad \ldots \quad x_n(t)]^T$. Its signal-2-norm is defined as

$$\|x(t)\|_2 = \sqrt{\int_{-\infty}^{\infty} \|x(\tau)\|^2 \, d\tau}$$

The integral is now taken over the square of the vector-2-norm of $x(t)$. Note that at a given time τ, $x(\tau)$ is simply a real or complex vector, i.e. $x(\tau) \in \mathbb{R}^n$ or $x(\tau) \in \mathbb{C}^n$, and that the same norm symbol denotes on the left hand side in the above equation a signal norm and on the right hand side a finite dimensional vector norm.

The Matrix-2-Norm

Consider two complex vectors $x \in \mathbb{C}^n$ and $y \in \mathbb{C}^m$, and a linear mapping $y = Ax$. The complex matrix $A \in \mathbb{C}^{m \times n}$ can be interpreted as an operator that maps x into y, or more generally $\mathbb{C}^n$ into $\mathbb{C}^m$. One can compare the vector norms of x and y, and associate a "gain" with A as the ratio of these vector norms. This ratio depends on x, and an important property of the matrix A is the maximum value of $\|y\|/\|x\|$ over all nonzero $x \in \mathbb{C}^n$ (the "maximum gain" of A). This positive real number can be defined as a norm of the matrix A; since it depends also on the choice of vector norm, it is called an *induced norm*. The

matrix-2-norm induced by the vector-2-norm is defined as

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \tag{12.1}$$

Again, we will usually drop the subscript and write $\|A\|$ for the matrix-2-norm. It is straightforward to verify that the matrix-2-norm - and indeed all induced matrix-p-norms - satisfy the four properties of a norm.

To find the value of $\|A\|$, we take squares on both sides of (12.1) to get

$$\|A\|^2 = \max_{x \neq 0} \frac{\|Ax\|^2}{\|x\|^2} = \max_{x \neq 0} \frac{x^H A^H A x}{x^H x} = \max_{x \neq 0} \frac{x^H M x}{x^H x}$$

where we introduced the Hermitian matrix $M = A^H A$. In order to find the maximum value of the last term, we diagonalize M, i.e. we compute $M = V \Lambda V^{-1}$, where Λ is a diagonal matrix containing the eigenvalues of M, and V is a matrix with right eigenvectors of M as columns.

We will first establish the following useful facts about the Hermitian matrix $M = A^H A$

1) M is positive semi-definite ($x^H M x \geq 0 \ \forall x \in \mathbb{C}^n$)

2) the eigenvalues of M are real

3) the eigenvectors of M are orthogonal (two vectors x and y are orthogonal if $x^H y = 0$)

With $y = Ax$, property (1) follows immediately from

$$x^H M x = x^H A^H A x = y^H y \geq 0$$

Note that this implies that $x^H M x$ is real even if x is complex. That the eigenvalues of M are real can be shown as follows. Let λ be an eigenvalue and v be an eigenvector of M, and consider

$$Mv = \lambda v$$

Multiplying with v^H from the left yields $v^H M v = \lambda v^H v$. We established already that the left hand side of this equation is real, and on the right hand side $v^H v$ is also real. Thus, λ must be real.

To show that two eigenvectors of M belonging to different eigenvalues are orthogonal, consider

$$Mv_1 = \lambda_1 v_1, \qquad Mv_2 = \lambda_2 v_2, \qquad \lambda_1 \neq \lambda_2$$

We have

$$(\lambda_1 v_1)^H v_2 = (Mv_1)^H v_2 = v_1^H M v_2 = v_1^H \lambda_2 v_2$$

thus $\lambda_1 v_1^H v_2 = \lambda_2 v_1^H v_2$, and from the assumption $\lambda_1 \neq \lambda_2$ it then follows that $v_1^H v_2 = 0$.

A consequence of property (3) is that if all eigenvectors v_i of M are normalized such that $\|v_i\| = 1$, $i = 1, \ldots, n$, the eigenvector matrix V is *unitary*, i.e. $V^H V = I$, or $V^{-1} = V^H$. (Strictly speaking, we have shown this only for matrices with distinct eigenvalues. It can be shown however that even a matrix with repeated eigenvalues has a full set of orthogonal eigenvectors.)

Note that properties (2) and (3) are true for any Hermitian matrix even when it is not positive semidefinite.

We now return to finding the value of $\|A\|$ by solving

$$\max_{x \neq 0} \frac{x^H A^H A x}{x^H x}$$

With the diagonalization $A^H A = V \Lambda V^H$ this becomes

$$\max_{x \neq 0} \frac{x^H V \Lambda V^H x}{x^H x}$$

and introducing $y = V^H x$ and thus $x = Vy$ (using orthonormality of V), we obtain

$$\max_{y \neq 0} \frac{y^H \Lambda y}{y^H V^H V y} = \max_{y \neq 0} \frac{y^H \Lambda y}{y^H y} = \max_{y \neq 0} \frac{\lambda_1 |y_1|^2 + \lambda_2 |y_2|^2 + \ldots + \lambda_n |y_n|^2}{|y_1|^2 + |y_2|^2 + \ldots + |y_n|^2}$$

where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of $A^H A$. Assume that the eigenvalues are ordered such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$. Then it is easy to see that the maximum value of the above expression is λ_1, which is achieved if we choose $y = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}^T$, and the minimum value is λ_n, achieved by choosing $y = \begin{bmatrix} 0 & \ldots & 0 & 1 \end{bmatrix}^T$.

Because the above expression is the square of the matrix-2-norm of A, we have thus established that

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \sqrt{\lambda_{\max}(A^H A)}$$

and we also found that

$$\min_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \sqrt{\lambda_{\min}(A^H A)}$$

The square roots of the eigenvalues of $A^H A$ are called the *singular values* of A. This leads us to the most important and useful of matrix factorizations, the *singular value decomposition* (SVD), which is discussed in the next chapter.

Exercises

Problem 12.1 Show that $\|AB\| \leq \|A\| \|B\|$

Chapter 13

The Singular Value Decomposition

In the last chapter we used the fact that any Hermitian matrix M can be factored into

$$M = V\Lambda V^H$$

where V is the eigenvector matrix of M and unitary, and Λ is the diagonal eigenvalue matrix of M. The same factorization is obviously not possible for non-Hermitian or even non-square matrices. A similar factorization is however possible in these cases, if we do not insist on the same matrix V on both sides, but allow different unitary matrices U and V as left and right factors.

Theorem 13.1 (Singular Value Decomposition) *For every matrix $A \in \mathbb{C}^{m \times n}$ there exist unitary matrices $U \in \mathbb{C}^{m \times m}$ and $V \in \mathbb{C}^{n \times n}$ such that*

$$A = U\Sigma V^H \tag{13.1}$$

and Σ is real and diagonal with non-negative entries.

The matrix Σ has the same size as A. For example, if A is a 3×2 or 2×3 matrix, then

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{bmatrix} \qquad \text{or} \qquad \Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \end{bmatrix}$$

respectively, where $\sigma_{1,2} \geq 0$. The diagonal entries σ_i are called the *singular values* of A.

From (13.1) we obtain $AV = U\Sigma$ and thus

$$Av_i = \sigma_i u_i, \qquad i = 1, \ldots, n$$

where v_i and u_i are the columns of V and U, respectively. Compare this with $Mv_i = \lambda_i v_i$ - an eigenvector v_i is transformed into $\lambda_i v_i$, whereas A transforms v_i into $\sigma_i u_i$. From (13.1) we also have

$$AA^H = U\Sigma V^H V \Sigma^T U^H = U\Sigma\Sigma^T U^H \tag{13.2}$$

and

$$A^H A = V \Sigma^T U^H U \Sigma V^H = V \Sigma^T \Sigma V^H \tag{13.3}$$

Equation (13.2) shows that U is the eigenvector matrix of AA^H, and (13.3) shows that V is the eigenvector matrix of $A^H A$. The eigenvalue matrices are $\Sigma \Sigma^T$ and $\Sigma^T \Sigma$, respectively. Again, if A is 3×2 then

$$\Sigma \Sigma^T = \begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \Sigma^T \Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

This shows that the singular values of A are the square roots of the eigenvalues of AA^H and $A^H A$.

Proof of Theorem 13.1

To prove Theorem 13.1, we show how to construct U, V and Σ that satisfy (13.1) for a given matrix A. We start with the diagonalization of $A^H A$: we established already that there exists a unitary matrix V such that

$$A^H A = V \Lambda V^H$$

where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is the diagonal eigenvalue matrix of $A^H A$, and the columns v_i of V are the corresponding eigenvectors. Thus

$$A^H A v_i = \lambda_i v_i \qquad \text{and} \qquad v_i^H A^H A v_i = \lambda_i v_i^H v_i = \lambda_i$$

because V is unitary, and therefore

$$\|A v_i\|^2 = \lambda_i \tag{13.4}$$

This implies that $\lambda_i \geq 0$. Assume that the eigenvalues $\lambda_1, \ldots, \lambda_r$ are positive and the remaining $n - r$ eigenvalues λ_i and vectors Av_i are zero. Note that $r \leq \min(n, m)$. Define

$$\sigma_i = \sqrt{\lambda_i}, \qquad u_i = \frac{1}{\sigma_i} A v_i, \qquad i = 1, \ldots, r$$

It follows from (13.4) that $\|u_i\| = 1$. Moreover, we have

$$u_i^H u_j = \frac{v_i^H A^H A v_j}{\sigma_i \sigma_j} = \frac{\lambda_i v_i^H v_j}{\sigma_i \sigma_j} = 0, \qquad i \neq j$$

This shows that the vectors $u_1, \ldots, u_r$ defined above have the properties required of column vectors for U to be unitary. If $r < m$, one can fill up the matrix U with $m - r$ further orthogonal vectors (by using Gram-Schmidt orthogonalization) to make it into a $m \times m$ unitary matrix.

Now it remains to show that the matrices U, V as defined above satisfy

$$U^H A V = \Sigma$$

where Σ is diagonal with σ_i as diagonal entries. The (i,j) entry of $U^H A V$ is

$$(U^H A V)_{i,j} = u_i^H A v_j = \begin{cases} \sigma_j u_i^H u_j, & j \leq r \\ 0, & j > r \end{cases}$$

Because $\sigma_j u_i^H u_j$ is zero if $i \neq j$ and σ_j if $i = j$, the above shows that the entries of $U^H A V$ are all zero except for the first r entries on the main diagonal, which are the singular values of A. This completes the proof. ∎

Exercises

Problem 13.1 Calculate the singular value decomposition of the matrices

$$\begin{bmatrix} 2 & 1 & 1 \\ 4 & 2 & 3 \\ 6 & 3 & 4 \end{bmatrix} ; \qquad \begin{bmatrix} 2 & 0 \\ 0 & -3 \\ 0 & 0 \end{bmatrix} ; \qquad \begin{bmatrix} -1 \\ 2 \\ 2 \end{bmatrix}$$

Problem 13.2 Use the singular value decomposition to calculate the rank of the matrices

$$\begin{bmatrix} \epsilon & 2\epsilon \\ 1 & 2 \end{bmatrix} ; \qquad \begin{bmatrix} \epsilon & 1 \\ 0 & 0 \end{bmatrix} ; \qquad \begin{bmatrix} \epsilon & 1 \\ \epsilon & 1+\epsilon \end{bmatrix}$$

where ϵ is very small number. Discuss the result.

Problem 13.3

Suppose a satellite takes the following picture and wants to send it to earth:

$$\text{Image} = \begin{bmatrix} 38 & 66 & 20 & 47 & 40 & 49 & 55 & 97 & 75 & 69 \\ 90 & 164 & 54 & 121 & 97 & 126 & 135 & 235 & 186 & 171 \\ 77 & 141 & 45 & 104 & 83 & 108 & 113 & 201 & 160 & 145 \\ 77 & 140 & 49 & 103 & 78 & 113 & 112 & 201 & 158 & 142 \\ 96 & 172 & 61 & 128 & 90 & 146 & 132 & 249 & 189 & 175 \\ 96 & 176 & 30 & 117 & 153 & 76 & 167 & 243 & 218 & 194 \\ 55 & 98 & 33 & 72 & 56 & 80 & 79 & 144 & 112 & 103 \\ 81 & 146 & 56 & 113 & 69 & 135 & 110 & 214 & 160 & 145 \\ 41 & 77 & 23 & 55 & 50 & 53 & 63 & 107 & 88 & 80 \\ 83 & 151 & 57 & 116 & 68 & 137 & 111 & 217 & 164 & 149 \end{bmatrix}$$

(8-Bit Image, gray-level (0-255), Resolution (10×10) pixels.)

The picture contains $10 \cdot 10 \cdot 8 = 800$ bits. Try to reduce the number of transmitted Bits such that the image can still be recovered at earth.

Note: the image matrix `Image` can be downloaded from the web site of this course.

Hint: Find the essential information in this image using SVD.

Chapter 14

System Norms

In this chapter, we introduce two system norms: the H_∞ norm and the H_2 norm. The H_∞ norm is a measure for the "gain" of a system in terms of the signal-2-norm - it is an induced norm. The H_2 norm on the other hand is not induced by a signal norm. We will first discuss both system norms for single-input single-output systems and then for multivariable systems.

The H_∞ Norm for Single-Input Single-Output Systems

For a stable, proper single-input single-output (SISO) system with transfer function $G(s)$, the H_∞ norm is defined as

$$\|G(s)\|_\infty = \sup_\omega |G(j\omega)|$$

Here "sup" stands for *supremum* or *least upper bound*, and means the same as "max", but in addition allows for the possibility that its value is only approached asymptotically.

$$u(t) \longrightarrow \boxed{G(s)} \longrightarrow y(t)$$
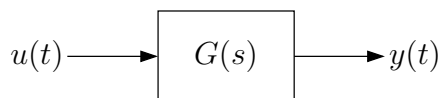
Figure 14.1: Single-input single-output system

A graphical interpretation of this definition is shown in Figure 14.2. The H_∞ norm of a SISO system is simply the maximum gain over all frequencies, and can be read off the Bode magnitude plot of the frequency response.

For a system with input $u(t)$ and output $y(t)$ (see Figure 14.1), the H_∞ norm is equal to the norm induced by the signal-2-norm:

$$\|G(s)\|_\infty = \max_{u \neq 0} \frac{\|y(t)\|}{\|u(t)\|}$$
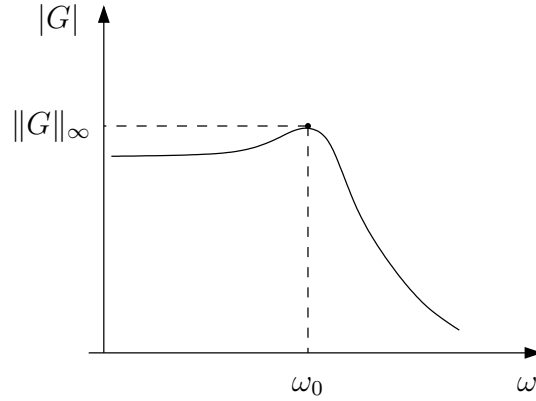
Figure 14.2: H_∞ norm as peak value in the frequency response

if $u(t)$ is an energy signal, and

$$\|G(s)\|_\infty = \max_{u \neq 0} \frac{\|y(t)\|_{\text{rms}}}{\|u(t)\|_{\text{rms}}}$$

if $u(t)$ is a power signal. We will not prove this fact here; referring to Figure 14.2 it should however be clear that the power signal leading to the largest "gain" in terms of rms values is a sinusoidal input at frequency ω_0. In fact, the steady state response to an input $u(t) = \sin \omega_0 t$ is

$$y(t) = \|G(s)\|_\infty \sin(\omega_0 t + \phi)$$

where $\phi = \arg G(j\omega_0)$ is the phase shift at that frequency. Note that the phase response has no effect on the H_∞ norm of the system.

The H_2 Norm for SISO Systems

The H_2 norm for a SISO system with a stable, strictly proper transfer function $G(s)$ is defined as

$$\|G(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} |G(j\omega)|^2 \, d\omega}$$

The restriction to strictly proper systems is necessary because otherwise $|G(j\omega)| > 0$ as $\omega \to \infty$ and the integral does not exist. This norm is not induced by a signal norm, but there are two important interpretations of this norm in terms of certain input signals - one stochastic and one deterministic.

For a stochastic interpretation of the H_2 norm, assume that the input $u(t)$ is white noise with $\text{E}[u(t)u(t + \tau)] = \delta(\tau)$. In that case, the rms value of the output signal is equal to the H_2 norm of the system

$$\|y(t)\|_{\text{rms}} = \|G(s)\|_2$$

This fact makes it possible to express the LQG problem as the problem of minimizing the H_2 norm of the generalized plant introduced in chapter 11.

A deterministic interpretation goes as follows. Let

$$\dot{x} = Ax + bu, \qquad y = cx$$

be a state space realization of $G(s)$; note that $D = 0$ because $G(s)$ is strictly proper. Let

$$g(t) = ce^{At}b$$

denote the impulse response of the system. By Parseval's theorem, the above definition of the H_2 norm is then equivalent to the following definition in terms of the impulse response

$$\|G(s)\|_2 = \sqrt{\int_0^\infty |g(t)|^2 \, dt} = \|g(t)\|_2$$

The H_2 norm of the system is equal to the signal-2-norm of its impulse response. This makes it possible to express the deterministic LQR problem as a H_2 optimization problem.

The H_∞ Norm for MIMO Systems

Consider the system shown in Figure 14.3, where input and output are now assumed to be vector signals. For SISO systems, the H_∞ norm was defined as the maximum gain over frequency. Now we are dealing with a transfer function matrix $G(s)$, and the definition of the H_∞ norm should be consistent with the interpretation as a system norm induced by the signal-2-norm or the rms value of power signals.

$$u(t) \longrightarrow \boxed{\quad G(s) \quad} \longrightarrow y(t)$$
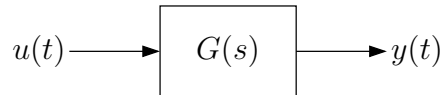
Figure 14.3: Multi-input multi-output system

What is different now is the fact that it is not only the "size" of the signals that matters, but also the "direction" of the signal vectors. To illustrate this point, consider the following 2 by 2 system consisting only of constant gains.

Example 14.1

For a system with transfer function matrix

$$G(s) = \begin{bmatrix} 3 & 0 \\ 4 & 1 \end{bmatrix}$$

the response to a constant input $u(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}^T \sigma(t)$ is $y(t) = \begin{bmatrix} 3 & 4 \end{bmatrix}^T \sigma(t)$, and $\|y(t)\|_{\text{rms}} = 5/\sqrt{(2)}$. The response to $u(t) = \begin{bmatrix} 0 & 1 \end{bmatrix}^T \sigma(t)$ is $y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix}^T \sigma(t)$, and $\|y(t)\|_{\text{rms}} =$

$1/\sqrt{(2)}$. Since in both cases $\|u(t)\|_{\text{rms}} = 1/\sqrt{(2)}$, the "gain" in terms of rms values is 5 for the first input signal and 1 for the second input signal.

To introduce the definition of the H_∞ norm for MIMO systems, we will again consider sinusoidal input signals, and we will find the combination of inputs that maximizes the output signal. The above example showed that we need to take the direction of the input signal into account. With a vector of sinusoidal input signals, it is not only the direction in terms of the magnitude (amplitude) of the individual signals, but also their phase that has an effect on the output. To account for both magnitude and phase, we will introduce complex signals and complex "directions".

A sinusoidal input with amplitude u_0 and phase φ can be interpreted as the imaginary part of a complex signal

$$u(t) = u_0 \sin(\omega t + \varphi) = \text{Im} \left[u_0 e^{j(\omega t + \varphi)} \right]$$

Now define the complex signal

$$\tilde{u}(t) = u_0 e^{j(\omega t + \varphi)} = \hat{u} e^{j\omega t}$$

where the complex amplitude

$$\hat{u} = u_0 e^{j\varphi}$$

contains both magnitude and phase information about the signal. The steady state response to the complex input signal $\tilde{u}(t) = \hat{u} e^{j\omega t}$ is

$$\tilde{y}(t) = \hat{y} e^{j\omega t}$$

where the complex amplitude of the output signal is the amplitude of the input signal multiplied by the transfer function evaluated at $s = j\omega$

$$\hat{y} = G(j\omega)\hat{u}$$

The physical signals $u(t)$ and $y(t)$ can be recovered by taking the imaginary parts of the complex signals. We now apply the above signal representation with complex amplitudes to a multivariable system with m inputs and l outputs. At a given frequency ω, the input and output signals are

$$\tilde{u}(t) = \begin{bmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_m \end{bmatrix} e^{j\omega t} = \hat{u} e^{j\omega t}, \qquad \tilde{y}(t) = \begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_l \end{bmatrix} e^{j\omega t} = \hat{y} e^{j\omega t}$$

where $\tilde{u}(t)$ is applied to the system input and $\tilde{y}(t)$ is the response in steady state. To find the induced 2-norm of the system, it is enough to consider the amplitude vectors $\hat{u}$ and $\hat{y}$ because

$$\frac{\|\tilde{y}(t)\|_{\text{rms}}}{\|\tilde{u}(t)\|_{\text{rms}}} = \frac{\|\hat{y}\|}{\|\hat{u}\|}$$

The output amplitude vector at a given frequency ω is obtained by multiplying the input amplitude vector with $G(j\omega)$. For example if the system is 2 by 2, we have

$$\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} = \begin{bmatrix} G_{11}(j\omega) & G_{12}(j\omega) \\ G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix}$$

Note that for a given frequency, the transfer function matrix is just a complex matrix $G(j\omega) \in \mathbb{C}^{l \times m}$. The maximum value of $\|\hat{y}\|/\|\hat{u}\|$ *at that frequency* is given by the maximum singular value of the transfer function matrix

$$\max_{\hat{u} \neq 0} \frac{\|\hat{y}\|}{\|\hat{u}\|} = \bar{\sigma}(G(j\omega))$$

We will use the notation $\bar{\sigma}(G)$ and $\underline{\sigma}(G)$ for the maximum and minimum singular value of G. The H_∞ norm of the system is defined as the maximum value of this induced matrix norm *over all frequencies*

$$\|G(s)\|_\infty = \sup_\omega \bar{\sigma}(G(j\omega))$$
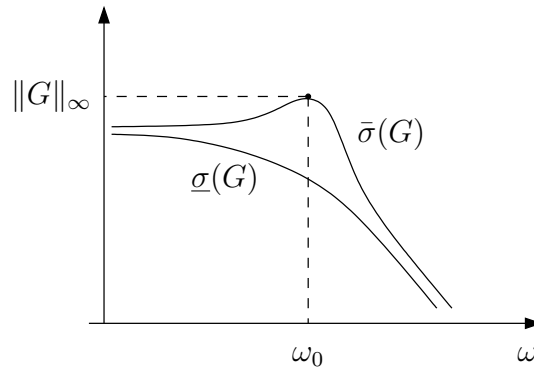
Figure 14.4: H_∞ norm as peak value in the singular value plot

A graphical interpretation of the H_∞ norm is shown in Figure 14.4: it is the peak value of the maximum singular value over frequency. Singular value plots of multivariable systems usually show the minimum and the maximum singular value - indicating the minimum and maximum gain at a given frequency over all input directions.

The H_2 Norm for MIMO Systems

To extend the definition of the H_∞ norm to MIMO systems, we used the induced 2-norm for matrices, which is equal to the maximum singular value. For the H_2 norm, we need to introduce a new matrix norm. The *Frobenius norm* of a matrix A is defined as

$$\|A\|_F = \sqrt{\text{trace}(A^H A)}$$

The definition of the H_2 norm of a multivariable system is

$$\|G(s)\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \|G(j\omega)\|_F^2 \, d\omega}$$

Again, using Parseval's theorem, one can show that an equivalent definition in time domain is

$$\|G(s)\|_2 = \sqrt{\int_0^{\infty} \|g(t)\|_F^2 \, dt}$$

where $g(t) = Ce^{At}B$ is the impulse response matrix of the system.

Exercises

Problem 14.1

For the aircraft model in Exercise 11.1, plot the singular values of the closed-loop system with an LQG controller over frequency (you can use the MATLAB command `sigma`). What do the singular values at low frequencies tell you about the systems tracking capability? What does the increasing difference between minimum and maximum singular values at high frequencies indicate about the systems sensitivity against measurement noise?

Chapter 15

Computing System Norms

In this chapter we show how to compute the H_2 norm and the H_∞ norm of a given transfer function. The H_2 norm can be computed by solving a single Lyapunov equation, while the computation of the H_∞ norm is an iterative procedure. The computation of both norms is based on a state space realization

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

of the transfer function $G(s)$.

Computing the H_2 norm

The H_2 norm is defined only for plants with a strictly proper transfer function, thus we assume $D = 0$. From the definition we have

$$\|G(s)\|_2 = \sqrt{\text{trace} \int_0^\infty (g^T(t)g(t)) \; dt}$$

where

$$g(t) = Ce^{At}B$$

is the impulse response matrix. Substituting this expression in the above equation and taking squares yields

$$\|G(s)\|_2^2 = \text{trace} \int_0^\infty B^T e^{A^T t} C^T C e^{At} B \; dt \tag{15.1}$$

$$= \text{trace}\, B^T \int_0^\infty e^{A^T t} C^T C e^{At} \; dt B \tag{15.2}$$

Defining

$$W_o = \int_0^\infty e^{A^T t} C^T C e^{At} \; dt \tag{15.3}$$

we have

$$\|G(s)\|_2 = \sqrt{\text{trace}\, B^T W_o B} \tag{15.4}$$

and it is straightforward to show (compare Exercise 9.1 in Part I, page 53) that W_o is the solution to the Lyapunov equation

$$A^T W_o + W_o A + C^T C = 0 \tag{15.5}$$

Using the fact that $\operatorname{trace} MN = \operatorname{trace} NM$ for two matrices M and N with compatible dimensions, we can rewrite (15.1) as

$$\|G(s)\|_2^2 = \operatorname{trace} \int_0^\infty C e^{At} B B^T e^{A^T t} C^T \, dt$$

and obtain as an alternative expression for the value of the H_2 norm

$$\|G(s)\|_2 = \sqrt{\operatorname{trace} C W_c C^T} \tag{15.6}$$

where

$$W_c = \int_0^\infty e^{At} B B^T e^{A^T t} \, dt \tag{15.7}$$

is the solution to

$$A W_c + W_c A^T + B B^T = 0 \tag{15.8}$$

Thus, the H_2 norm can be computed by solving a single Lyapunov equation. Note that equations (15.5) and (15.8) are linear in the elements of W_o and W_c, respectively.

The matrices W_c and W_o are called the *Controllability Gramian* and the *Observability Gramian*, respectively. Their singular value decompositions reveal important information about the properties of a system, and we will study Gramians in more detail later in the context of model order reduction. The terms controllability Gramian and observability Gramian are justified by the fact that

- a system (A, B) is controllable if and only if W_c is positive definite;

- a system (A, C) is observable if and only if W_o is positive definite.

Computing the H_∞ norm

The H_∞ norm is defined for systems with stable, proper transfer functions - there is no need to assume $D = 0$. However, to simplify the derivation of the result given below, we will consider strictly proper systems first.

Because the H_∞ norm of $G(s)$ is the maximum of $\bar{\sigma}(G(j\omega))$ over frequency, one could try to compute $\bar{\sigma}(G(j\omega))$ for many values of ω and then search for the maximum. A more efficient method that avoids the problem of determining the range and spacing of frequencies to be checked, is to check the imaginary eigenvalues of a certain Hamiltonian matrix.

The method for computing the H_∞ norm described below is an iterative procedure - in each step we can only check whether $\|G(s)\|_\infty$ is less than a given positive constant γ.

Consider a stable plant with transfer function $G(s) = C(sI - A)^{-1}B$. For a given $\gamma > 0$, define the Hamiltonian matrix

$$M_\gamma = \begin{bmatrix} A & \frac{1}{\gamma}BB^T \\ -\frac{1}{\gamma}C^TC & -A^T \end{bmatrix}$$

The key for computing $\|G(s)\|_\infty$ is the following Theorem.

Theorem 15.1   *Given a positive constant $\gamma > 0$, the following two statements are equivalent*

i)   *γ is a singular value of $G(j\omega_0)$ at some frequency ω_0*

ii)   *M_γ has at least one eigenvalue on the imaginary axis.*

This Theorem allows the following strategy for computing $\|G(s)\|_\infty$. We want to find the largest singular value over all frequencies, and we can start with a sufficiently large value of γ and use the above Theorem to check whether it is a singular value of $G(j\omega)$ at some frequency. Note that we do not need to know at which frequency. If M_γ has no imaginary eigenvalue, γ was too large and we try with a smaller value of γ. If M_γ does have an imaginary eigenvalue, γ was too small and we try with a larger value. A bisection method can be used as an efficient way of finding a value of γ that is equal to $\|G(s)\|_\infty$ within a guaranteed accuracy.

Proof of Theorem 15.1

We first show (i) $\Rightarrow$ (ii). Assume that γ is a singular value of $G(j\omega_0)$; to simplify notation we write $G_0 = G(j\omega_0)$. Let $G_0 = U\Sigma V^H$ be the singular value decomposition of G_0. Then, from $G_0V = U\Sigma$ and $G_0^H U = V\Sigma^T$, there exist nonzero vectors u and v such that

$$G_0 v = \gamma u, \qquad G_0^H u = \gamma v$$

Substituting $G(s) = C(sI - A)^{-1}B$ at $s = j\omega_0$ for G_0 yields

$$C(j\omega_0 I - A)^{-1}Bv = \gamma u, \qquad \text{and} \qquad B^T(-j\omega_0 I - A^T)^{-1}C^T u = \gamma v \qquad (15.9)$$

Introducing the vectors

$$p = (j\omega_0 I - A)^{-1}Bv \qquad \text{and} \qquad q = (-j\omega_0 I - A^T)^{-1}C^T u \qquad (15.10)$$

this becomes

$$Cp = \gamma u \qquad \text{and} \qquad B^T q = \gamma v$$

or

$$\begin{bmatrix} C & 0 \\ 0 & B^T \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} 0 & \gamma I \\ \gamma I & 0 \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix}$$

Solving for v and u yields

$$\begin{bmatrix} v \\ u \end{bmatrix} = \begin{bmatrix} 0 & \gamma I \\ \gamma I & 0 \end{bmatrix}^{-1} \begin{bmatrix} C & 0 \\ 0 & B^T \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} \tag{15.11}$$

Note that (15.11) guarantees that

$$\begin{bmatrix} p \\ q \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

From (15.10), the vectors p and q satisfy

$$(j\omega_0 I - A)p = Bv \qquad \text{and} \qquad (-j\omega_0 I - A^T)q = C^T u$$

or

$$\begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} + \begin{bmatrix} B & 0 \\ 0 & -C^T \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix} = j\omega_0 \begin{bmatrix} p \\ q \end{bmatrix}$$

Substituting from (15.11) yields

$$\left(\begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix} + \begin{bmatrix} B & 0 \\ 0 & -C^T \end{bmatrix} \begin{bmatrix} 0 & \gamma I \\ \gamma I & 0 \end{bmatrix}^{-1} \begin{bmatrix} C & 0 \\ 0 & B^T \end{bmatrix} \right) \begin{bmatrix} p \\ q \end{bmatrix} = j\omega_0 \begin{bmatrix} p \\ q \end{bmatrix} \tag{15.12}$$

It is straightforward to verify that the matrix expression in parentheses is M_γ and thus

$$M_\gamma \begin{bmatrix} p \\ q \end{bmatrix} = j\omega_0 \begin{bmatrix} p \\ q \end{bmatrix}$$

which shows that $j\omega_0$ is an eigenvalue of M_γ. This proves (i) $\Rightarrow$ (ii).

To prove (ii) $\Rightarrow$ (i), assume that $j\omega_0$ is an eigenvalue of M_γ. Then there exists a nonzero vector $\begin{bmatrix} p^T & q^T \end{bmatrix}^T$ that satisfies (15.12). Now use (15.11) to define $[v^T \ u^T]^T \neq 0$. Then from (15.11) and (15.12) it follows that γ is a singular value of $G(j\omega_0)$. This completes the proof. ∎

If $D \neq 0$, the associated Hamiltonian matrix to check is

$$M_\gamma = \begin{bmatrix} A - BR^{-1}D^T C & -\gamma BR^{-1}B^T \\ -\gamma C^T S^{-1} C & -(A - BR^{-1}D^T C)^T \end{bmatrix}$$

where R and S are given by

$$R = D^T D - \gamma^2 I \qquad \text{and} \qquad S = DD^T - \gamma^2 I$$

H_2 and H_∞ Spaces

Why are the two system norms introduced in the previous chapter called the H_2 norm and the H_∞ norm? The H in these norms refers to the British mathematician G.H. Hardy, after whom a certain type of vector spaces - the *Hardy spaces* - are named. The space of strictly proper, stable transfer functions is the Hardy space H_2, and the space of proper, stable transfer functions is the Hardy space H_∞.

Exercises

Problem 15.1

Use the MATLAB functions `norm(T,2)` and `norm(T,inf)` to compute the H_2 norm and the H_∞ norm of the closed-loop system T in Exercise 11.1. Compare the MATLAB code used in these functions with the results presented in this chapter.

Problem 15.2

For the aircraft model from Example 11.1 design an LQG controller using the following set of weighting matrices

$$A_p = A; \quad B_u = B; \quad C_v = -C; \quad D_{vu} = 0; \quad D_{zw} = 0;$$

Design 1:

$$C_z = 2 \cdot \begin{bmatrix} C \\ 0 \end{bmatrix}; \quad D_{zu} = \begin{bmatrix} 0 \\ I_3 \end{bmatrix}; \quad B_w = 2 \cdot \begin{bmatrix} B & 0 \end{bmatrix}; \quad D_{vw} = \begin{bmatrix} 0 & I_3 \end{bmatrix}$$

Design 2:

$$C_z = 50 \cdot \begin{bmatrix} C \\ 0 \end{bmatrix}; \quad D_{zu} = \begin{bmatrix} 0 \\ I_3 \end{bmatrix}; \quad B_w = 50 \cdot \begin{bmatrix} B & 0 \end{bmatrix}; \quad D_{vw} = \begin{bmatrix} 0 & I_3 \end{bmatrix}$$

- Plot the singular values of the closed-loop system with both controllers. (Use the MATLAB command `sigma`).

- What do the singular values at low frequencies tell you about the system tracking capability?

- Give some comments on the noise rejection capabilities of these systems. (Look at the singular values at high frequencies)

Problem 15.3

For the closed loop system of Problem 15.2 (design 2)

a) Calculate the H_2 norm of the closed loop system, using the controllability and observability Gramian.
   Hint: Use the MATLAB function `gram`.

b) Estimate the H_2 norm of the closed loop system $T(s)$ using

$$\|T(s)\|_2 = \left(\int_0^{T_f} \|g(t)\|_F^2 dt \right)^{1/2}$$

   where $g(t)$ is the impulse response matrix of the system and T_f is a sufficiently large value.
   Hint: Use the MATLAB function `impulse`, *and use the trapezoidal approximation to integrate.*

c) Calculate the H_2 norm of the closed loop system, using the MATLAB function `norm`. Compare with previous results.

Problem 15.4

For the closed loop system of Problem 15.2 (design 2)

a) Calculate the H_∞ norm of the closed loop system by iterating on γ, using Theorem 15.1.

b) Read the H_∞ norm of the closed loop system from the singular value plots obtained in Problem 15.2.

c) Calculate the H_∞ norm of the closed loop system, using the MATLAB function `norm`. Compare with previous results.

Chapter 16

Design Objectives and Sensitivity Functions

In this chapter we will discuss how the H_∞ norm can be used to express design specifications on control systems. For this purpose, we will first introduce the *sensitivity function* and the *complementary sensitivity function* of a feedback system.
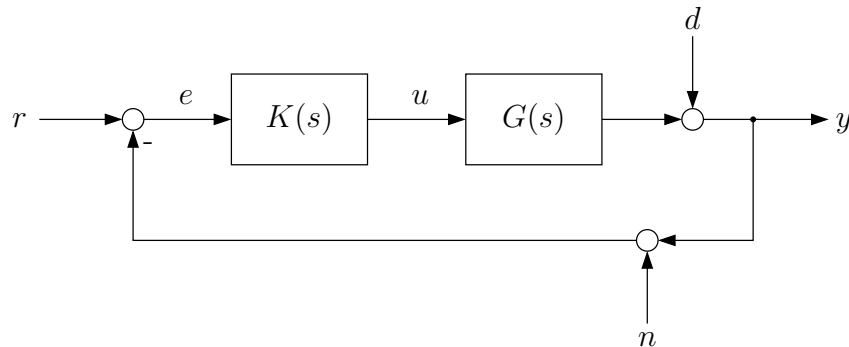
Figure 16.1: Closed-loop system

Consider the closed-loop system shown in Figure 16.1. External inputs are a reference input $r(t)$, an output disturbance $d(t)$ and measurement noise $n(t)$. The controlled output is $y(t)$, and we have

$$y = Gu + d$$

Substituting

$$u = K(r - y - n)$$

yields

$$y = GK(r - y - n) + d \qquad \text{or} \qquad (I + GK)y = GK(r - n) + d$$

so the controlled output is

$$y = (I + GK)^{-1}GK(r - n) + (I + GK)^{-1}d \tag{16.1}$$

Now define the transfer functions

$$S(s) = (I + G(s)K(s))^{-1} \qquad \text{and} \qquad T(s) = (I + G(s)K(s))^{-1}G(s)K(s)$$

The function $S(s)$ is called the *sensitivity function* of the feedback system. From (16.1), it is the transfer function from the output disturbance d to the controlled output y. It is easy to check that the functions $S(s)$ and $T(s)$ are related by

$$S + T = I \tag{16.2}$$

For this reason, the function $T(s)$ is called the *complementary sensitivity* of the system. From (16.1), it is the closed-loop transfer function from r to y, and it is also the transfer function from measurement noise to controlled output. In terms of S and T, we have

$$y = T(r - n) + Sd \tag{16.3}$$

There is a further interpretation of the sensitivity function. The control error $e(t)$ is

$$e = r - n - y = r - n - (GKe + d)$$

Solving for e yields

$$e = S(r - n - d) \tag{16.4}$$

which shows that the sensitivity S is also the transfer function from reference input, measurement noise and output disturbance, to the control error.

For the control system in Figure 16.1, we consider the following design objectives

- tracking: the controlled output should track the reference input, ideally $y(t) = r(t)$

- disturbance rejection: the controller should keep the controlled output at its desired value in the presence of a disturbance $d(t) \neq 0$. Ideally the transfer function from d to y would be zero.

- noise rejection: the controller should suppress the effect of measurement noise on the control input

- reasonable control effort: the above design objectives must be achieved within given constraints on the actuators, which means the magnitude of the control input must not exceed given limits.

In terms of sensitivity and complementary sensitivity, it is clear from (16.3) that perfect tracking and disturbance rejection require $T(s) = I$. From (16.2), this implies $S(s) = 0$, which is also consistent with (16.4) - the control error should be zero.

Turning to the third design objective, (16.3) shows that perfect noise rejection requires $T = 0$ (and consequently $S = I$). Clearly, tracking and disturbance rejection on one

hand and the suppression of measurement noise on the other hand are conflicting design objectives.

The control effort is related to the sensitivity because

$$u = Ke = KS(r - n - d) \tag{16.5}$$

To achieve the design objectives with reasonable control effort, the transfer function $K(s)S(s)$ should not become "too large". The function KS is called the *control sensitivity.*

The discussion so far showed that tracking and noise rejection are conflicting design objectives. On the other hand, it is clear that for physical reasons alone perfect tracking is not possible for signals at arbitrary high frequencies. One important decision that has to be made when designing a control system is to determine the closed-loop bandwidth. The bandwidth is the frequency range in which control is effective, i.e. where the reference input is tracked and disturbances are rejected. In practice, the design objectives are interpreted such that good tracking and disturbance rejection properties should be achieved for signals within the closed-loop bandwidth, and noise rejection for high frequency measurement noise beyond the bandwidth.

Mixed Sensitivity Design for SISO Systems

In the preceding discussion and the notation used so far, we assumed that the plant is a multi-input multi-output system. We will discuss how the design objectives can be expressed and achieved by constraints on S, T and KS for single-input single-output systems first. Figure 16.2 shows a typical shape of the magnitude of $S(s)$ and $T(s)$ for the control system in Figure 16.1 when the plant is a SISO system. At low frequencies, the sensitivity S is close to zero, which means good tracking and disturbance rejection. At higher frequencies, the magnitude of S increases, and at the frequency $\omega = \omega_b$, it is 0.71 (-3dB). We will consider *control to be effective* if $|S| < 0.71$, and define the closed-loop bandwidth as this frequency.

The complementary sensitivity T at low frequencies is approximately 1, which is consistent with $|S| \approx 0$ and indicates good tracking properties, because T is the transfer function from r to y. At high frequencies, $|T|$ rolls off and approaches zero. This is required for the rejection of high frequency measurement noise, and implies that $|S|$ approaches 1. An alternative way of defining the closed-loop bandwidth is to take it as the frequency ω_{bT} where $|T| = 0.71$. Figure 16.2 illustrates that the two frequencies ω_b and ω_{bT} are usually not equal, but also not far from each other.
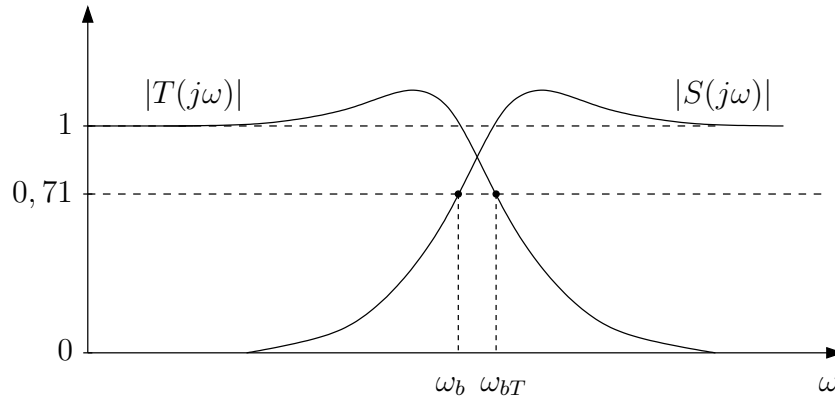
Figure 16.2: Sensitivity and complementary sensitivity of a SISO system

The "Waterbed Effect"

In Figure 16.2 both $|S|$ and $|T|$ are shown to have peak values greater than 1. This happens often and is in many cases unavoidable. A peak of $|T|$ greater than 1 indicates a resonant peak with overshoot in the step response, and $|S| > 1$ means that disturbances in this frequency range are not suppressed but actually amplified. These peaks are unavoidable if

- the pole excess of the plant is greater than one, or

- if the plant has one or more zeros in the right half plane

If the plant is minimum-phase and if the pole excess of the plant is greater than one, one can show that

$$\int_0^\infty \ln |S(j\omega)| \, d\omega = 0$$

A consequence of this is that the area below 1 and above $|S(j\omega)|$ is equal to the area above 1 and below $|S(j\omega)|$, when $|S(j\omega)|$ is plotted on a logarithmic scale. In other words, if we push $|S(j\omega)|$ down at low frequencies, it must pop up somewhere else (the "waterbed effect") - usually at high frequencies.

We will not prove this fact here but try to make it plausible with the help of a Nyquist plot. Consider Figure 16.3 where the Nyquist plot of the loop transfer function $L(s) = G(s)K(s)$ is shown for the control system in Figure 16.1 with a SISO plant. If $L(s)$ has a pole excess of 2, then the Nyquist plot of L will penetrate a disc of radius 1 around the critical point -1. Since the distance between $L(j\omega)$ and -1 is $|1 + L(j\omega)|$, we have inside the disc

$$|1 + L(j\omega)| < 1 \qquad \text{and therefore} \qquad |S(j\omega)| > 1$$

If the plant has right half plane zeros, the resulting phase lag will lead to $L(j\omega)$ penetrating the disc even if the pole excess is only 1.
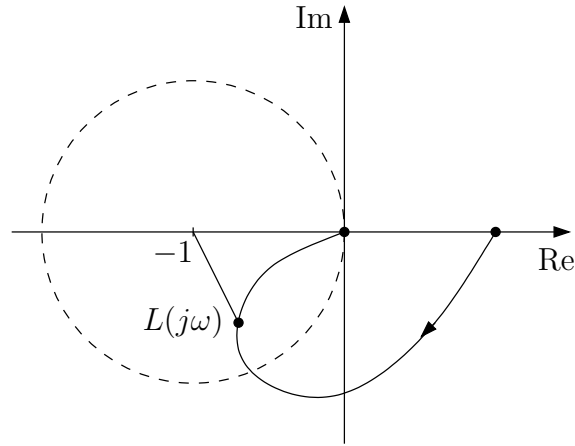
Figure 16.3: Penetration of the -1 disc by the Nyquist plot of $L(s)$

Low Frequency Design - Shaping the Sensitivity

In chapter 19 we will study methods for designing a controller such that the H_∞ norm of the closed-loop transfer function is less than a specified value. Here we will show how the H_∞ norm can be used to express constraints on the sensitivity that are required to achieve the design objectives. Properties of the closed-loop system that depend on the shape of $|S(j\omega)|$ include

- the bandwidth ω_b

- peak overshoot

- the system type (capability of tracking step or ramp inputs with zero steady state error)

To see how the sensitivity is related to the system type, consider Figure 16.4. The magnitude response of a type 1 system - a system with integral action in the loop - has at low frequencies a slope of -20 dB/dec. Because at low frequencies ($\omega \ll \omega_b$) we have $|L(j\omega)| \gg 1$, we can approximate the sensitivity by

$$S(j\omega) = \frac{1}{1 + L(j\omega)} \approx \frac{1}{L(j\omega)}$$

Thus, at low frequencies the graph of $|S(j\omega)|$ in dB is approximately equal to the mirror image of $|L(j\omega)|$ about the 0 dB line. In other words, a feedback system has integral action in the loop if the magnitude of the sensitivity has at low frequencies a positive slope of 20 dB/dec. Similarly, for a type 2 system we need $|S(j\omega)|$ to have a slope of 40 dB/dec etc.
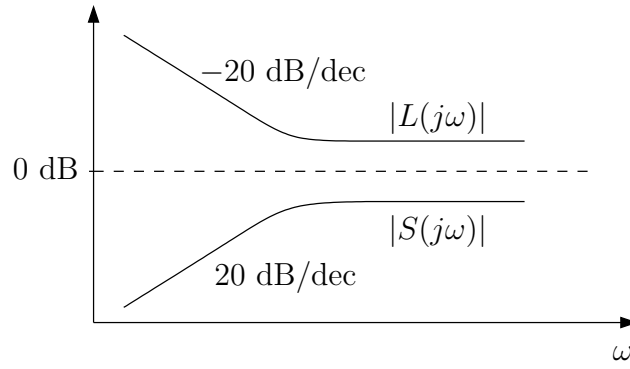
Figure 16.4: Relation between loop gain and sensitivity at low frequencies

We have seen that we need to keep $|S(j\omega)|$ small at low frequencies. One could use the H_∞ norm to express constraints on the magnitude of the sensitivity, for example by imposing

$$\|S(s)\|_\infty < M_S$$

for some small constant $M_S > 0$. Because the H_∞ norm is the peak value of the magnitude of $S(s)$, this would enforce

$$|S(j\omega)| < M_S \quad \forall \omega$$

Such a constraint would be unrealistic however, since we know that we cannot keep the sensitivity small at all frequencies. To express a constraint on a selected frequency range, one can instead introduce a weighting function $W_S(s)$ and impose

$$\|W_S(s)S(s)\|_\infty < 1$$

This enforces

$$|W_S(j\omega)S(j\omega)| < 1 \quad \forall \omega$$

or equivalently

$$|S(j\omega)| < \frac{1}{|W_S(j\omega)|} \quad \forall \omega$$

This shows that one can use the weighting function $W_S(s)$ - together with a constraint on the H_∞ norm - to shape the magnitude of $S(j\omega)$. If the H_∞ norm is less than 1, the inverse of the weighting function is an upper bound on the sensitivity. A typical shape of $1/|W_S(j\omega)|$ is shown in Figure 16.5 - the weighting function enforces integral action and an upper bound M_S on the peak value of $|S(j\omega)|$.
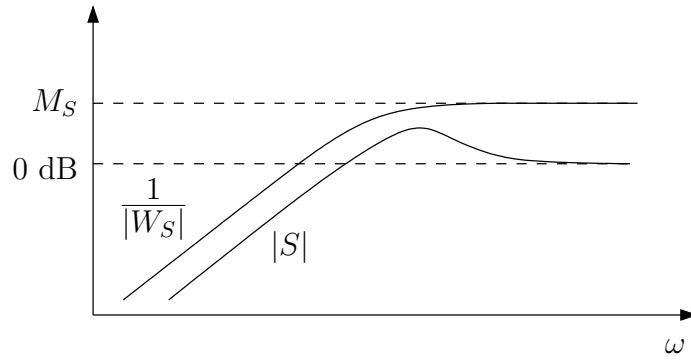
Figure 16.5: Shaping the sensitivity with a weighting function $W_S(s)$

High Frequency Design - Shaping $|T(s)|$ and $|K(s)S(s)|$

At high frequencies ($\omega \gg \omega_b$) the complementary sensitivity is required to roll off in order to suppress measurement noise. This requirement can be expressed by introducing a weighting function $W_T(s)$ for the complementary sensitivity and imposing

$$\|W_T(s)T(s)\|_\infty < 1$$

The function $W_T(s)$ can be used to shape $|T(j\omega)|$ just as $W_S(s)$ can be used to shape $|S(j\omega)|$.

It was mentioned earlier that in order to maintain a reasonable control effort, the control sensitivity $K(s)S(s)$ should not become "too large". Now we can be more specific and impose

$$\|W_K(s)K(s)S(s)\|_\infty < 1$$

where a further weighting function $W_K(s)$ has been introduced. This function can be chosen to impose limits on the control effort in a selected frequency range - typically at high frequencies.

Weighting Filters and Generalized Plant

The constraints on sensitivity, complementary sensitivity and control sensitivity discussed so far can be interpreted in the form of the block diagram in Figure 16.6. In this block diagram, the functions S, T and KS represent the following transfer functions:

- sensitivity S: $\quad r \to e$

- complementary sensitivity T: $\quad r \to y$

- control sensitivity KS: $\quad r \to u \quad$ (because $e = Sr$ and $u = Ke$)
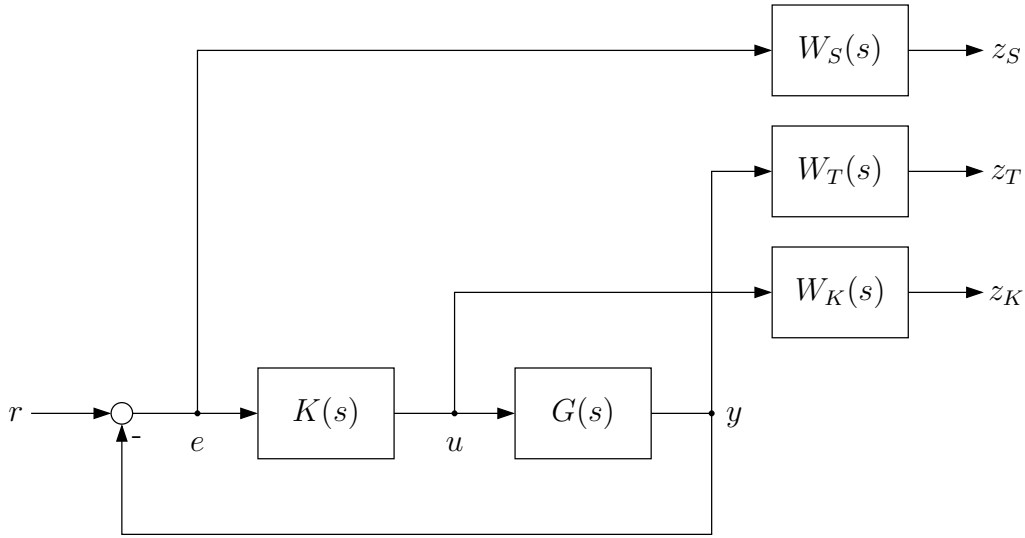
Figure 16.6: Constraints on S, T and KS as weighting filters

The weighting functions are here used as filters to generate fictitious output signals z_S, z_T and z_K. This is done in a way such that the weighted sensitivity functions - on which a H_∞ constraint is to be imposed - can now be seen as the transfer functions from r to the fictitious signals

- weighted sensitivity $W_S S$: $r \to z_S$

- weighted complementary sensitivity $W_T T$: $r \to z_T$

- weighted control sensitivity $W_K KS$: $r \to z_K$

In Figure 16.7, the block diagram has been redrawn to bring it into the form of the generalized plant shown in Figure 11.1.

Referring to this generalized plant, each of the three constraints

$$\|W_S(s)S(s)\|_\infty < 1 \tag{16.6}$$
$$\|W_T(s)T(s)\|_\infty < 1 \tag{16.7}$$
$$\|W_K(s)K(s)S(s)\|_\infty < 1 \tag{16.8}$$

can be expressed as a constraint on the H_∞ norm of the closed-loop transfer function from the external input r to one of the outputs z_S, z_T or z_K.

Of these three constraints, the first one is active at low frequencies, while the second and third are active at high frequencies. Usually the low frequency constraint is combined with one of the high frequency constraints to shape the overall loop transfer function. One can approximate such a combined constraint by considering the H_∞ norm of the transfer function from r to an output vector z that contains two (or possibly three)
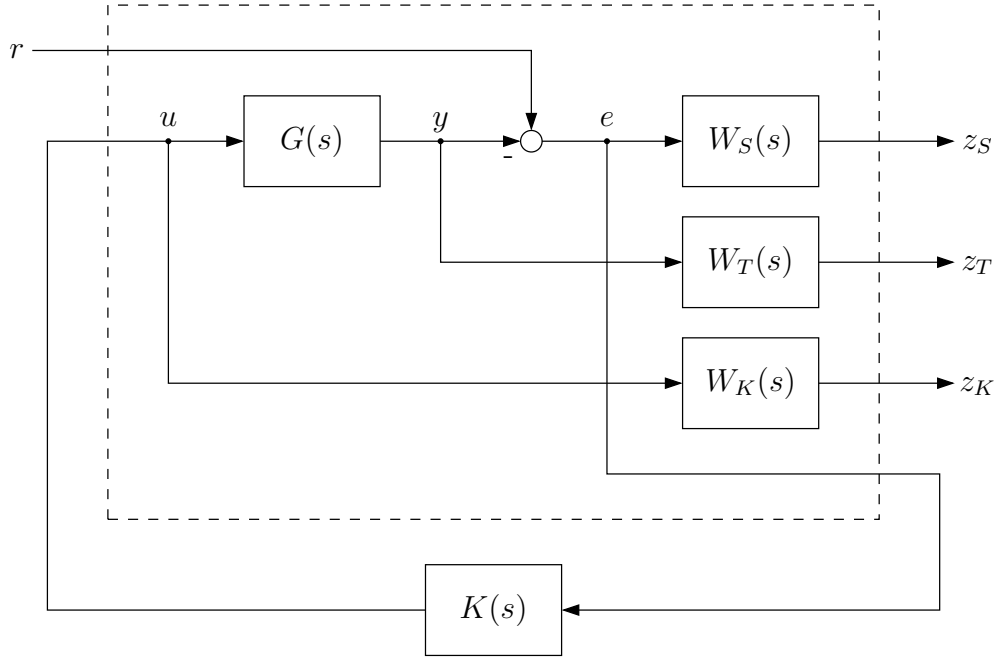
Figure 16.7: Constraints on S, T and KS in terms of a generalized plant

output signals. As an illustration, assume we want to design a controller with integral action and given actuator constraints. We choose z_S and z_K as outputs and (ignoring z_T in this case) define the fictitious output vector z of the generalized plant as

$$z = \begin{bmatrix} z_S \\ z_K \end{bmatrix}$$

Let $T_{zr}(s)$ denote the closed-loop transfer function from r to z such that $z = T_{zr}r$. From Figure 16.7 we have

$$\begin{bmatrix} z_S \\ z_K \end{bmatrix} = \begin{bmatrix} W_S S \\ W_K KS \end{bmatrix} r, \qquad \text{therefore} \qquad T_{zr}(s) = \begin{bmatrix} W_S(s)S(s) \\ W_K(s)K(s)S(s) \end{bmatrix}$$

When $W_S(s)$ and $W_K(s)$ have been chosen to express the design specifications, the task is to find a controller $K(s)$ such that the closed-loop transfer function satisfies $\|T_{zr}(s)\|_\infty < 1$ or

$$\left\| \begin{bmatrix} W_S S \\ W_K KS \end{bmatrix} \right\|_\infty < 1 \tag{16.9}$$

We will discuss in chapter 19 how a controller that satisfies this condition can be computed. Here we assume that a controller has been found that satisfies (16.9). From the definition of the H_∞ norm, (16.9) is equivalent to

$$\sup_\omega \bar{\sigma} \left(\begin{bmatrix} W_S(j\omega)S(j\omega) \\ W_K(j\omega)K(j\omega)S(j\omega) \end{bmatrix} \right) < 1$$

We are still considering SISO systems, and at a given frequency, we are now looking for the maximum singular value of a two-dimensional complex vector. For a complex vector $x = [x_1 \ \ x_2]^T$ we have

$$\bar{\sigma}(x) = \sqrt{\lambda_{\max}(x^H x)} = \sqrt{x^H x} = \sqrt{|x_1|^2 + |x_2|^2} = \|x\|$$

The condition (16.9) is therefore

$$\sup_\omega \sqrt{|W_S S|^2 + |W_K KS|^2} < 1 \tag{16.10}$$

If $|W_S S| \ll 1$, this is equivalent to

$$\sup_\omega |W_K KS| < 1$$

On the other hand, if $|W_K KS| \ll 1$, (16.10) is the same as

$$\sup_\omega |W_S S| < 1$$

If either of the two terms $|W_S S|$ or $|W_K KS|$ is small, (16.9) is a close approximation of the H_∞ constraint on the remaining term. The largest approximation error occurs if $|W_S S| = |W_K KS|$, in this case the approximation of (16.6) and (16.8) by (16.9) becomes

$$\sup_\omega |W_S S| < 0.71 \qquad \text{and} \qquad \sup_\omega |W_K KS| < 0.71$$

respectively. Thus, the largest approximation error is -3dB - which means that (16.9) is a reasonable approximation of the constraints (16.6) and (16.8) on S and KS.

Mixed Sensitivity Design for MIMO Systems

For SISO plants, we used the fact that the condition

$$\|W_S(s)S(s)\|_\infty < 1 \tag{16.11}$$

is equivalent to

$$|S(j\omega)| < \frac{1}{|W_S(j\omega)|} \quad \forall \omega \tag{16.12}$$

For MIMO plants, the sensitivity $S(s)$ is a l by l transfer function matrix (where l is the number of plant outputs). Accordingly, we need a l by l matrix $W_S(s)$ of shaping filters. Usually $W_S(s)$ is chosen diagonal, and we will begin with the choice

$$W_S(s) = \begin{bmatrix} w_S(s) & & 0 \\ & \ddots & \\ 0 & & w_S(s) \end{bmatrix}$$

where the same scalar shaping filter $w_S(s)$ is used for each output of $S(s)$. With this choice, (16.11) can be written as

$$\sup_{\omega} \bar{\sigma}(W_S(j\omega)S(j\omega)) = \sup_{\omega} \bar{\sigma}(w_S(j\omega)S(j\omega)) = \sup_{\omega} |w_S(j\omega)|\, \bar{\sigma}(S(j\omega)) < 1$$

This is equivalent to

$$\bar{\sigma}(S(j\omega)) < \frac{1}{|w_S(j\omega)|} \quad \forall\omega$$

While for a SISO plant the shaping filter acts as the upper bound in (16.12) on the magnitude of the sensitivity, it acts as an upper bound on the *maximum singular value* of the sensitivity for a MIMO plant. Figure 16.8 shows a typical pattern of the maximum and minimum singular value plots of $S(j\omega)$ and $T(j\omega)$.

Similarly, the constraints $\|W_T T\|_\infty < 1$ and $\|W_K K S\|_\infty < 1$ on complementary sensitivity and control sensitivity are equivalent to

$$\bar{\sigma}(T(j\omega)) < \frac{1}{|w_T(j\omega)|} \quad \forall\omega \qquad \text{and} \qquad \bar{\sigma}(K(j\omega)S(j\omega)) < \frac{1}{|w_K(j\omega)|} \quad \forall\omega$$

where $w_T(s)$ and $w_K(s)$ are scalar shaping filters.
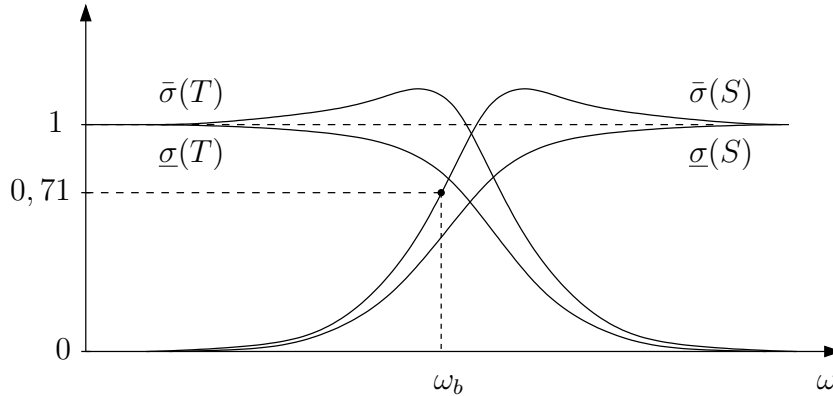
Figure 16.8: Sensitivity and complementary sensitivity of a MIMO system

High Gain and Low Gain Directions

At a given frequency ω_0, the maximum and minimum singular values $\bar{\sigma}(S(j\omega_0))$ and $\underline{\sigma}(S(j\omega_0))$ represent the maximum and minimum gain of the transfer function $S(s)$ over all input directions - which are the maximum and minimum gains from a vector d of sinusoidal disturbances to the output vector y. The singular value decomposition of $S(j\omega_0)$ provides not only the gains but also the input directions associated with the largest and smallest gain. From $S(j\omega_0) = U\Sigma V^H$ follows that $SV = U\Sigma$ and

$$S(j\omega_0)v_1 = \sigma_1 u_1$$

Here v_1 and u_1 are the first columns of V and U, respectively. Assuming that the singular values are ordered by magnitude, we have $\bar{\sigma} = \sigma_1$. The input direction associated with the largest gain is therefore v_1, and the corresponding output direction is u_1. Similarly, the input direction associated with the lowest gain is the column of V corresponding to the smallest singular value.

Exercises

Problem 16.1

What is the largest approximation error when three weighting filters are used to express constraints on S, T and KS for a SISO system, and the controller is designed to satisfy

$$\left\| \begin{bmatrix} W_S S \\ W_T T \\ W_K KS \end{bmatrix} \right\|_\infty < 1$$

Problem 16.2

Confirm the information about the steady state tracking capabilities of the LQG control loop for the aircraft model obtained in Exercise 14.1: plot the singular values of the closed-loop sensitivity over frequency. Use a singular value decomposition of the sensitivity at low frequencies to find the "best" and the "worst" input directions.

Problem 16.3

Consider the following open loop transfer functions and the closed loop system in Figure 16.9, where $G(s)$ is either

$$G_1(s) = \frac{2}{s(s+1)} \quad \text{or} \quad G_2(s) = \frac{2-s}{s(s+2)}$$
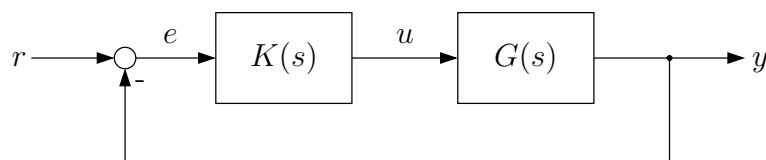
Figure 16.9: Closed-loop system

a)   For $G_1(s)$ plot the sensitivity function $|S|$ for $K(s) = 0.1$, $K(s) = 1.0$, $K(s) = 10$ and compare the bandwidth and the peak value of $|S|$.

b) Plot the closed-loop step response of $G_1(s)$ with $K(s) = 0.1$, $K(s) = 1.0$, $K(s) = 10$ and compare the maximum overshoot and the rise time.

c) For $G_2(s)$ plot the sensitivity function $|S|$ for $K(s) = 0.1$, $K(s) = 0.5$, $K(s) = 1.5$ and compare the bandwidth and the peak value of $|S|$.

Chapter 17

Mixed Sensitivity - A Design Example

In this chapter we will design a multivariable controller for the aircraft model in example 11.1 that has been used in Exercises 11.1, 14.1, 15.1 and 16.2. We will use software tools for designing controllers that satisfy constraints on the H_∞ norm of the closed-loop transfer function - *how* these software tools work will be studied in chapter 19. Here we will discuss how the sensitivity functions can be shaped with the help of weighting filters, and how available software tools can be used to obtain a controller that satisfies the constraints. Examples are given to illustrate the effect of different choices for weighting filters. The actual design is left as an exercise.
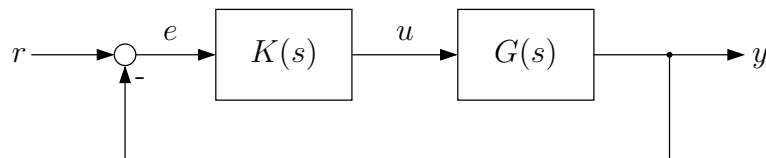
Figure 17.1: Closed-loop system

The closed-loop system is shown in Figure 17.1. Initially we consider the two design objectives

- fast tracking of step changes for all three reference inputs, with little or no overshoot

- control input must satisfy $|u_3| < 20$

Scaling the Weighting Filters

It was shown in the previous chapter how weighting filters for loop shaping can be expressed in the form of the generalized plant in Figure 16.7. In our design we will shape the

sensitivity S and the control sensitivity KS, to achieve desired properties of the closed-loop transfer function $T_{zw}(s)$ from the reference input r to the fictitious output vector $z = [z_S^T \quad z_K^T]^T$. The design will be carried out by choosing suitable weighting filters W_S and W_K, and by imposing

$$\|T_{zw}\|_\infty = \left\| \begin{bmatrix} W_S S \\ W_K KS \end{bmatrix} \right\|_\infty < 1 \tag{17.1}$$

Of course the closed-loop transfer function $T_{zw}(s)$ depends on the controller $K(s)$, and the task is to find a controller that satisfies this constraint. If there are no other constraints on the closed-loop system, one can in fact compute the controller that minimizes $\|T_{zw}(s)\|_\infty$. Let

$$\gamma_o = \min_K \|T_{zw}(s)\|_\infty$$

denote the optimal value of the H_∞ norm. This value may or may not be less than one. If $\gamma_o > 1$, the constraints expressed by the weighting functions W_S and W_K are too strong and a controller that satisfies them does not exist. In this case, the constraints must be relaxed. On the other hand, if $\gamma_o < 1$, controllers that satisfy the constraints do exist and the constraints can actually be strengthened.

A simple way of adjusting the weighting filters is scaling. Assume that the minimum value of $\|T_{zw}(s)\|_\infty$ is $\gamma_o \neq 1$ and that this value is achieved with the optimal controller $K_o(s)$. Introduce the scaled weighting functions

$$\tilde{W}_S(s) = \frac{1}{\gamma_o} W_S(s) \qquad \text{and} \qquad \tilde{W}_K(s) = \frac{1}{\gamma_o} W_K(s)$$

Replace the weighting filters in (17.1) by $\tilde{W}_S$ and $\tilde{W}_K$. The H_∞ norm is minimized by the same controller $K_o(s)$ as before, and we have

$$\left\| \begin{bmatrix} \tilde{W}_S S \\ \tilde{W}_K KS \end{bmatrix} \right\|_\infty = 1$$

This is illustrated in Design 1 below.

Choice of Weighting Functions - Sensitivity

We begin the design with scalar weighting filters $w_S(s)$ and $w_K(s)$, as discussed in the previous chapter. For the loop to have integral action, a positive slope of 20 dB/dec of the sensitivity is required at low frequencies. This could be enforced by including a factor $1/s$ in the weigthing function $w_S(s)$. However, it is clear from Figure 16.7 - and also from (17.1) - that the weighting filters are factors of the closed-loop transfer function $T_{zw}(s)$, and if w_S has a pole at the origin then the same is true for T_{zw}. Because the H_∞ norm is only defined for proper, stable transfer functions, the weighting filters must therefore also be proper and stable.
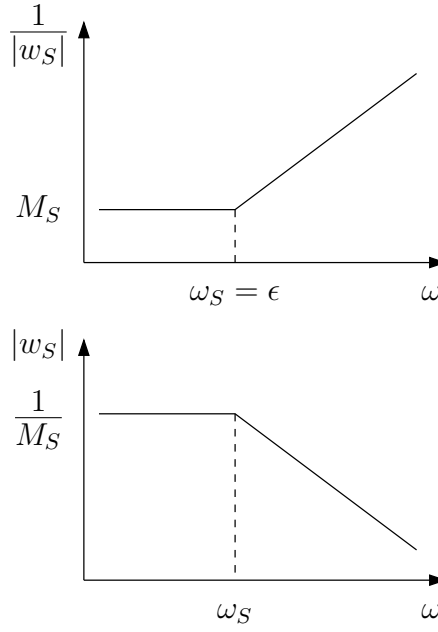
Figure 17.2: Weighting filter $w_S(s)$

To enforce integral action in the loop, one can choose the weighting function to include a factor

$$\frac{1}{s + \epsilon}$$

where $\epsilon > 0$ is a small constant. Figure 17.2 shows the shape of a first order weighting filter and its inverse. M_S is a small constant that is chosen as an upper bound on the sensitivity at low frequencies. The transfer function of the weighting filter is

$$w_S(s) = \frac{\omega_S/M_S}{s + \omega_S} \tag{17.2}$$

where $\omega_S = \epsilon$.

Choice of Weighting Functions - Control Sensitivity

The weighting filter $w_K(s)$ can be used to impose an upper bound on the control sensitivity. Because the control sensitivity should roll off at high frequencies, the inverse of this filter should have low-pass behaviour and thus the filter itself should be a high-pass filter. But, as mentioned before, weighting filters must be stable and proper, and if we start with a zero at ω_K (see Figure 17.3), an additional pole at a frequency well above the bandwidth is required to make $w_K(s)$ proper. In Figure 17.3, this pole is placed at $c\omega_K$, where c is a sufficiently large constant.

From Figure 17.3, the filter transfer function is

$$w_K(s) = \frac{c}{M_K} \frac{s + \omega_K}{s + c\omega_K} \tag{17.3}$$
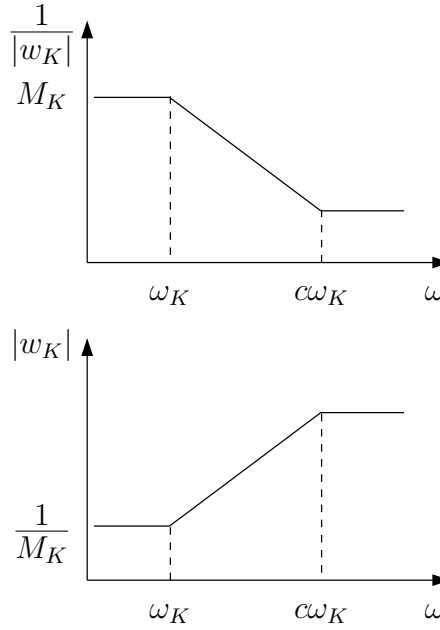
Figure 17.3: Weighting filter $w_K(s)$

Loop Shaping

With the filter transfer functions (17.2) and (17.3), we have made choices concerning the structure and order of the weighting functions W_S and W_K - we use the same scalar, first order filters for all outputs. These choices are restrictive and can be changed if required. They have however the advantage that there is only a small number of design parameters. Initially we fix $c = 10^3$ (placing the pole of w_K three decades above the zero). That leaves us with the design parameters ω_S, M_S, ω_K and M_K.

The parameter ω_S can be used to determine the closed-loop bandwidth, and M_S can be used to push the steady state error towards zero. The shape of w_S according to (17.2) is different from that discussed in the previous chapter (compare Figure 16.5), in so far that we are not imposing an upper bound on $|S|$. The reason for this is that we can use the upper bound M_K on the control sensitivity to impose a limit on a sensitivity peak, as will be seen below. The corner frequency ω_K of W_K should be chosen high enough not to interfere with the bandwidth constraint on $|S|$.

Following these considerations, we start the design procedure with an initial choice of parameters

Design 1:

ω_S	M_S	ω_K	M_K
10^{-4}	10^{-4}	10^2	10^3

The task is now to compute a controller that achieves $\|T_{zw}\|_\infty < 1$, where T_{zw} is a generalized plant of the form shown in Figure 16.7 (without output z_T), with weighting filters W_S and W_K based on (17.2) and (17.3) and parameter values as above. Here we will use the function `hinfsyn()` of the robust control toolbox for MATLAB to find the optimal controller - the controller that minimizes the H_∞ norm of T_{zw}.
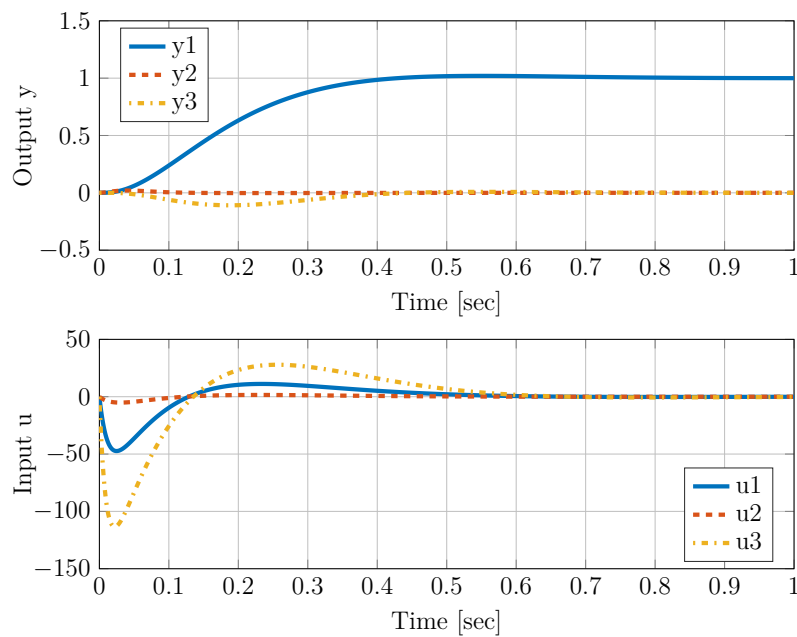
Figure 17.4: Design 1: response to $r(t) = [\sigma(t) \ \ 0 \ \ 0]^T$

Since the aircraft model has three inputs and three outputs, the optimal controller $K_o(s)$ is a three-by-three transfer function matrix. We will later see that the dynamic order of the optimal controller is equal to the dynamic order of the generalized plant. In the present case, the plant is of fifth order, and we have first order weighting filters for each of the three outputs of S and KS. This leaves us with a generalized plant of order 11, and therefore with an 11th order controller.

The response of the closed-loop system in Figure 17.1 with the optimal controller $K_o(s)$ to a reference step input $r(t) = \begin{bmatrix} \sigma(t) & 0 & 0 \end{bmatrix}^T$ is shown in Figure 17.4. The response is fast, but the control input u_3 violates the actuator constraint - this requires some adjustment. First we note that the optimal controller $K_o(s)$ achieves a closed-loop H_∞ norm of less

than one; in fact we have

$$\gamma_o = \min_K \|T_{zw}\|_\infty = 0.1794$$

in this case. Figure 17.5 shows the singular values of the sensitivity together with $1/|W_S|$, and Figure 17.6 shows the control sensitivity together with $1/|W_K|$.

The fact that $\gamma_o < 1$ means that we can tighten the constraints. Introduce the scaled weighting filters

$$\tilde{W}_S(s) = \frac{1}{\gamma_o} W_S(s) \qquad \text{and} \qquad \tilde{W}_K(s) = \frac{1}{\gamma_o} W_K(s)$$

The optimal controller with these new filters is the same as before, but now we obtain $\gamma_o = 1$. The inverse magnitude responses of the scaled filters - together with the sensitivity functions - are plotted in Figures 17.7 and 17.8. They show that at low frequencies the active constraint is $\|W_S S\|_\infty < 1$, and at high frequencies $\|W_K KS\|_\infty < 1$. At frequencies where a constraint is active, the constrained function (S or KS) fits tightly the shape of the filter.

Figure 17.8 shows also that at low frequencies, the control sensitivity KS is constant and slightly greater than one. The corner frequency where - moving from higher to lower frequencies - the magnitude of KS starts to decrease from its enforced peak value, is roughly the frequency where the magnitude of S reaches one. It is the closed-loop bandwidth, which is enforced by the weighting filter W_S. The reason for the constant low-frequency value of $|KS|$ is the near-integral behaviour of the controller $K(s)$. The constraint on $|S|$ leads to a controller with poles very close to the origin in all controller channels. At low frequencies, the controller gain is therefore very large, and the inverse in $KS = K(I + GK)^{-1}$ is dominated by GK. As a result, the maximum singular value of KS at low frequencies is approximately equal to the inverse of the smallest singular value of G. In the present case, the smallest singular value of the aircraft model transfer function at low frequencies is slightly less than one.
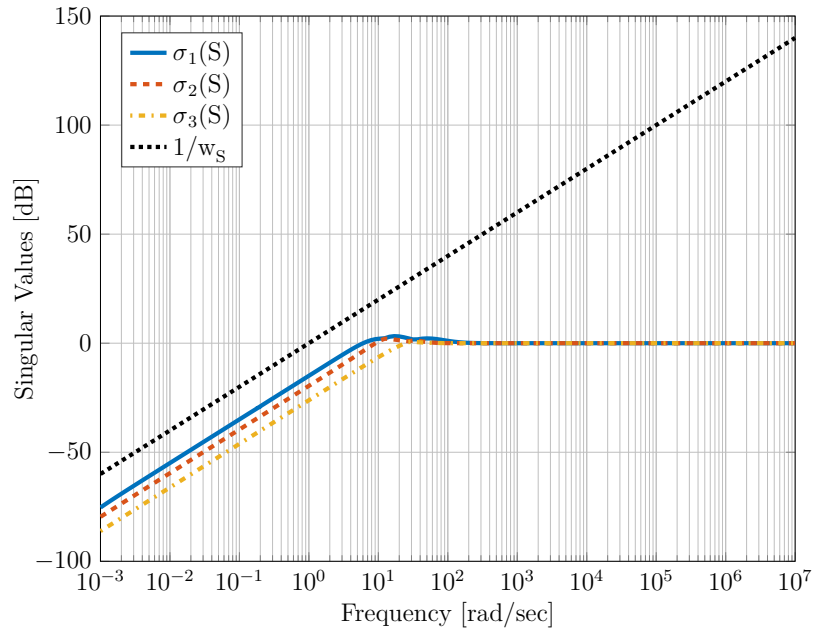
Figure 17.5: Design 1: Sensitivity S and constraint
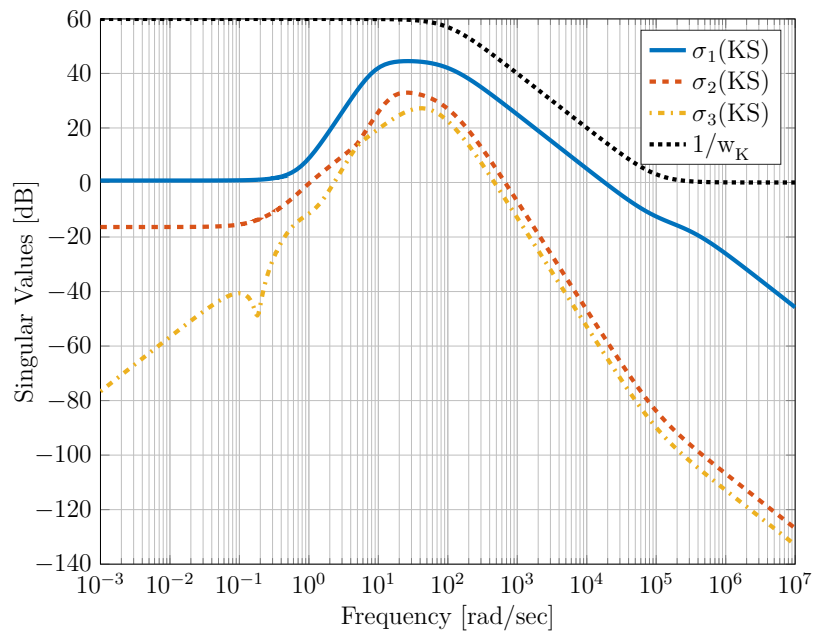
Figure 17.6: Design 1: Control sensitivity KS and constraint
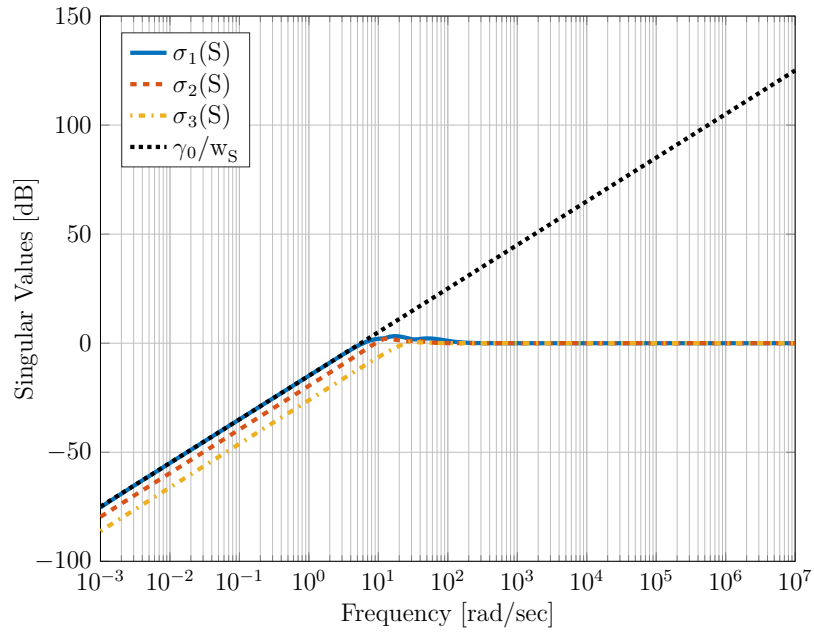
Figure 17.7: Design 1: Sensitivity S and scaled constraint
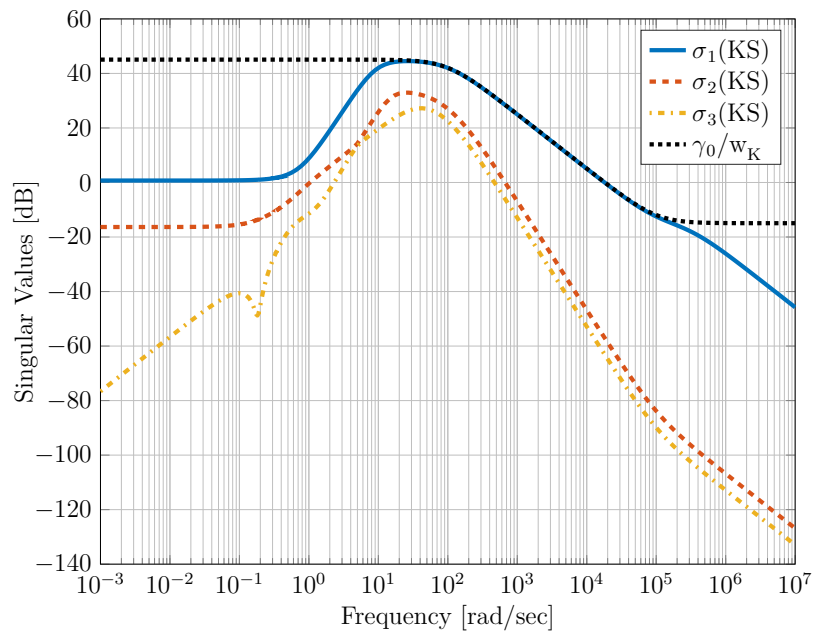
Figure 17.8: Design 1: Control sensitivity KS and scaled constraint

We now return to the problem of re-tuning the controller such that it meets the actuator constraint $|u_3| < 20$. We need to reduce the upper bound M_K on the control sensitivity KS. We try $M_K = 10^2$ instead of 10^3 and leave the other parameters unchanged, so our next attempt is

Design 2:

ω_S	M_S	ω_K	M_K
10^{-4}	10^{-4}	10^2	10^2
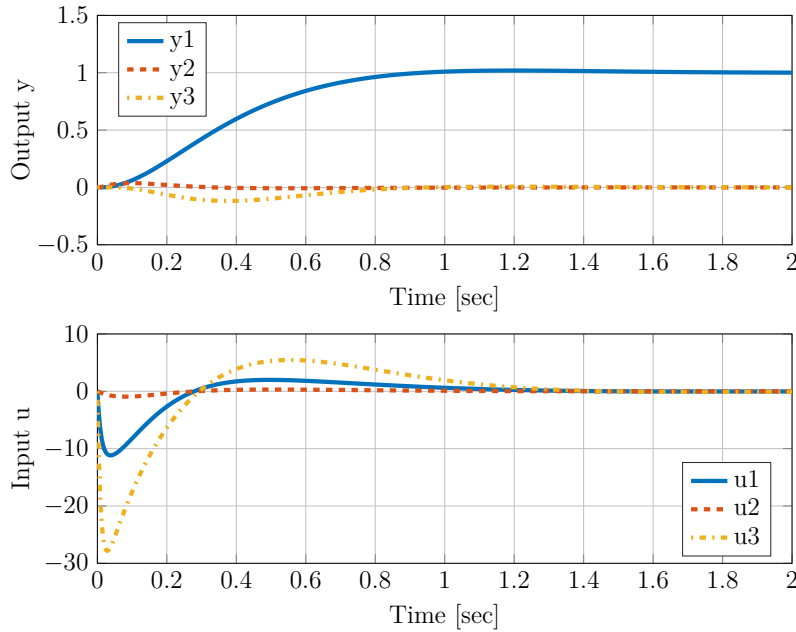
Figure 17.9: Design 2: response to $r(t) = [\sigma(t)\ \ 0\ \ 0]^T$

The step response with the resulting controller is shown in Figure 17.9. Clearly the peak value of the elevator angle u_3 has been significantly reduced, although it is still slightly too large. The sensitivity and control sensitivity - together with the scaled weightings - are shown in Figures 17.10 and 17.11. Notice that the closed-loop bandwidth has been reduced, even though we did not change the weight W_S. This is because the scaled weightings are shown - the H_∞ norm achieved with the unscaled weights is now $\gamma_o = 0.3805$, and the scaled filter $\tilde{W}_S$ represents now a lower bandwidth. Intuitively it should be clear that reducing the control effort (reducing M_K) leads to a slower response.

Because we are minimizing $\|T_{zw}\|_\infty$ rather than rigidly enforcing $\|T_{zw}\|_\infty < 1$, the functions W_S and W_K act as weights - or "soft constraints" - rather than hard constraints. The effect of reducing M_K is illustrated in Figure 17.12. The curves labelled 1 represent scaled constraints (dashed) and sensitivity shapes (solid) of the previous design. The dashed line labelled 2 shows the constraint on KS when M_K is reduced. Because the constraints were scaled for the previous design, the new design will give a controller with $\gamma_o > 1$, which
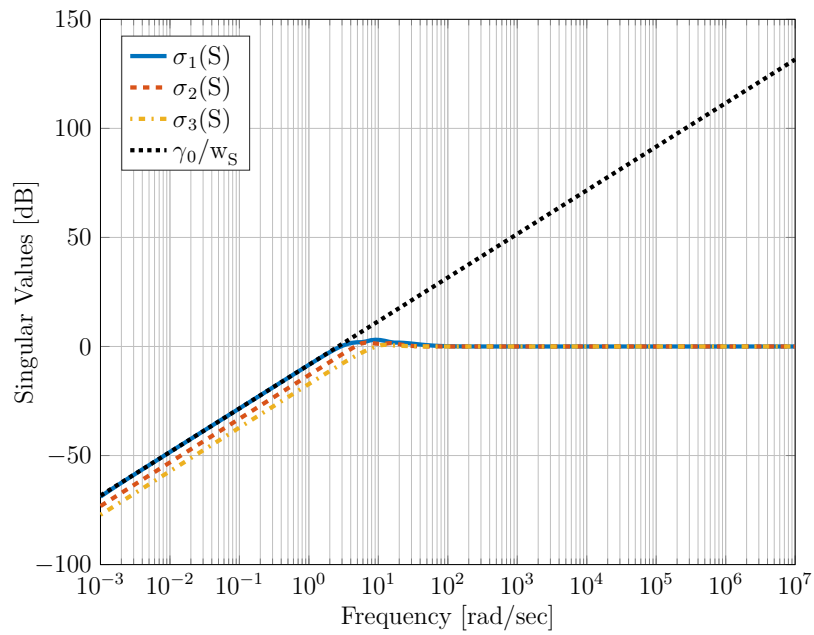
Figure 17.10: Design 2: Sensitivity S and scaled constraint
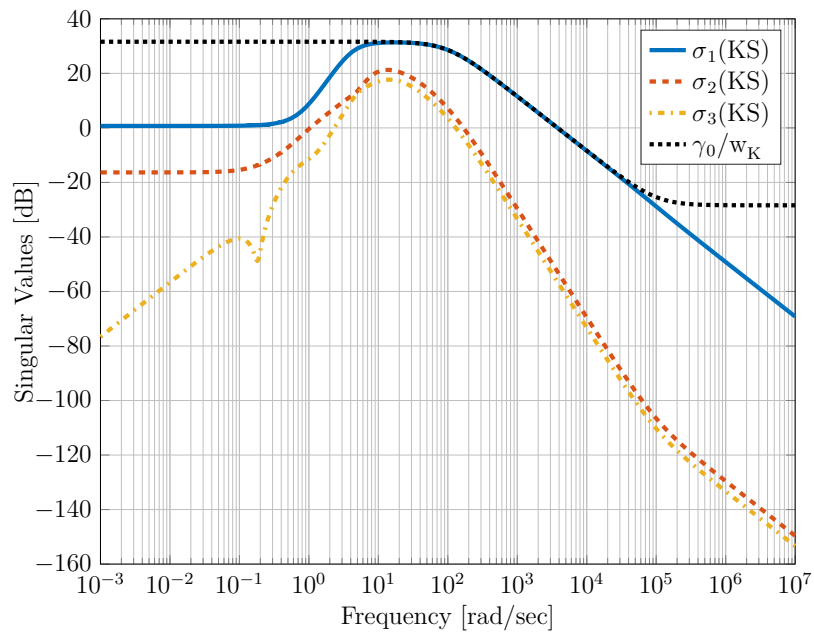
Figure 17.11: Design 2: Control sensitivity KS and scaled constraint

means the constraints will be violated. The violation of the constraints is spread equally over S and KS, indicated by the solid lines labelled 2. Rescaling the constraints then leads to the plots in Figures 17.10 and 17.11.

Returning to the step response in Figure 17.9, we noted that the control effort needs to be slightly reduced to meet the design specification. Moreover, the response to a reference
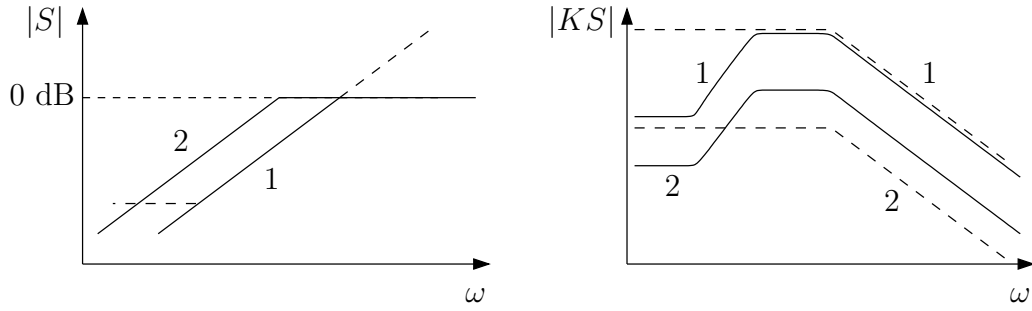
Figure 17.12: "Soft constraints" on sensitivity and control sensitivity: effect of reducing M_K

step change in y_1 leads to a perturbation in y_3. To improve tracking of r_3, we increase the weight on the corresponding channel in W_S and make the following adjustment to obtain Design 3:

$$W_S(s) = \begin{bmatrix} w_S(s) & 0 & 0 \\ 0 & w_S(s) & 0 \\ 0 & 0 & w_S(s) \end{bmatrix} \rightarrow \begin{bmatrix} w_S(s) & 0 & 0 \\ 0 & w_S(s) & 0 \\ 0 & 0 & 10w_S(s) \end{bmatrix}$$
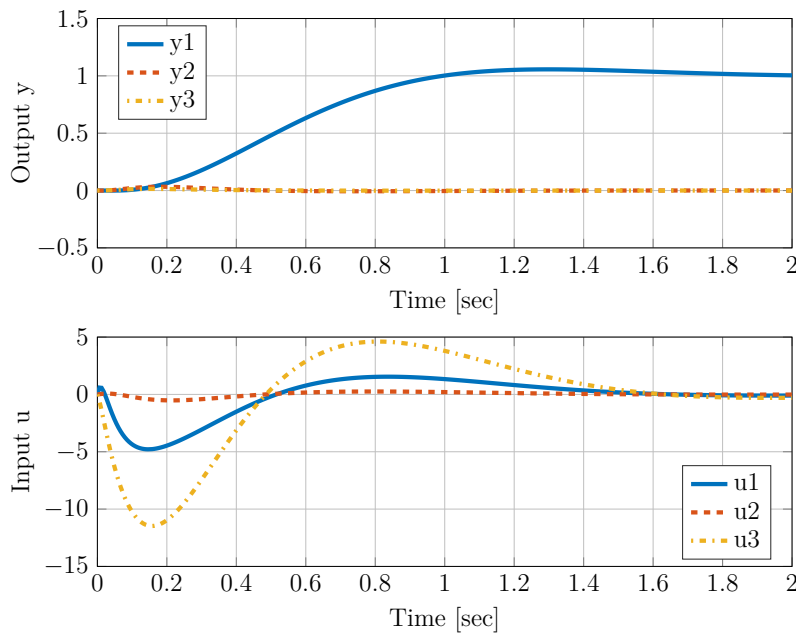

Figure 17.13: Design 3: response to $r(t) = [\sigma(t)\ \ 0\ \ 0]^T$

The step response with the resulting controller is shown in Figure 17.13. The perturbation of y_3 in the response has disappeared, and $|u_3| < 13$. This means that the controller is not using the full range available, and the response can be made faster by increasing the closed-loop bandwidth. With $\omega_S = 2.5 \cdot 10^{-4}$ we obtain
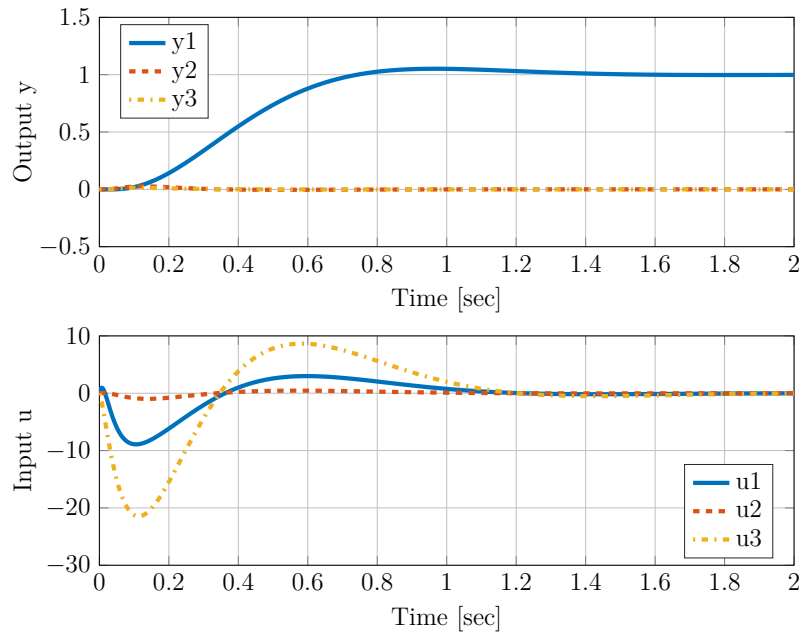
Figure 17.14: Design 4: response to $r(t) = [\sigma(t) \ \ 0 \ \ 0]^T$

Design 4:

ω_S	M_S	ω_K	M_K
$2.5 \cdot 10^{-4}$	10^{-4}	10^2	10^2

where $W_S = \text{diag}\,(w_S, w_S, 10w_S)$ and $W_K = \ \text{diag}\,(w_K, w_K, w_K)$. Figure 17.14 shows that the tracking performance is satisfactory.

Input Disturbances

The sensitivity S is the transfer function from an output disturbance to the controlled output, and when $|S|$ is made small at low frequencies, the closed-loop system has a built-in capability of good rejection of output disturbances. The same is not automatically guaranteed for disturbances occurring at the plant input, see Figure 17.15. In fact, even though the response to a reference step achieved with Design 4 is excellent, the closed-loop system shows poor rejection of input disturbances - the response to a disturbance step applied at plant input is shown in Figure 17.16.

The surprisingly poor response to an input step disturbance reveals a major weakness of the loop shaping approach: we are trying to shape the closed-loop dynamics without taking the open-loop dynamics into account. Often the only way to satisfy the constraints imposed on the closed-loop transfer function is to cancel undesirable plant poles by controller zeros. Figure 17.17 shows that this is exactly what happened here: shown are open-loop poles close to the imaginary axis - with a low damping ratio - that have been cancelled by transmission zeros of the controller.
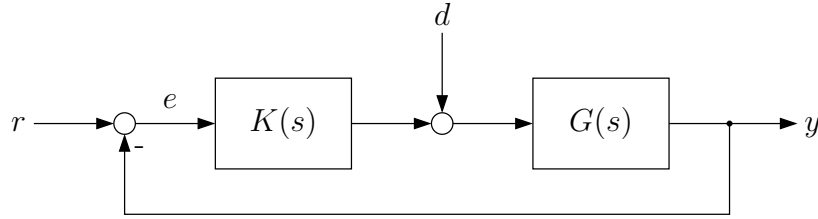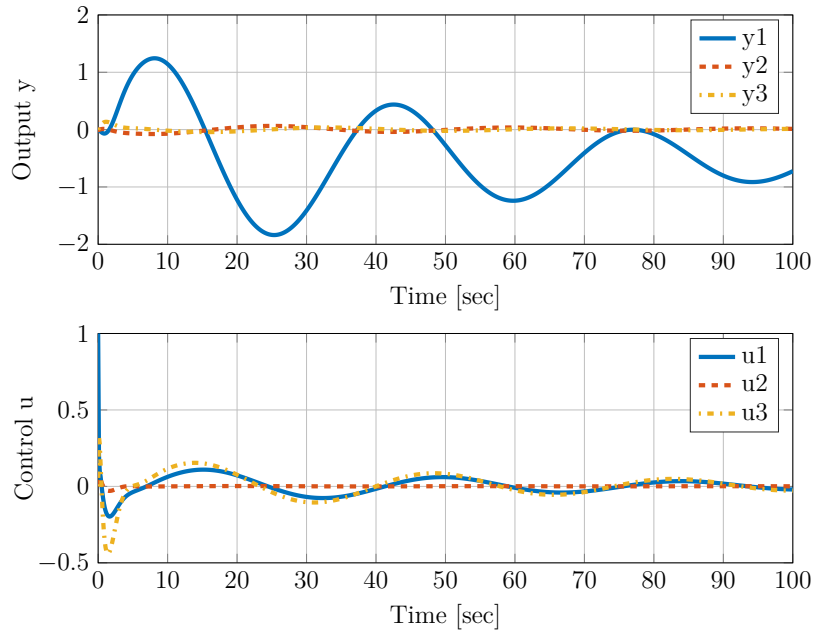
Figure 17.15: Input disturbance

Figure 17.16: Design 4: response to input disturbance $d(t) = [\sigma(t)\ 0\ 0]^T$

Because input disturbances are likely to occur - in the present example they could be caused by hysteresis and friction at the aircraft control surfaces - the design needs to be modified to improve its disturbance rejection. One way to deal with this problem is to consider the transfer function from the input disturbance d to the controlled output y. We have (assuming $r = 0$)

$$y = G(d - Ky) \qquad \text{thus} \qquad (I + GK)y = Gd$$

and

$$y = (I + GK)^{-1}Gd = SGd$$

Figure 17.18 shows the singular values of the transfer function SG - it clearly displays the resonant peak responsible for the poor response in Figure 17.16.

The problem is that the pole-zero cancellations are not visible when examining the transfer functions $S = (I+GK)^{-1}$ and $T = (I+GK)^{-1}GK$, but are visible in $SG = (I+GK)^{-1}G$. This can be seen most easily by considering the SISO version of this problem. Using the

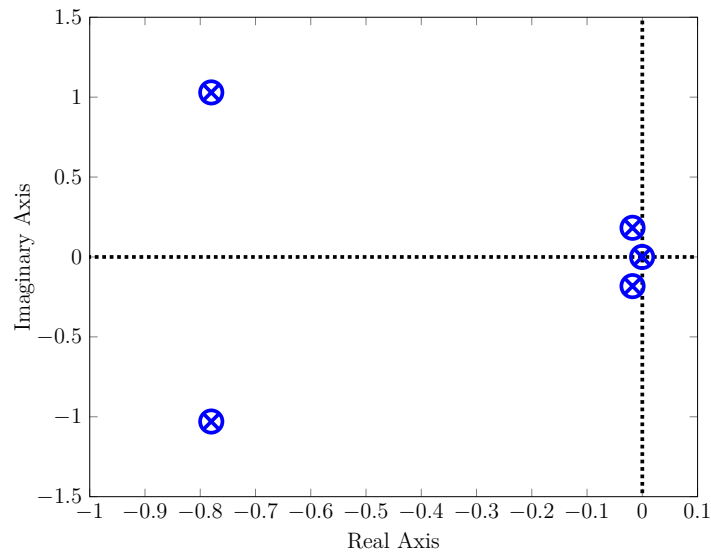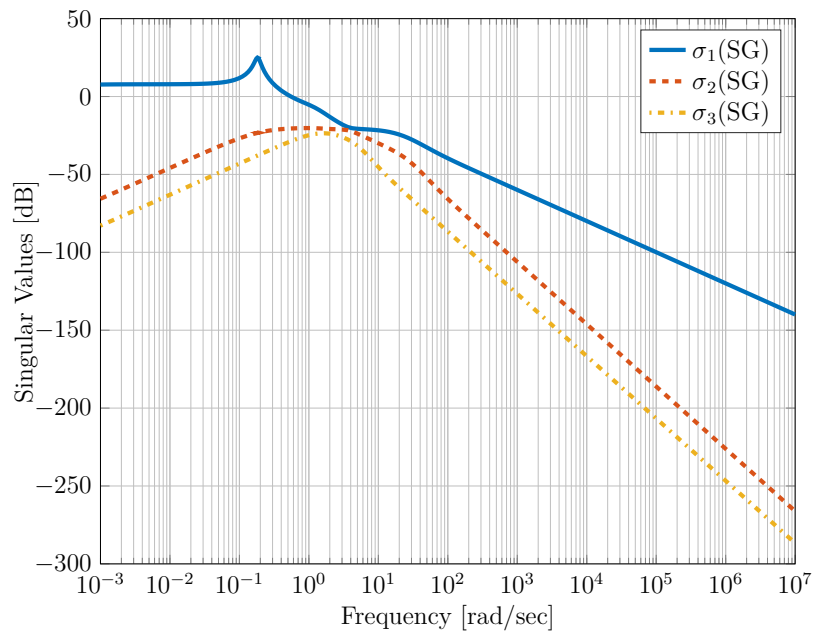Figure 17.17: Pole-zero cancellation in Design 4

Figure 17.18: Design 4: Singular values of SG

notation

$$G(s) = \frac{n_g(s)}{d_g(s)}, \qquad K(s) = \frac{n_k(s)}{d_k(s)}$$

we have

$$SG = \frac{G}{1 + GK} = \frac{n_g d_k}{d_g d_k + n_g n_k}$$

A pole-zero cancellation means that the polynomials d_g and n_k have a common factor, so

we can write

$$d_g = \bar{d}_g \tilde{d}_g \qquad \text{and} \qquad n_k = \bar{n}_k \tilde{d}_g$$

where $\tilde{d}_g$ contains the plant poles that are cancelled by controller zeros. Substituting this in the above yields

$$SG = \frac{n_g d_k}{\tilde{d}_g(\bar{d}_g d_k + n_g \bar{n}_k)}$$

which shows that the cancelled plant poles turn up as poles of the transfer function from d to y.
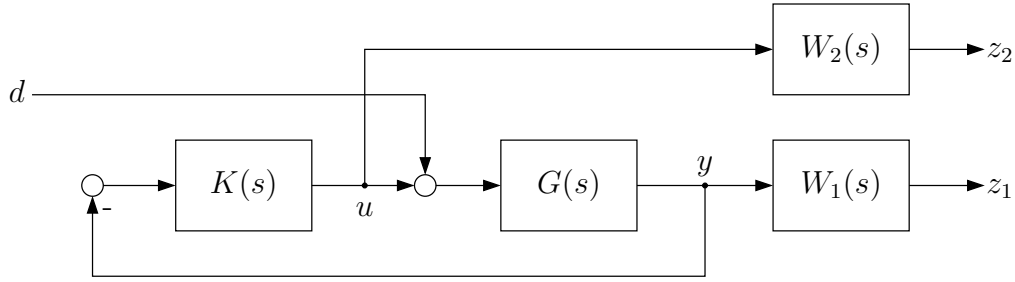

Figure 17.19: Generalized plant for input disturbance rejection

To prevent such undesirable pole-zero cancellations, we can try to shape the transfer function SG. Figure 17.19 shows one way of doing this: the weighting filter W_1 shapes SG, and a second filter W_2 shapes the transfer function from d to the controller output u. It is straightforward to check that the transfer function from d to u is

$$T_K = -(I + KG)^{-1}KG$$

This function has the same structure as the complementary sensitivity, only the loop gain GK is replaced by KG - the loop gain seen from the controller output. Defining $z = [z_1^T \ z_2^T]^T$, we can compute the controller that minimizes the H_∞ norm of the closed-loop transfer function from d to z. With the choices

$$w_1(s) = \frac{\omega_1/M_1}{s + \omega_1} \qquad \text{and} \qquad w_2(s) = \frac{c}{M_2} \frac{s + \omega_2}{s + c\omega_2}$$

where we fix again $c = 10^3$, and

Design 5:

ω_1	M_1	ω_2	M_2
10^{-3}	10^{-7}	10^2	2

we obtain the response shown in Figure 17.20 to an input disturbance step. The singular values of SG and T_K together with the scaled constraints are plotted in Figures 17.22 and 17.23. The plots show that the constraint on SG is active at low frequencies and the constraint on T_K near the corner frequency of $W_2(s)$.

The setup in Figure 17.19 does not consider the reference input r. Therefore, one cannot constrain the sensitivity S explicitly, but only as a factor in SG. So we would expect the tracking performance to be inferior to that in Design 4. This is confirmed by the response to a reference step input, shown in Figure 17.21. The control effort needs to be
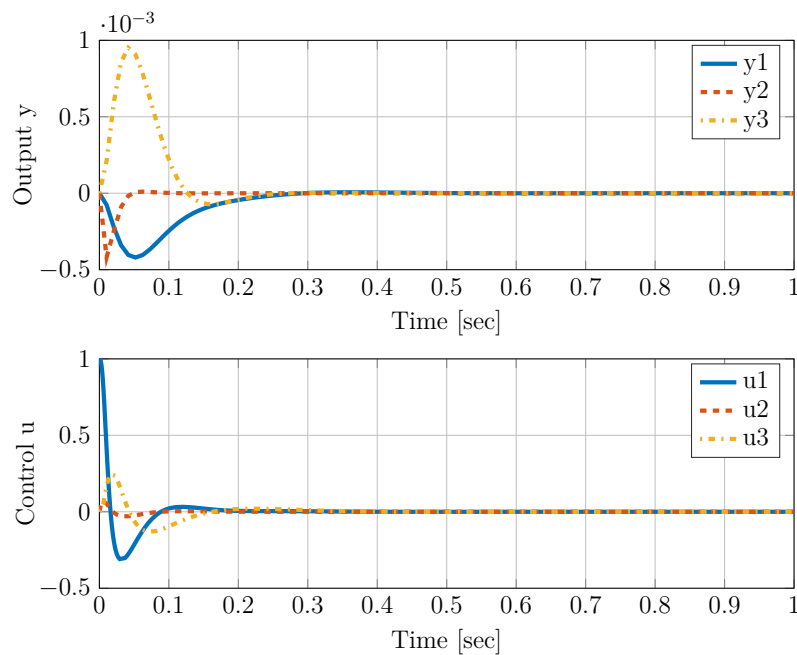

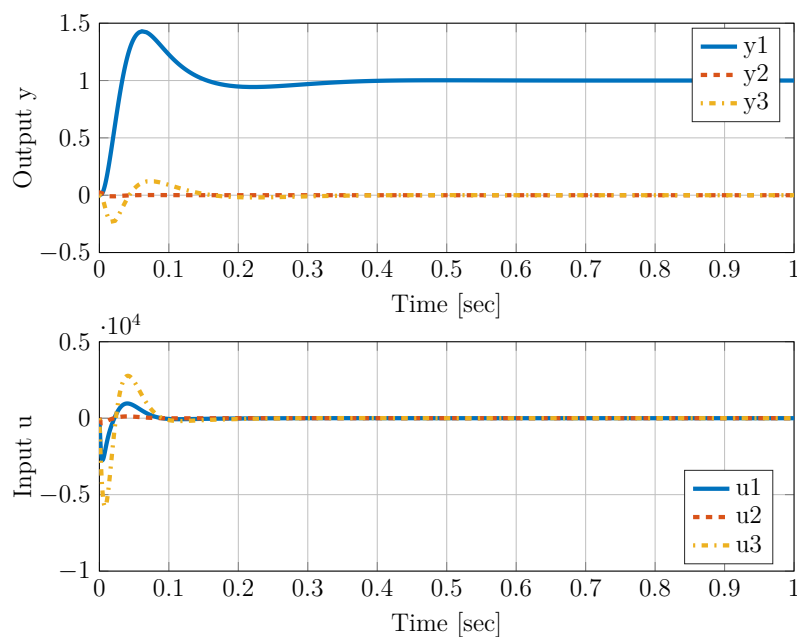Figure 17.20: Design 5: response to input disturbance $d(t) = [\sigma(t) \ \ 0 \ \ 0]^T$

Figure 17.21: Design 5: response to $r(t) = [\sigma(t) \ \ 0 \ \ 0]^T$

considerably reduced to meet the design specification. The final design for this problem is left as an exercise at the end of the chapter. The 4-block design is presented to achieve both good reference tracking and disturbance rejection with a reasonable control effort.
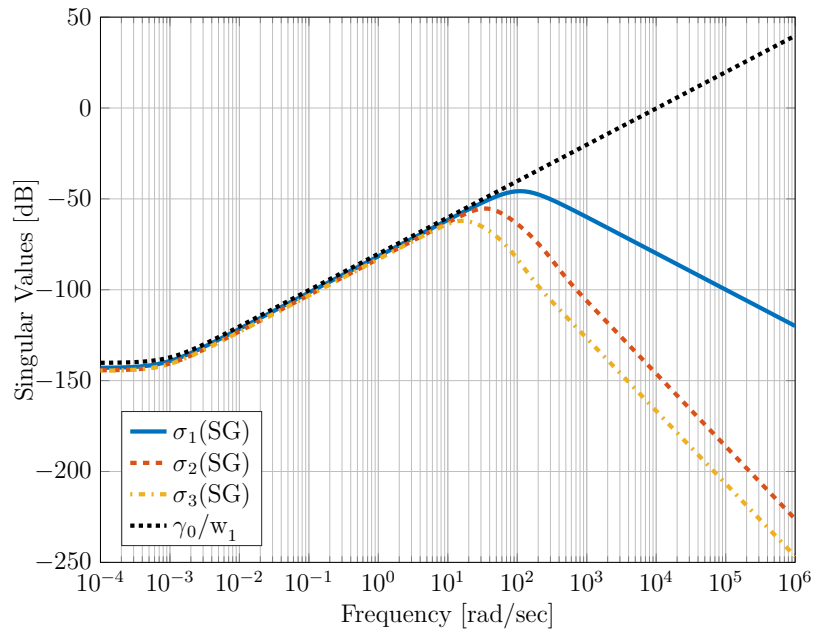

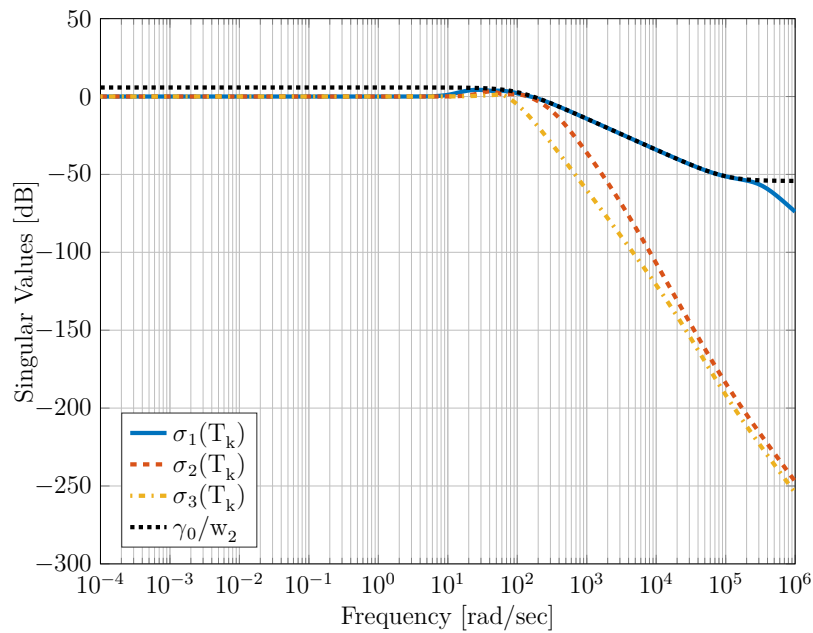Figure 17.22: Design 5: Singular values of SG and scaled constraint

Figure 17.23: Design 5: Singular values of T_K and scaled constraint

Constructing the Generalized Plant

The actual computation of the controller in the examples in this chapter was carried out with the function `hinfsyn()` of the robust control toolbox for MATLAB. The design constraints are entered in the form of generalized plant models. The construction of the generalized plant is illustrated here for the first four designs with constraints on S and KS.

The generalized plant is shown in Figure 17.24. Unlike the LQG example in chapter 11, the generalized plant P contains not only the dynamics of the physical plant

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$$

but also the dynamics of the weighting filters. It is convenient to construct first a generalized plant without the filters. Define $\tilde{z}_S$ and $\tilde{z}_K$ as the signals that enter the weighting filters, i.e. $\tilde{z}_S = e$ and $\tilde{z}_K = u$, and consider the transfer function $\tilde{P}$ obtained by replacing z with $\tilde{z}$

$$\begin{bmatrix} \tilde{z}_S \\ \tilde{z}_K \\ e \end{bmatrix} = \tilde{P} \begin{bmatrix} r \\ u \end{bmatrix}$$
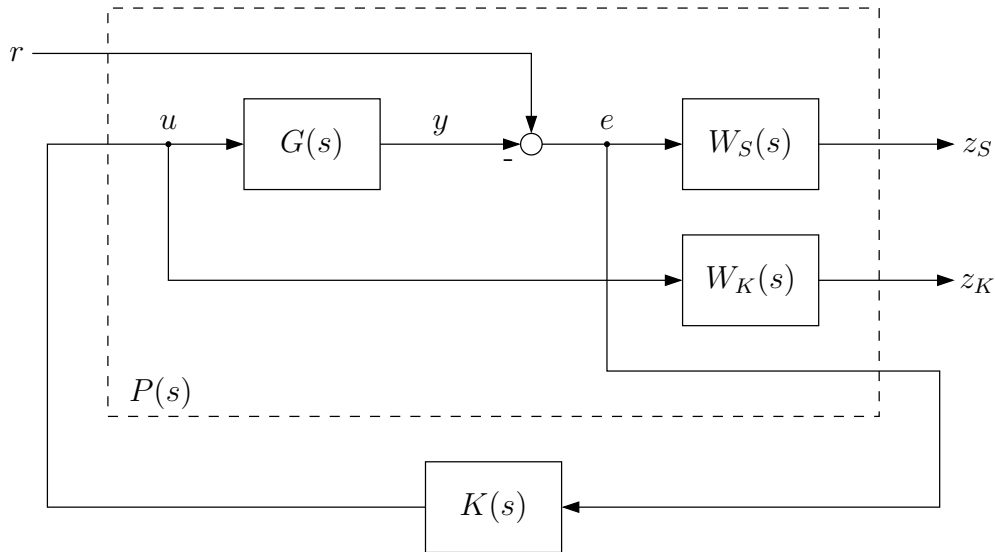

Figure 17.24: Generalized plant model for Design 1 - 4

From Figure 17.24, one finds that

$$\tilde{P}(s) = \left[\begin{array}{c|cc} A & 0 & B \\ \hline \begin{bmatrix} -C \\ 0 \end{bmatrix} & \begin{bmatrix} I \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ I \end{bmatrix} \\ -C & I & 0 \end{array}\right]$$

The generalized plant P is then easily obtained as

$$P(s) = \begin{bmatrix} w_S(s) & 0 & 0 \\ 0 & w_K(s) & 0 \\ 0 & 0 & I \end{bmatrix} \cdot \tilde{P}(s) = \left[\begin{array}{c|c} A_P & B_P \\ \hline C_P & D_P \end{array} \right]$$

The state space model (A_P, B_P, C_P, D_P) can then be used as input argument to the function `hinfsyn()`.

Exercises

Problem 17.1

Consider the configuration shown in Figure 17.25, where

$$G(s) = \frac{200}{10s + 1} \frac{1}{(0.05s + 1)^2}; \qquad G_d(s) = \frac{10}{10s + 1}$$

are the transfer functions to y from u and d respectively. The control objectives are
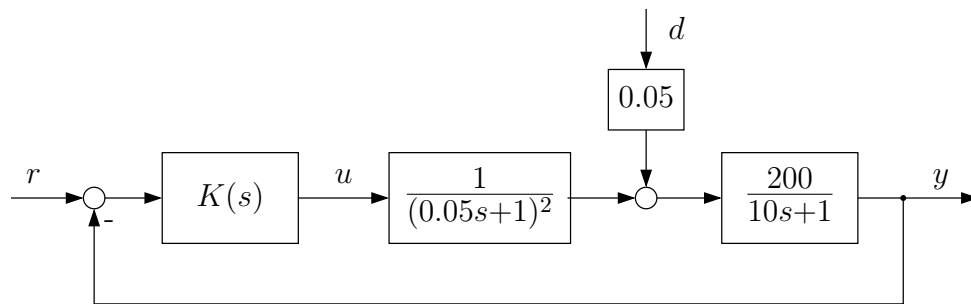

Figure 17.25: System block diagram

1. Command tracking: The rise time should be less than 0.3 sec. and the overshoot should be less than 5%.

2. Disturbance rejection: The output in response to a unit step disturbance should remain within the range $[-1, 1]$ at all times, and it should return to 0 as quickly as possible ($|y(t)|$ should at least be less than 0.1 after 3 sec.).

3. Input constraints: $u(t)$ should remain within the range $[-1, 1]$ at all times to avoid input saturation (this is easily satisfied for most designs).

a) Try to design a controller that achieves these objectives.

    Hint: Minimise

$$\left\| \begin{bmatrix} W_S S \\ W_K K S \end{bmatrix} \right\|_\infty$$

b) Repeat the design, but this time with 10 times stronger disturbance:
    $G_d(s) = 100/(10s + 1)$ (gain of 0.5 in disturbance gain block in Figure 17.25).

Problem 17.2

a) Design a controller for the aircraft model in example 11.1 that achieves good tracking
    properties. Initially, ignore input disturbances and measurement noise, and tune
    your controller for a fast response to reference step inputs. What is the fastest
    response that can be achieved within the given actuator constraints $|u_3| < 20$?

b) Check the noise sensitivity of your design. You can do this by injecting a sinusoidal
    input at 10^4 rad/sec into the feedback signals. Improve the noise sensitivity if
    required.

c) Check the rejection of input disturbance steps. Modify the design to improve input
    disturbance rejection.

Chapter 18

Design Specifications as LMI Constraints

So far we discussed the use of weighting functions for shaping the closed-loop dynamics via H_∞ constraints, and we assumed the availability of software tools for computing the controller. Similarly, in chapter 11 we discussed how the LQG problem can be reformulated in terms of a generalized plant, and we used available software tools for computing a LQG controller in Exercise 11.1. We used in fact the same function `hinfsyn()` of the robust control toolbox for MATLAB that is used in Exercise 17.2 for H_∞ loop shaping. Both H_∞ constraints and H_2 constraints can be expressed in the form of linear matrix inequalities (LMI), and efficient LMI solvers can be employed to compute controllers that satisfy these constraints. In this chapter we will study how to express design specifications in the form of linear matrix inequalities. The next chapter will then introduce methods for computing controllers by solving LMI problems.

We will discuss three types of design specifications: constraints on the H_2 and the H_∞ norm, and constraints on the location of closed-loop poles in the complex plane.

Linear Matrix Inequalities

We begin with a brief review of linear matrix inequalities. A linear matrix inequality (LMI) has the form

$$M(p) = M_0 + p_1 M_1 + \cdots + p_N M_N \; < \; 0 \tag{18.1}$$

where $M_0, \; M_1, \ldots, \; M_N$ are given symmetric matrices, $p = \begin{bmatrix} p_1 & p_2 & \ldots & p_N \end{bmatrix}^T$ is a column vector of real scalar variables (the decision variables), and the matrix inequality $M(p) < 0$ means that the left hand side is negative definite. An important property of LMIs is that the set of all solutions p is convex.

Linear matrix inequalities can be used as constraints for the minimization problem

$$\min_p c^T p \qquad \text{subject to} \qquad M(p) < 0 \tag{18.2}$$

where the elements of the vector c in the linear cost function are weights on the individual decision variables. This problem is convex and can be solved by efficient, polynomial-time interior-point methods. Several LMI constraints can be combined into a single constraint of type (18.1). For example the constraint

$$M_1(p) < 0 \qquad \text{and} \qquad M_2(p) < 0$$

is equivalent to the single LMI constraint

$$\begin{bmatrix} M_1(p) & 0 \\ 0 & M_2(p) \end{bmatrix} < 0 \tag{18.3}$$

The problem (18.2) is quite general, and a variety of problems can be reduced to this form. It has been known for a long time that certain control problems can be expressed in this way. Already in 1971 the question was raised whether linear matrix inequalities can be exploited for numerical purposes in control engineering, but it took more than 20 years for this to happen. It became possible when interior-point methods for solving LMI problems were developed in the the 1980s. Control engineers became aware of these developments in the early 1990s, standard software tools were made available in the mid 1990s, and from then on LMI techniques developed into an active area of research.

Optimization problems of the form (18.2) will be used here to search for the best controller - in the sense of a given performance index - that satisfies LMI constraints on the closed-loop system shown in Figure 18.1.
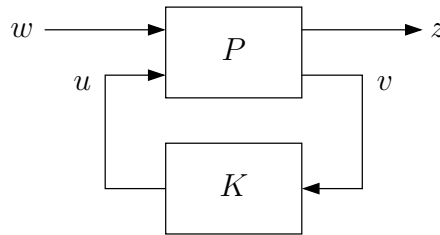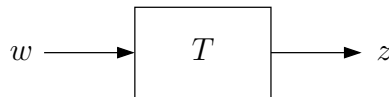

Figure 18.1: Generalized plant

Figure 18.2: Closed-loop system

In this chapter, we will only consider constraints on the transfer function from w to z, without discussing its dependence on the controller $K(s)$ - so we will consider the system shown in Figure 18.2. How to construct controllers that satisfy the constraints will be the subject of the following chapter.
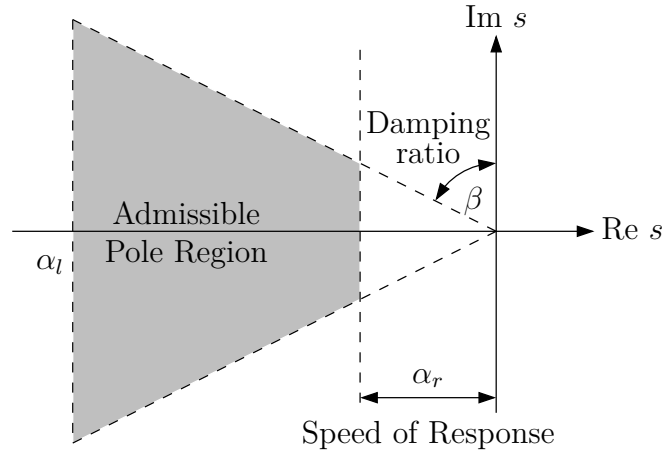
Figure 18.3: Desired pole region

Pole Region Constraints

It is well known that the dynamic properties of a linear, time-invariant system are determined by the location of the poles of its transfer function in the complex plane. For second order systems, the relationship between the pole location and damping ratio, rise time and settling time is particularly simple. For higher order systems these relations are still useful because system dynamics are often governed by a dominant pole pair. In practice the designer often works on specifications that include a minimum damping ratio and a minimum speed of response, and the objective is to find the best controller (in terms of a suitable performance index) that satisfies these constraints. These constraints can be expressed as a region in the complex plane where the closed-loop poles should be located, as shown in Figure 18.3. A minimum speed of response is represented by a right boundary on the admissible pole region at $-\alpha_r$, and a minimum damping ratio by a conic sector. To avoid excessive control action and high noise sensitivity, the closed-loop bandwidth can be limited by placing a left boundary at $-\alpha_l$ on the admissible pole region.

The condition that the poles of a system are located within a given region in the complex plane can be formulated as an LMI constraint. As an example, consider the dynamic system

$$\dot{x}(t) = Ax(t) \tag{18.4}$$

This system is stable if and only if the matrix A has all eigenvalues in the left half plane, which in turn is true if and only if there exists a positive definite, symmetric matrix P that satisfies the Lyapunov inequality

$$PA^T + AP < 0 \tag{18.5}$$

This inequality is linear in the matrix variable P, and one can use LMI solvers to search for solutions. It is straightforward to rewrite (18.5) in the standard form (18.1) of an LMI. To see this, assume that A is a 2 by 2 matrix and write the symmetric matrix variable P

as

$$P = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_3 \end{bmatrix} = p_1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + p_2 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + p_3 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Substitution of the right hand side for P in (18.5) yields an LMI in the form of (18.1) with three decision variables p_1, p_2 and p_3. Available software tools operate directly on matrix variables, so that it is usually not necessary to carry out this transformation.
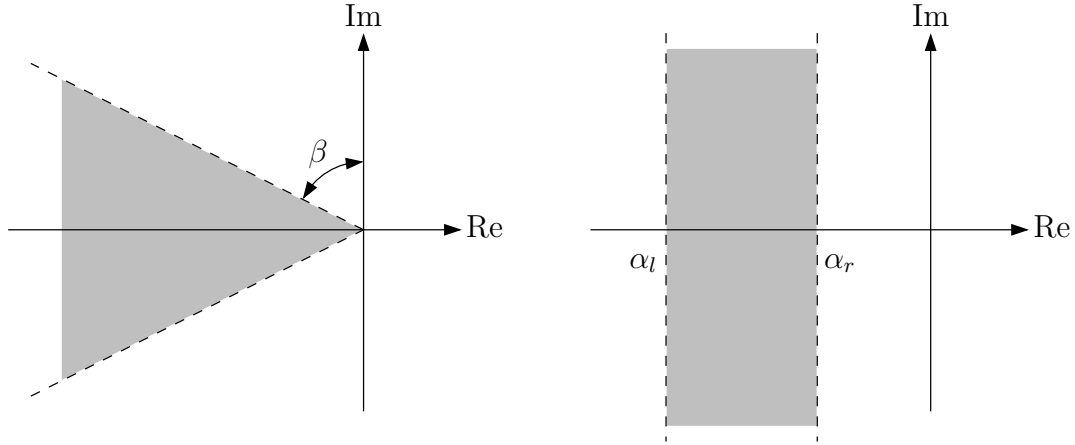

Figure 18.4: LMI regions

The LMI (18.5) represents a necessary and sufficient condition for the matrix A to have all eigenvalues in the left half plane. It is possible to generalize this result: one can express an arbitrary region $\mathcal{D}$ in the complex plane such as the one shown in Figure 18.3 (as long as it is convex and symmetric about the real axis) in terms of two matrices $L = L^T$ and M as the set of all complex numbers that satisfy an LMI constraint

$$\mathcal{D} = \{s \in \mathbb{C} : L + Ms + M^T \bar{s} < 0\} \tag{18.6}$$

where $\bar{s}$ denotes the complex conjugate of s. Such a region is called an *LMI region*. One can show that a necessary and sufficient condition for a matrix A to have all eigenvalues in $\mathcal{D}$ is the existence of a positive definite, symmetric matrix P that satisfies

$$L \otimes P + M \otimes (AP) + M^T \otimes (AP)^T < 0 \tag{18.7}$$

The symbol $\otimes$ stands for the Kronecker product: if M is a 2 by 2 matrix then

$$M \otimes P = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \otimes P = \begin{bmatrix} m_{11}P & m_{12}P \\ m_{21}P & m_{22}P \end{bmatrix}$$

Thus if P is also 2 by 2 the Kronecker product is 4 by 4. Inequality (18.7) is an LMI in the matrix variable P. It is easy to see that the Lyapunov inequality (18.5) is obtained as a special case of (18.7) with $L = 0$ and $M = 1$ and $\mathcal{D}$ as the left half plane. One can verify that the conic sector shown in Figure 18.4 is an LMI region with

$$L_c = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad M_c = \begin{bmatrix} \cos\beta & -\sin\beta \\ \sin\beta & \cos\beta \end{bmatrix} \tag{18.8}$$

and the vertical strip is an LMI region with

$$L_v = \begin{bmatrix} 2\alpha_l & 0 \\ 0 & -2\alpha_r \end{bmatrix}, \qquad M_v = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{18.9}$$

The intersection of both regions is the pole region shown in Figure 18.3; it can be represented as an LMI region by combining the two constraints as in (18.3) with

$$L = \begin{bmatrix} L_c & 0 \\ 0 & L_v \end{bmatrix}, \qquad M = \begin{bmatrix} M_c & 0 \\ 0 & M_v \end{bmatrix} \tag{18.10}$$

Constraints on the H_2 Norm

Consider a stable system with state space realization

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bw(t) \\ z(t) &= Cx(t) \end{aligned} \tag{18.11}$$

We will later interpret this system as the closed-loop system in Figure 18.2. Let $T(s)$ denote the strictly proper transfer function matrix from w to z

$$T(s) = C(sI - A)^{-1}B$$

It was shown in chapter 15 that the H_2 norm of $T(s)$ can be computed from

$$\|T\|_2^2 = \operatorname{trace} CP_0 C^T \tag{18.12}$$

where P_0 is the positive definite matrix that solves the Lyapunov equation

$$AP_0 + P_0 A^T + BB^T = 0 \tag{18.13}$$

Now assume that we want to impose a constraint on $\|T\|_2$, say

$$\|T\|_2 < \nu$$

for some $\nu > 0$. To see how (18.12) and (18.13) can be used to express this constraint, consider the solution P to the following modified version of (18.13)

$$AP + PA^T + BB^T + Q = 0 \tag{18.14}$$

where $Q > 0$ is a positive definite matrix. Subtracting (18.13) from (18.14) gives

$$A(P - P_0) + (P - P_0)A^T + Q = 0$$

Because $Q > 0$ and A is stable, this implies $P - P_0 > 0$ or

$$P > P_0 \tag{18.15}$$

The reverse is also true: if (18.15) holds, then a matrix $Q > 0$ exists that satisfies (18.14). Since the above holds for any $Q > 0$, we can replace (18.14) by the matrix inequality

$$AP + PA^T + BB^T < 0 \tag{18.16}$$

and we have from (18.15) and (18.12) that

$$\|T\|_2^2 < \operatorname{trace} CPC^T \tag{18.17}$$

if and only if P satisfies (18.16). The above results can be summarized as

Theorem 18.1 *For the system (18.11), $\|T\|_2 < \nu$ if and only if there exists a positive definite matrix P that satisfies (18.16) and*

$$\operatorname{trace} CPC^T < \nu^2 \tag{18.18}$$

In the next chapter it will be shown how to compute controllers that satisfy constraints on $\|T\|_2$. The system matrices A, B and C will then be closed-loop system matrices that contain controller parameters, and the terms BB^T and $\operatorname{trace} CPC^T$ will be nonlinear in these parameters. The following equivalent formulation of the above result is then used instead. Introduce a new symmetric matrix variable W (which is used as a slack variable), then we have

Theorem 18.2 *$\|T\|_2 < \nu$ if and only if there exist symmetric matrices P and W that satisfy*

$$\begin{bmatrix} AP + PA^T & B \\ B^T & -I \end{bmatrix} < 0, \qquad \begin{bmatrix} W & CP \\ PC^T & P \end{bmatrix} > 0 \qquad and \qquad \operatorname{trace} W < \nu^2 \tag{18.19}$$

To see that the inequalities in (18.19) are equivalent to (18.18) and (18.16), note that

$$\begin{bmatrix} M & L \\ L^T & N \end{bmatrix} > 0$$

where $M = M^T$ and $N = N^T$, is equivalent to

$$N > 0 \qquad and \qquad M - LN^{-1}L^T > 0$$

This fact is frequently used to convert nonlinear inequalities into LMI form; the term $M - LN^{-1}L^T$ is the Schur complement with respect to N. In the above case, this leads to $W > CPC^T$ and consequently to $\operatorname{trace} W > \operatorname{trace} CPC^T$, from which the equivalence of (18.18) and (18.19) follows.

Constraints on the H_∞ Norm

Consider again the system in Figure 18.2 with transfer function $T(s)$ and state space realization

$$\dot{x}(t) = Ax(t) + Bw(t), \qquad x(0) = 0$$
$$z(t) = Cx(t) + Dw(t) \tag{18.20}$$

Assuming that $T(s)$ is stable, the H_∞ norm of the system is

$$\|T\|_\infty^2 = \max_{w \neq 0} \frac{\int_0^\infty z^T(t)z(t)\,dt}{\int_0^\infty w^T(t)w(t)\,dt}$$

where we also assume that $x(0) = 0$. From the above it follows that $\|T\|_\infty < \gamma$ is equivalent to

$$\int_0^\infty (z^T(t)z(t) - \gamma^2 w^T(t)w(t))\,dt < 0$$

holding true for all square integrable, non-zero $w(t)$. Now introduce a Lyapunov function $V(x) = x^T P x$ with $P = P^T > 0$. Because $x(0) = x(\infty) = 0$, the constraint $\|T\|_\infty < \gamma$ is then enforced by the existence of a matrix $P = P^T > 0$ such that

$$\frac{dV(x)}{dt} + \frac{1}{\gamma}z^T(t)z(t) - \gamma w^T(t)w(t) < 0 \tag{18.21}$$

for all $x(t)$, $w(t)$; this can be seen by integrating (18.21) from $t = 0$ to $t = \infty$. To turn (18.21) into a linear matrix inequality, substitute

$$\frac{dV(x)}{dt} = x^T(A^T P + PA)x + x^T PBw + w^T B^T Px$$

and $z = Cx + Dw$ in (18.21) to obtain

$$\begin{bmatrix} x^T & w^T \end{bmatrix} \begin{bmatrix} A^T P + PA + \frac{1}{\gamma}C^T C & PB + \frac{1}{\gamma}C^T D \\ B^T P + \frac{1}{\gamma}D^T C & -\gamma I + \frac{1}{\gamma}D^T D \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} < 0$$

For $\|T\|_\infty < \gamma$ the above must hold for all x and w, thus the block matrix must be negative definite. This condition can be rewritten as

$$\begin{bmatrix} A^T P + PA & PB \\ B^T P & -\gamma I \end{bmatrix} + \frac{1}{\gamma}\begin{bmatrix} C^T \\ D^T \end{bmatrix}\begin{bmatrix} C & D \end{bmatrix} < 0$$

It can be shown that the existence of a solution to the above LMI is in fact not only a sufficient but also a necessary condition for $\|T\|_\infty < \gamma$. Using the Schur complement, we can rewrite this condition and summarize the result as follows.

Theorem 18.3 $\|T\|_\infty < \gamma$ *if and only if there exists a positive definite, symmetric matrix P that satisfies the linear matrix inequality*

$$\begin{bmatrix} A^T P + PA & PB & C^T \\ B^T P & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0 \tag{18.22}$$

Exercises

Problem 18.1

Show that the matrices $L = L^T$ and M from (18.6) define the LMI region $\mathcal{D}(\alpha_r, \alpha_l, \beta)$ shown in Figure 18.3.

Problem 18.2

Write MATLAB programs computing the H_∞ norm and the H_2 norm of a system using the LMI conditions (18.22) and (18.19). Use the "LMI lab" toolbox of MATLAB to express the LMI conditions. Verify your results by computing the H_2 and H_∞ norms of the closed-loop system $T(s)$ in Exercise 11.1 and compare the results with the results obtained by the `norm` function (see Exercise 15.1).

Hint: To get started see either the help of `rctlmi` *or the documentation of LMI lab in "Robust Control Toolbox", "Linear Matrix Inequalities", "The LMI lab" in the MATLAB help.*

Chapter 19

Controller Design Using Linear Matrix Inequalities

In the previous chapter it was shown how design specifications can be expressed as linear matrix inequality constraints. In this chapter we will see how one can compute controllers that satisfy these constraints.

We will now interpret the plant model on which constraints were placed in the previous chapter, as the closed loop system in Figure 19.1.
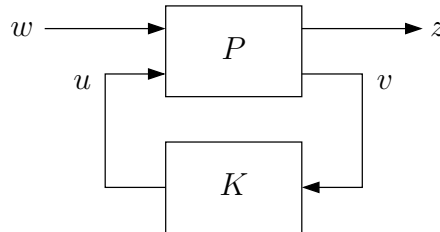

Figure 19.1: Generalized plant

Recall that the generalized plant $P(s)$ has a state space realization

$$
\begin{aligned}
\dot{x}(t) &= Ax(t) + B_w w(t) + B_u u(t) \\
z(t) &= C_z x(t) + D_{zw} w(t) + D_{zu} u(t) \\
v(t) &= C_v x(t) + D_{vw} w(t)
\end{aligned}
\tag{19.1}
$$

where $w(t)$ represents external inputs (reference input, disturbances and noise), $u(t)$ represents the control input, $z(t)$ is a vector of fictitious output signals used to assess the performance of the control system, and $v(t)$ is the measured output that is used for feedback control. The type of controller considered here is dynamic, linear time-invariant

output feedback control. The controller dynamics are represented by a state space model

$$\dot{\zeta}(t) = A_K \zeta(t) + B_K v(t)$$
$$u(t) = C_K \zeta(t) + D_K v(t) \tag{19.2}$$

To use the LMI constraints in the previous chapter for controller synthesis, they must be applied to the closed-loop system matrices. It is straightforward to check that a state space realization of the closed loop-system in Figure 19.1 is

$$\dot{x}_c(t) = A_c x_c(t) + B_c w(t)$$
$$z(t) = C_c x_c(t) + D_c w(t) \tag{19.3}$$

where

$$A_c = \begin{bmatrix} A + B_u D_K C_v & B_u C_K \\ B_K C_v & A_K \end{bmatrix}, \qquad B_c = \begin{bmatrix} B_w + B_u D_K D_{vw} \\ B_K D_{vw} \end{bmatrix}$$

$$C_c = \begin{bmatrix} C_z + D_{zu} D_K C_v & D_{zu} C_K \end{bmatrix}, \qquad D_c = D_{zw} + D_{zu} D_K D_{vw} \tag{19.4}$$

Let the transfer function from w to z of the closed-loop system be

$$T(s) = \left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array} \right] = C_c(sI - A_c)^{-1} B_c + D_c$$

The matrices (A_c, B_c, C_c, D_c) depend on the controller $K(s)$. The problem considered in this chapter is to design a controller such that constraints on the eigenvalue locations of A_c and on $\|T\|_2$ and $\|T\|_\infty$ are satisfied.

Linearizing Change of Variables - State Feedback

Three LMI constraints were discussed in the previous chapter:

- a pole region constraint in the form of matrix inequality (18.7) which is linear in the matrix variable P

- a bound on the H_2-norm in the form of matrix inequalities (18.16) and (18.19) which are linear in P and W

- a bound on the H_∞-norm in the form of matrix inequality (18.22) which is linear in P.

When the system matrices A, B, C and D in these inequalities are replaced by the closed-loop matrices A_c, B_c, C_c and D_c, the controller matrices A_K, B_K, C_K and D_K appear as variables, and the resulting inequalities are not linear because they contain products of variables. As a consequence, they cannot be solved with LMI solvers. One

can however introduce a change of variables to transform the constraints into equivalent LMI constraints which can be solved. This transformation is much simpler in the special case where all state variables are measured and available for feedback (i.e. the special case $C_v = I$) and where the controller has the form of static state feedback

$$u(t) = Fx(t)$$

This corresponds to the choice of controller matrices $A_K = 0$, $B_K = 0$, $C_K = 0$ and $D_K = F$.

In this section, a full discussion of the state feedback case is given to illustrate the linearizing change of variables, and in the following section an outline is given for the output feedback case.

Pole Region Constraints

Consider first the LMI condition for the homogenous system (18.4): the existence of a symmetric matrix P that satisfies the Lyapunov inequality

$$PA^T + AP < 0$$

is a necessary and sufficient condition for the system to be stable. In order to use this condition to search for stabilizing state feedback controllers, the system matrix A can be replaced by the system matrix of the closed-loop system

$$\dot{x}(t) = (A + B_u F)x(t)$$

The Lyapunov inequality becomes

$$P(A + B_u F)^T + (A + B_u F)P < 0$$

and because the variables are now P and F, this inequality is nonlinear in the variables. It is however easy to make it linear - introduce a new variable $Y = FP$ to obtain

$$PA^T + AP + Y^T B_u^T + B_u Y < 0$$

which is linear in the variables P and Y and can be solved with LMI solvers. Moreover, the set of all solution pairs (P, Y) represents the set of all stabilizing state feedback controllers $F = YP^{-1}$.

The same variable transformation can be applied to general pole region constraints in the form of (18.7): a necessary and sufficient condition for a state feedback controller $u = Fx$ to place the closed-loop poles inside an LMI region characterized by two matrices L and M, is the existence of matrices $P = P^T > 0$ and Y that satisfy the LMI

$$L \otimes P + M \otimes (AP + B_u Y) + M^T \otimes (AP + B_u Y)^T < 0 \tag{19.5}$$

Moreover, the set of all solution pairs (P, Y) represents the set of all state feedback controllers $F = YP^{-1}$ that place the closed-loop poles inside the specified region.

H_2 **Performance**

It was shown in the previous chapter that the H_2-norm of the system (18.11) is less than ν if and only if matrices $P = P^T > 0$ and $W = W^T$ exist that satisfy

$$AP + PA^T + BB^T < 0, \qquad \begin{bmatrix} W & CP \\ PC^T & P \end{bmatrix} > 0 \qquad \text{and} \qquad \text{trace}\, W < \nu^2 \qquad (19.6)$$

To use these conditions as constraints for controller synthesis, the system matrices A, B and C must be replaced by the matrices of the closed-loop system. The system (18.11) does not have a control input; if the system (19.1) is considered instead with $D_{zw} = 0$, $D_{vw} = 0$ and $C_v = I$, the closed-loop system under state feedback $u = Fx$ becomes

$$\dot{x}(t) = (A + B_u F)x(t) + B_w w(t)$$
$$z(t) = (C_z + D_{zu}F)x(t)$$

If A and C are replaced in (19.6) by $(A + B_u F)$ and $(C_z + D_{zu}F)$, respectively, the resulting inequalities are nonlinear in the variables P, W and F, but they can again be made linear by introducing $Y = FP$ as a new variable. It follows that a necessary and sufficient condition for a state feedback controller to achieve a H_2-norm less than ν is the existence of matrices $P = P^T > 0$, $W = W^T$ and Y that satisfy the LMIs

$$PA^T + AP + Y^T B_u^T + B_u Y + B_w B_w^T < 0 \qquad (19.7)$$

$$\begin{bmatrix} W & C_z P + D_{zu}Y \\ PC_z^T + Y^T D_{zu}^T & P \end{bmatrix} > 0 \qquad (19.8)$$

$$\text{and} \qquad \text{trace}\, W < \nu^2$$

and the set of all solutions (P, W, Y) represents the set of all state feedback controllers $F = YP^{-1}$ that achieve this norm constraint.

H_∞ **Performance**

The H_∞ constraint (18.22) can be used as a constraint for controller synthesis in the same way as the H_2 constraint. It is however convenient to rewrite it as

$$\begin{bmatrix} AP + PA^T & B & PC^T \\ B^T & -\gamma I & D^T \\ CP & D & -\gamma I \end{bmatrix} < 0 \qquad (19.9)$$

To verify that this condition is equivalent to (18.22), introduce $\tilde{P} = P^{-1}$ and multiply (18.22) from left and right by $\text{diag}(\tilde{P}, I, I)$. In (19.9) the tilde has been dropped.

State feedback $u = Fx$ applied to the system (19.1) yields the closed-loop system

$$\dot{x}(t) = (A + B_u F)x(t) + B_w w(t)$$
$$z(t) = (C_z + D_{zu}F)x(t) + D_{zw}w(t)$$

Replacing the system matrices in (19.9) by the closed-loop matrices and using again the variable transformation $Y = FP$ leads to the following result: a necessary and sufficient condition for a state feedback controller to achieve a H_∞-norm less than γ is the existence of matrices $P = P^T > 0$ and Y that satisfy

$$\begin{bmatrix} AP + PA^T + B_u Y + Y^T B_u^T & B_w & PC_z^T + Y^T D_{zu}^T \\ B_w^T & -\gamma I & D_{zw}^T \\ C_z P + D_{zu} Y & D_{zw} & -\gamma I \end{bmatrix} < 0 \qquad (19.10)$$

and the set of all solutions (P, Y) represents the set of all state feedback controllers $F = YP^{-1}$ that achieve this norm constraint.

Linearizing Change of Variables - Output Feedback

When a dynamic controller (19.2) is used instead of state feedback in the configuration shown in Figure 11.1, the resulting matrix inequality constraints on the closed-loop system are again nonlinear in the variables. As in the state feedback case, they can be transformed into equivalent LMI constraints by a change of variables. However, whereas with state feedback the introduction of a new variable $Y = FP$ was all that was required to make the constraints linear, the situation is more complex in the output feedback case, and in this section only an outline of the approach is presented.

Pole Region Constraints

Consider the pole region constraint

$$L \otimes P + M \otimes (PA_c)^T + M^T \otimes (PA_c) < 0 \qquad (19.11)$$

for the closed-loop system matrix

$$A_c = \begin{bmatrix} A + B_u D_K C_v & B_u C_K \\ B_K C_v & A_K \end{bmatrix}$$

to have its eigenvalues in $\mathcal{D}$, where $\mathcal{D}$ is an LMI region characterized by the two matrices L and M as in (18.6). In (19.11) A_c has been replaced by A_c^T (which has the same eigenvalues). The matrix variables are P and the controller matrices A_K, B_K, C_K and D_K; the inequality is nonlinear because the term PA_c contains products of variables. In the following it will be assumed that the controller has the same order as the plant, i.e. A_K has the same size as A. Now partition P and P^{-1} as

$$P = \begin{bmatrix} X & R \\ R^T & * \end{bmatrix}, \qquad P^{-1} = \begin{bmatrix} Y & S \\ S^T & * \end{bmatrix}$$

where dimensions are compatible with the block matrix A_c, X and Y are symmetric and $(*)$ denotes a matrix block that is of no particular interest. One can verify that

$$PT_Y = T_X \qquad \text{where} \qquad T_Y = \begin{bmatrix} Y & I \\ S^T & 0 \end{bmatrix}, \qquad T_X = \begin{bmatrix} I & X \\ 0 & R^T \end{bmatrix} \tag{19.12}$$

The matrices T_X and T_Y can be used to transform the nonlinear constraint (19.11) into a linear one. This transformation is based on the fact that

$$T_Y^T P A_c T_Y = T_X^T A_c T_Y = \begin{bmatrix} AY + B_u \tilde{C}_K & A + B_u \tilde{D}_K C_v \\ \tilde{A}_K & XA + \tilde{B}_K C_v \end{bmatrix} \tag{19.13}$$

where new controller variables

$$\begin{aligned}
\tilde{A}_K &= RA_K S^T + RB_K C_v Y + XB_u C_K S^T + X(A + B_u D_K C_v)Y \\
\tilde{B}_K &= RB_K + XB_u D_K \\
\tilde{C}_K &= C_K S^T + D_K C_v Y \\
\tilde{D}_K &= D_K
\end{aligned} \tag{19.14}$$

have been introduced. Moreover

$$T_Y^T P T_Y = \begin{bmatrix} Y & I \\ I & X \end{bmatrix} \tag{19.15}$$

The important point here is that the block matrices in (19.13) and (19.15) are linear in the transformed controller variables (19.14) and the auxiliary variables X and Y. Using the fact that for a nonsingular matrix T_Y the condition $M < 0$ is equivalent to $T_Y^T M T_Y < 0$, such a congruence transformation can be applied to (19.11) - where $\text{diag}(T_Y, T_Y, \dots, T_Y)$ is used instead of T_Y - to obtain

$$L \otimes (T_Y^T P T_Y) + M \otimes (T_X^T A_c T_Y)^T + M^T \otimes (T_X^T A_c T_Y) < 0 \tag{19.16}$$

This inequality is linear in the variables $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$, Y and X. It is equivalent to (19.11), therefore the existence of these matrices satisfying (19.16) is a necessary condition for A_c to have its poles in $\mathcal{D}$. On the other hand, having found solutions $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$, Y and X to the LMI (19.16), one can compute via singular value decomposition nonsingular S and R satisfying

$$RS^T = I - XY$$

and solve (19.14) for D_K, C_K, B_K and A_K. Note that the constraint

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

guarantees that $I - XY < 0$ and therefore that nonsingular matrices S and R satisfying the above exist. The resulting controller has the property that the closed-loop eigenvalues are

in $\mathcal{D}$. Thus, the existence of matrix variables satisfying (19.15) and (19.16) is a necessary and sufficient condition for the closed-loop system to have its poles in $\mathcal{D}$.

In summary, a pole region constraint in the form of a matrix inequality with variable terms P and PA_c can be transformed by a congruence transformation T_Y into an equivalent constraint with the terms $T_Y^T P T_Y$ and $T_Y^T P A_c T_Y$. The resulting inequality is linear in transformed controller matrices and auxiliary variables. Having solved this LMI, one can compute the desired controller matrices from (19.14).

H_2 and H_∞ Constraints

The same transformation can be used to make the H_2 constraint (18.19) and the H_∞ constraint (18.22) linear in the transformed controller variables. In addition to the terms P and PA_c, the constraints (18.19) and (18.22) applied to the closed-loop system involve the terms PB_c and $C_c D_c$, which are transformed into $T_Y^T P B_c$ and $C_c D_c T_Y$, respectively. These terms are again linear in the new controller variables (19.14) and the auxiliary variables X and Y.

List of LMI Constraints

The constraint $P > 0$ is shared by pole region, H_2 and H_∞ constraints; with a congruence transformation T_Y, it is transformed into

$$T_Y^T P T_Y = \begin{bmatrix} Y & I \\ I & X \end{bmatrix} > 0 \tag{19.17}$$

The following conditions refer to the configuration shown in Figure 19.1 with plant model (19.1) and controller (19.2). The controller has the same dynamic order as the plant. The transformed controller matrices $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$ are defined by (19.14). $(*)$ denotes matrix blocks that can be inferred by symmetry.

Pole Region Constraint

The closed-loop matrix A_c has all eigenvalues in $\mathcal{D}$ if and only if there exist matrices $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$, X and Y that satisfy (19.17) and

$$L \otimes \begin{bmatrix} Y & I \\ I & X \end{bmatrix} + M \otimes \begin{bmatrix} AY + B_u\tilde{C}_K & A + B_u\tilde{D}_KC_v \\ \tilde{A}_K & XA + \tilde{B}_KC_v \end{bmatrix} + M^T \otimes \begin{bmatrix} AY + B_u\tilde{C}_K & A + B_u\tilde{D}_KC_v \\ \tilde{A}_K & XA + \tilde{B}_KC_v \end{bmatrix}^T < 0 \qquad (19.18)$$

H_2 Constraints

$\|T\|_2 < \nu$ if and only if there exist matrices $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$, X, Y and W that satisfy (19.17) and

$$\begin{bmatrix} AY + YA^T + B_u\tilde{C}_K + (B_u\tilde{C}_K)^T & \tilde{A}_K^T + A + B_u\tilde{D}_KC_v & (C_zY + D_{zu}\tilde{C}_K)^T \\ * & A^TX + XA + \tilde{B}_KC_v + (\tilde{B}_KC_v)^T & (C_z + D_{zu}\tilde{D}_KC_v)^T \\ * & * & -I \end{bmatrix} < 0 \qquad (19.19)$$

$$\begin{bmatrix} Y & I & B_w + B_u\tilde{D}_KD_{vw} \\ * & X & XB_w + \tilde{B}_KD_{vw} \\ * & * & W \end{bmatrix} > 0 \qquad (19.20)$$

and $\quad \text{trace}\, W < \nu^2 \quad$ and $\quad D_c = D_{zw} + D_{zu}\tilde{D}_KD_{vw} = 0$

H_∞ Constraints

$\|T\|_\infty < \gamma$ if and only if there exist matrices $\tilde{A}_K$, $\tilde{B}_K$, $\tilde{C}_K$, $\tilde{D}_K$, X and Y that satisfy (19.17) and

$$\begin{bmatrix} AY + YA^T + B_u\tilde{C}_K + (B_u\tilde{C}_K)^T & \tilde{A}_K^T + A + B_u\tilde{D}_KC_v & B_w + B_u\tilde{D}_KD_{vw} & (C_zY + D_{zu}\tilde{C}_K)^T \\ * & A^TX + XA + \tilde{B}_KC_v + (\tilde{B}_KC_v)^T & XB_w + \tilde{B}_KD_{vw} & (C_z + D_{zu}\tilde{D}_KC_v)^T \\ * & * & -\gamma I & (D_{zw} + D_{zu}\tilde{D}_KD_{vw})^T \\ * & * & * & -\gamma I \end{bmatrix} < 0 \qquad (19.21)$$

Design Algorithm

The design process can be summarised in the following algorithm.

1. Specify the design objective and generate the required matrices

    a) Pole region constraint - express the desired pole-region in terms of the matrices L and M (see for example (18.8), (18.9) and (18.10)).

    b) H_2 performance - set $B_u = B$, $C_v = -C$ and construct the generalised plant by appropriately selecting the matrices B_w, C_z, D_{zu} and D_{vw} (see for example chapter 11).

    c) H_∞ performance - choose weighting filters $W_S(s)$, $W_K(s)$ or/and $W_T(s)$ and construct the generalised plant (see chapter 16). Extract the matrices A, B_w, B_u, C_z, C_v, D_{zw}, D_{zu}, D_{vw}.

2. Express the design specifications as an LMI problem. The following table summarises the relevant LMI constraints

| Design objective | Feedback type | | LMI objective |
	state-feedback	output feedback	
with any	$P = P^T$ $P > 0$	$X = X^T,\ Y = Y^T$ $\begin{bmatrix} Y & I \\ I & X \end{bmatrix} > 0$	
Pole region constraint	(19.5)	(19.18)	feasibility
H_2 performance	(19.7), (19.8)	(19.19), (19.20)	$\min \operatorname{trace} W$
H_∞ performance	(19.10)	(19.21)	$\min \gamma$

3. Construct the controller

    *   State-feedback case: $F = YP^{-1}$

    *   Output-feedback case: perform singular value decomposition of $I - XY$

    $$U\Sigma V^T = I - XY \quad \Rightarrow \quad R = U\sqrt{\Sigma}, \quad S^T = \sqrt{\Sigma}V^T$$

    Then use the inverse transformation of (19.14):

    $$D_K = \tilde{D}_K$$
    $$C_K = (\tilde{C}_K - D_K C_v Y)(S^T)^{-1}$$
    $$B_K = R^{-1}(\tilde{B}_K - XB_u D_K)$$
    $$A_K = R^{-1}\left(\tilde{A}_K - RB_K C_v Y - XB_u C_K S^T - X(A + B_u D_K C_v)Y\right)(S^T)^{-1}$$

Exercises

Problem 19.1

Design an LQG controller for the aircraft model in Example 11.1 using LMIs and the weighting matrices from Problem 15.2 [Design 2]

$$C_z = 50 \cdot \begin{bmatrix} C \\ 0 \end{bmatrix} \qquad D_{zu} = \begin{bmatrix} 0 \\ I_3 \end{bmatrix}; \qquad B_w = 50 \cdot \begin{bmatrix} B & 0 \end{bmatrix}; \qquad D_{vw} = \begin{bmatrix} 0 & I_3 \end{bmatrix}.$$

Hint: Use that the LQG problem can be expressed as an H_2 design problem, when the generalised plant in Fig. 11.3 is used. See the design algorithm on page 141 for details.

Compare the obtained controller with the one obtained using the `h2syn` function.

Problem 19.2

Consider the state space model

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1.25 & 1.25 & 0 & 0 \\ 1.25 & -1.25 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \end{bmatrix} w$$

$$z = \begin{bmatrix} 0.75 & -0.75 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

a) Design a state feedback controller that minimises the H_∞ norm (i.e. $\gamma = \|T_{zw}\|_\infty$) of the closed loop transfer function from input w to output z, and assigns the closed-loop poles to the LMI region $\mathcal{D}$ with $\alpha_l = -\infty$, $\alpha_r = -10$ and $\beta = 45°$.

b) For $\alpha_r = [-1, -2 - 5, -10, -20, -30, -50, -100]$, calculate the H_∞ norm and plot γ versus α_r.

Chapter 20

LMI Approach to Multi-Objective Design

So far it has been shown how the three types of design specifications introduced in Chapter 18, can be expressed as LMI constraints on the synthesis of a dynamic controller (19.2). The LMI constraints involve transformed controller matrices and auxiliary variables. Let $\tilde{K}$ denote the collection of matrices

$$\tilde{K} = (\tilde{A}_K, \tilde{B}_K, \tilde{C}_K, \tilde{D}_K, X, Y)$$

These variables in the LMI constraints provide the data from which the controller can be computed. The three types of design specifications considered here are

- the pole region constraint (18.7) which was shown to be equivalent to (19.16), an LMI of the form
$$M_{reg}(\tilde{K}) < 0$$

- the constraint (18.19) on the H_2-norm in the form of an upper bound ν, which can be transformed into an LMI of the form
$$M_2(\tilde{K}, W) < 0$$
where W is used as a slack variable and satisfies $\mathrm{trace}\, W < \nu^2$

- the constraint (18.22) on the H_∞-norm in the form of an upper bound γ, which can be transformed into
$$M_\infty(\tilde{K}, \gamma) < 0$$

Explicit formulae for $M_{reg}(\tilde{K})$, $M_2(\tilde{K}, W)$ and $M_\infty(\tilde{K}, \gamma)$ were given in the previous chapter. The controller design can be carried out by solving a convex optimization problem of the form

$$\min_p c^T p \qquad \text{subject to} \qquad M(p) < 0$$

where $M(p)$ is a combination of the above constraints. The decision variables - the elements of the vector p - are the elements of the matrix variables $\tilde{K}$ and auxiliary variables. The linear cost function $c^T p$ can be chosen to represent the search for the "best" controller that satisfies the given specifications.

Mixed H_2/H_∞ Design

To combine H_2 and H_∞ norm constraints, one can partition the input and output vectors $w(t)$ and $z(t)$ as

$$w = \begin{bmatrix} w_\infty \\ w_2 \end{bmatrix}, \qquad z = \begin{bmatrix} z_\infty \\ z_2 \end{bmatrix} \tag{20.1}$$

Figure 20.1 shows the configuration with separate channels for H_2 and H_∞ constraints. A possible choice for a generalized plant is then

$$P(s) = \begin{bmatrix} A & \begin{bmatrix} B_w^\infty & B_w^2 \end{bmatrix} & B_u \\ \begin{bmatrix} C_z^\infty \\ C_z^2 \end{bmatrix} & 0 & \begin{bmatrix} D_{zu}^\infty \\ D_{zu}^2 \end{bmatrix} \\ C_v & \begin{bmatrix} D_{vw}^\infty & D_{vw}^2 \end{bmatrix} & 0 \end{bmatrix}$$
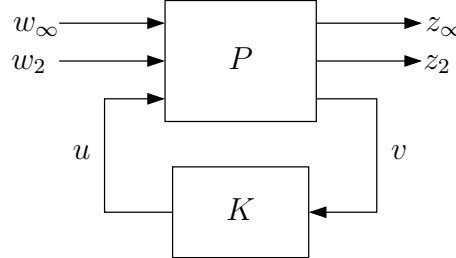

Figure 20.1: Generalized plant for mixed H_2/H_∞ design

Let $T_2(s)$ denote the transfer function from w_2 to z_2 and $T_\infty(s)$ from w_∞ to z_∞, then the H_2 constraint should be applied to $T_2(s)$ and the H_∞ constraint to $T_\infty(s)$. For example, one might want to design a controller for a plant with uncertain parameters such that the closed-loop poles of the nominal system are constrained to be in a given pole region, and where an additional H_∞ constraint (say $\|T_\infty\|_\infty < \gamma$ where $\gamma = 1$) guarantees robust stability for the admissible range of system parameters. One can then search for a controller that minimizes a quadratic performance index (H_2-norm of $T_2(s)$) under these constraints by solving the convex optimization problem (18.2) in the form

$$\min_{\tilde{K},W} \operatorname{trace} W \qquad \text{subject to} \qquad \begin{bmatrix} M_{reg}(\tilde{K}) & 0 & 0 \\ 0 & M_2(\tilde{K},W) & 0 \\ 0 & 0 & M_\infty(\tilde{K},\gamma=1) \end{bmatrix} < 0 \tag{20.2}$$

From the solution $\tilde{K}$ to this problem one can compute a controller that satisfies the pole region and H_∞ constraints as well as $\|T_2\|_2^2 < \operatorname{trace} W$. A design example of this type is presented in the next chapter to illustrate the approach.

Alternatively, one can minimize the H_∞ norm subject to the H_2 norm being less than a specified upper bound ν, in this case the minimization is over $\tilde{K}$, W and γ as variables subject to $\operatorname{trace} W < \nu^2$. Similarly, it is possible to minimize a linear combination of the H_2-norm and the H_∞-norm.

Existence of Solutions and Conservatism of Design

An important feature of the LMI approach to controller design is that it allows to combine different design specifications into a single convex optimization problem. However, this advantage comes at a price: combining different constraints may introduce conservatism into the design. Even though each of the LMI constraints considered here represents a necessary and sufficient condition for a certain property of the system, combining them into a single convex optimization problem is only possible by relaxing the conditions to sufficient conditions, i.e. by giving up necessity. This aspect is now briefly discussed.

When the plant is controllable and observable, it is clear that there exist controllers that place the closed-loop eigenvalues in a given region and at the same time achieve finite values for the H_2 and H_∞-norm. On the other hand, it may turn out that the constraints for the problem

$$\min_{\tilde{K},W,\nu,\gamma} c_1\nu + c_2\gamma \qquad \text{subject to} \qquad \begin{bmatrix} M_{reg}(\tilde{K}) & 0 & 0 \\ 0 & M_2(\tilde{K},W) & 0 \\ 0 & 0 & M_\infty(\tilde{K},\gamma) \end{bmatrix} < 0 \qquad (20.3)$$

and $\operatorname{trace} W < \nu^2$ are infeasible, i.e. that a combination of decision variables that satisfy the constraints does not exist.

To see why this may happen, recall that the pole region constraint involves the term $A_c P_{reg}$, whereas the H_2 constraint involves the terms $A_c P_2$ and $C_c P_2$, and the H_∞ constraint involves $P_\infty A_c$ and $P_\infty B_c$. Here different subscripts for the matrix variable P have been introduced to indicate in which constraint it is used. A *necessary and sufficient* condition for the closed-loop system to satisfy all constraints simultaneously is the existence of three matrices P_{reg}, P_2 and P_∞ each of which satisfies the corresponding LMI constraint. However, the linearizing change of variables - and thus the synthesis of a controller - is only possible by imposing the additional constraint

$$P = P_{reg} = P_2 = P_\infty \qquad (20.4)$$

Whereas the search for a controller under just one of the three constraints is a search over *all* controllers that satisfy this constraint, the assumption (20.4) reduces the search under combined constraints to a search over only a subset of all controllers that satisfy the combined constraints.

Chapter 21

Design Example: Robust Control of a Power System Stabilizer

In this chapter it is shown how a combination of LMI constraints can be used to design a robust controller for system with uncertain parameters. The design of a power system stabilizer for a range of operating conditions is presented as a design example. First a model is constructed that includes information about the parameter uncertainty. The small gain theorem is then introduced as an important tool for robust controller design. With the help of the small gain theorem, a H_∞ constraint on the uncertain model is derived that guarantees robust stability. This H_∞ constraint is then combined with pole region and H_2 constraints for shaping the nominal performance.
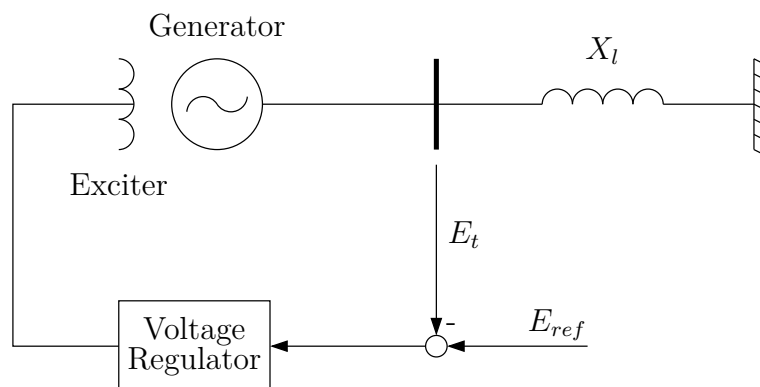
Figure 21.1: Single machine on infinite busbar

Problem Description

Power system stabilizers (PSS) are used to suppress oscillation in power systems. A power station connected to a large power system is often represented by a single equivalent ma-

chine connected through an impedance to an infinite busbar for the purpose of stability and controller design studies. Such a model is shown in Figure 21.1. In this representation, the dynamic interaction between the various machines in the power station is not considered, but it is still adequate for many types of studies, especially when the machines are identical and operate at nearly the same load levels.
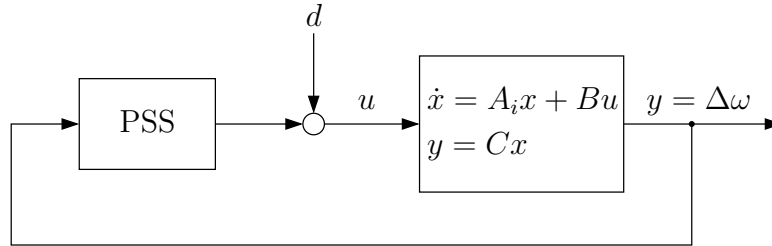

Figure 21.2: PSS control loop

Power systems constantly experience changes in operating conditions due to variations in generation and load patterns, as well as changes in transmission networks. There is a corresponding large variation in the small signal dynamic behavior of a power system. This can be expressed as a parametric uncertainty in the small signal linearized model of the system. The problem considered here is the robust design of PSS so that adequate damping can be provided, while guaranteeing stability over a wide range of operating conditions. In this simulation study a linearized state space model

$$\dot{x}(t) = A(P, Q, X_l)\, x(t) + Bu(t)$$
$$y(t) = Cx(t)$$

with four state variables is used to model the single-machine-infinite-bus system. The system matrix A in this model is a function of real power P, reactive power Q and line impedance X_l. It is assumed that these parameters vary in the intervals

$$0.4 \le P \le 1.0, \qquad -0.2 \le Q \le 0.5, \qquad 0.2 \le X_l \le 0.7$$

Varying the parameters independently in steps of 0.1 over these intervals generates a family of linearized models

$$\dot{x} = A_i x + Bu, \qquad y = Cx, \qquad i = 0, \ldots, N \tag{21.1}$$

where the system matrices A_i describe the system dynamics in different operating conditions and $N = 336$. The distribution of open-loop eigenvalues of the family of 336 linear systems is shown in Figure 21.3.

The PSS control loop is shown in Figure 21.2. The controlled output is the machine shaft speed $\Delta\omega$, and the control input u is a voltage which is added to the terminal reference voltage. The performance of the controller is assessed in terms of rejection of a disturbance step d at plant input.
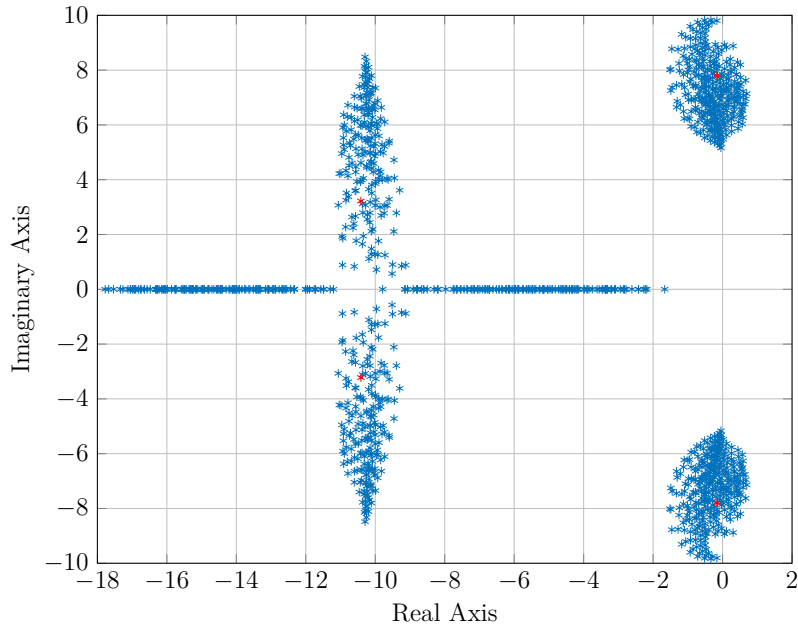
Figure 21.3: Open-loop eigenvalues for the given range of uncertain parameters (red: nominal plant)

The objective is to design a controller that maintains a damping ratio of at least 10% and a real part of all closed-loop poles less than -0.5 at nominal operating conditions, and at the same time guarantees stability for all admissible operating conditions.

In this chapter it is shown how the LMI constraints introduced in the previous chapter can be used to design a dynamic output feedback controller of the form (19.2) that meets the specifications. A H_∞ constraint will be used to guarantee robust stability; for this purpose a suitable representation of the parameter uncertainty is required.

Representing Parameter Uncertainty

The available information about parameter variation can be included in a generalized plant model in the following way. Replace B in (21.1) by B_u and C by C_y, let A_0 denote the system matrix under nominal operating conditions. Augment the system with an additional input vector w and output vector z to obtain

$$\dot{x} = A_0 x + B_w w + B_u u, \qquad z = C_z x \tag{21.2}$$

Now introduce feedback

$$w = \Delta z$$

through a gain matrix Δ that is unknown, but satisfies $\|\Delta\| \leq 1$. The dynamics of the system (21.2) are then governed by

$$\dot{x} = (A_0 + B_w \Delta C_z)x + B_u u \tag{21.3}$$

and with a suitable choice of the matrices B_w and C_z, all models representing the uncertain system (21.1) can be "covered" by the model (21.2) in the sense that any admissible system matrix

$$A_i = A_0 + B_w \Delta C_z$$

can be generated by choosing a matrix Δ with norm less than 1. A block diagram of this uncertainty representation is shown in Figure 21.4. The uncertainty has been "isolated" by mapping it into the matrix Δ - note that once B_w and C_z have been fixed, all parameters of the model inside the dashed box are known. The matrices B_w and C_z - together with the condition that $\|\Delta\| < 1$ - represent *a priori* knowledge about the possible range of parameter variations.

The construction of suitable matrices B_w and C_z will be discussed below.
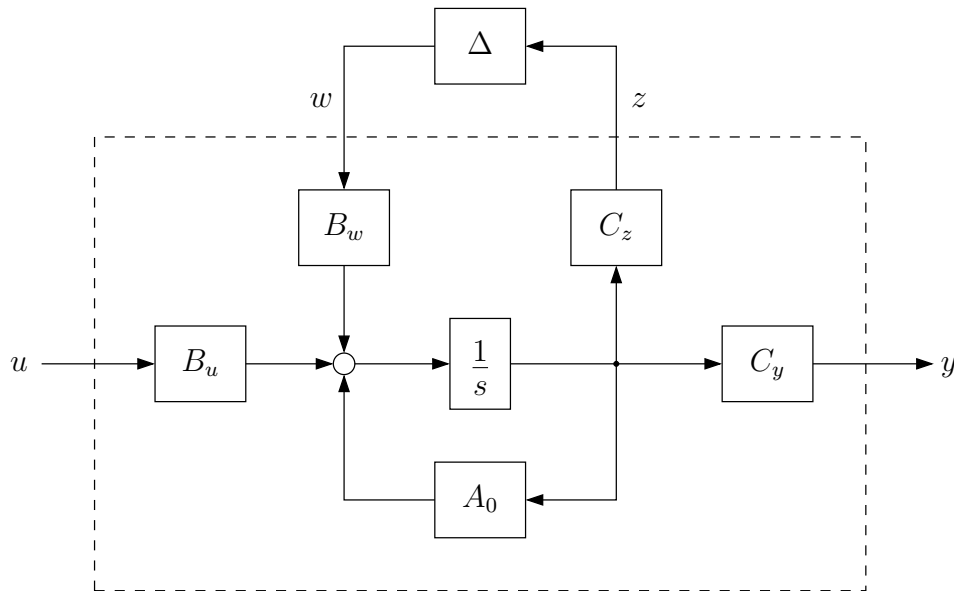
Figure 21.4: Representation of model uncertainty

The Small Gain Theorem

The Power System Stabilizer must - apart from providing adequate damping and response times - guarantee that the closed-loop system is stable for all admissible operating conditions. If we use the representation in Figure 21.4 and map the parameter uncertainty into an unknown matrix Δ, we therefore need a condition for the system to be stable with all Δ that satisfy $\|\Delta\| < 1$. Such a condition is provided by the following Theorem. Consider the loop in Figure 21.5. We have
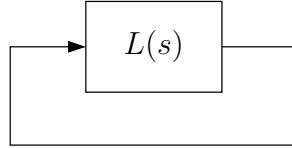
Figure 21.5: Loop for small gain theorem

Theorem 21.1 *Assume $L(s)$ is stable. Then the loop is stable if $\|L(j\omega)\| < 1$ for all ω.*

This Theorem is a simple version of the *Small Gain Theorem*. Most methods for robust controller design are based on one or another version of this Theorem. For SISO systems it simply says that if the Nyquist plot of $L(s)$ does not leave the unit disc, the closed-loop system is stable (because $L(j\omega)$ cannot encircle the critical point -1). For MIMO systems, the small gain condition $\|L(j\omega)\| < 1$ implies $\bar{\sigma}(L(j\omega)) < 1$ for all ω. As a consequence, $\det(I - L(j\omega))$ cannot encircle the origin and it follows from the generalized Nyquist Theorem that the closed-loop system is stable.

Generalized Plant and Model Uncertainty

We will now express the design problem in the form of a generalized plant that includes not only weighting functions, but also the representation of the model uncertainty. The configuration is shown in Figure 21.6, where the uncertainty representation in Figure 21.4 has been combined with the generalized plant in Figure 20.1 by connecting the uncertain block Δ to the H_∞ channels. The generalized plant transfer function is then

$$
P(s) = \left[
\begin{array}{c|cc|c}
A & \multicolumn{2}{c|}{\left[\begin{array}{cc} B_w^\infty & B_w^2 \end{array}\right]} & B_u \\
\hline
\left[\begin{array}{c} C_z^\infty \\ C_z^2 \end{array}\right] & \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} & & \left[\begin{array}{c} 0 \\ D_{zu}^2 \end{array}\right] \\
\hline
C_v & \multicolumn{2}{c|}{\left[\begin{array}{cc} 0 & D_{vw}^2 \end{array}\right]} & 0
\end{array}
\right]
\tag{21.4}
$$

To obtain a condition on robust closed-loop stability, we consider the loop formed by Δ and the system inside the dashed box in Figure 21.6. A simplified block diagram
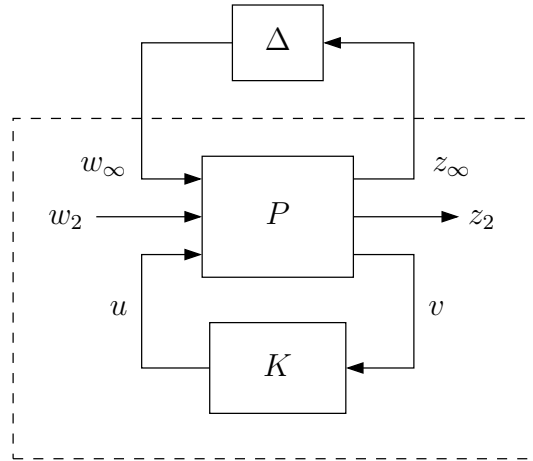
Figure 21.6: Generalized plant

is shown in Figure 21.7. Assuming that the controller K stabilizes the nominal plant model (A_0, B_u, C_y), we can invoke the small gain theorem to establish that the closed-loop system is stable if the magnitude of the loop transfer function ΔT_{zw}^∞ is less than one at all frequencies, or if

$$\|L(j\omega)\| = \|\Delta T_{zw}^\infty(j\omega)\| \leq \|\Delta\| \|T_{zw}^\infty(j\omega)\| < 1 \qquad \forall \omega$$

It follows that the closed-loop system is stable for all Δ satisfying $\|\Delta\| < 1$ if

$$\|T_{zw}^\infty(j\omega)\|_\infty < 1$$
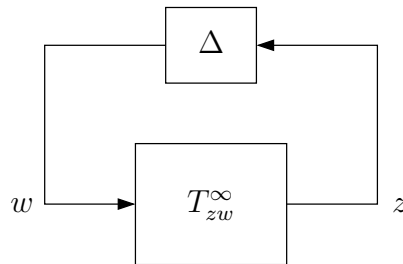
Figure 21.7: Configuration for robust closed-loop stability

The following problem can now be posed in the form of linear matrix inequalities and solved with LMI solvers: Design a controller that minimizes the H_2-norm of the transfer function from w_2 to z_2 (pick the best controller in terms of a quadratic performance index) under the constraints that the H_∞-norm of the transfer function from w_∞ to z_∞ is less than 1 (guarantee robust stability), and that the closed-loop eigenvalues of the nominal system are located within a specified region in the complex plane (meet specs on damping ratio and speed of response).

Constructing the Uncertainty Model

To apply the design procedure described above, the model uncertainty - represented by a family of 336 linear models - must be expressed in terms of the matrices B_w^∞ and C_z^∞. For this purpose, write the uncertain system matrix A_i at a given operating point as

$$A_i = A_0 + \tilde{A}_i, \qquad \tilde{A}_i = B_w^\infty \Delta_i C_z^\infty \tag{21.5}$$

The matrices B_w^∞ and C_z^∞ must be chosen such that a matrix Δ_i satisfying

$$\tilde{A}_i = B_w^\infty \Delta_i C_z^\infty \qquad \text{and} \qquad \|\Delta_i\| \le 1$$

exists for all admissible operating conditions $(i = 1, \dots, N)$. The first requirement is equivalent to

$$\tilde{A}_i \in \mathcal{R}\,(B_w^\infty), \qquad \tilde{A}_i^T \in \mathcal{R}\,(C_z^{\infty T}), \qquad i = 1, \dots, N$$

where $\mathcal{R}(M)$ denotes the column space of a matrix M. Define

$$\Psi_B = \begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 & \dots & \tilde{A}_N \end{bmatrix}, \qquad \Psi_C = \begin{bmatrix} \tilde{A}_1^T & \tilde{A}_2^T & \dots & \tilde{A}_N^T \end{bmatrix}$$

and introduce the singular value decomposition of these two matrices

$$\Psi_B = U_B \Sigma_B V_B^T, \qquad \Psi_C = U_C \Sigma_C V_C^T$$

Let $r_B = \operatorname{rank} \Psi_B$ and $r_C = \operatorname{rank} \Psi_C$, then matrices Δ_i satisfying (21.5) exist for $i = 1, \dots, N$ if $B_w^\infty = \tilde{B}_w$ and $C_z^\infty = \tilde{C}_z$ where

$$\tilde{B}_w = [u_1^b \; u_2^b \; \dots u_{r_B}^b], \qquad \tilde{C}_z = [u_1^c \; u_2^c \; \dots u_{r_C}^c]^T$$

and u_i^b and u_i^c are the columns of U_B and U_C, respectively. The non-zero elements of the matrices

$$\Sigma_B = \operatorname{diag}(\sigma_1^b, \sigma_2^b, \dots), \qquad \Sigma_C = \operatorname{diag}(\sigma_1^c, \sigma_2^c, \dots)$$

are the singular values of Ψ_B and Ψ_C ordered by decreasing magnitude; small singular values indicate that the dimension of the uncertainty (r_B or r_C) can be reduced.

To meet the requirement $\|\Delta_i\| \le 1$, $i = 1, \dots, N$, the columns of B_w^∞ and the rows of C_z^∞ need to be scaled. This can be done as follows. Let

$$\bar{\Sigma}_B = \operatorname{diag}(\sigma_1^b, \dots, \sigma_{r_B}^b), \qquad \bar{\Sigma}_C = \operatorname{diag}(\sigma_1^c, \dots, \sigma_{r_C}^c)$$

and take $B_w' = \tilde{B}_w \bar{\Sigma}_B$, $C_z' = \bar{\Sigma}_C \tilde{C}_z$ as a first scaling. In (21.5), replace B_w^∞ and C_z^∞ with B_w' and C_z', respectively, solve for Δ_i in all operating points and find $\kappa = \max_i \|\Delta_i\|$. Then matrices Δ_i that satisfy (21.5) and $\|\Delta_i\| \le 1$ exist for all operating points if B_w^∞ and C_z^∞ are chosen as

$$B_w^\infty = \sqrt{\kappa} B_w', \qquad C_z^\infty = \sqrt{\kappa} C_z'$$

In order to construct the uncertain model, a nominal system matrix A_0 must be chosen and the matrices B_w^∞ and C_z^∞ must be computed.

Here, the nominal system matrix is chosen as

$$A_0 = \frac{1}{2}(A_{\max} + A_{\min})$$

where $A_{\max}$ and $A_{\min}$ are matrices whose elements are the maximum and minimum values of the corresponding elements of A_i over $i = 1, \ldots, N$, respectively.

Computing the singular value decomposition of Ψ_B and Ψ_C leads to

$$\bar\Sigma_B = \mathrm{diag}\,(2163.5,\ 449.3,\ 0,\ 0), \qquad \bar\Sigma_C = \mathrm{diag}\,(2209.6,\ 14.5,\ 2.6,\ 0.4)$$

Thus $r_B = 2$, and the drop by more than two orders of magnitude after the first singular value of Ψ_C suggests that without significant loss of information the uncertainty can be projected onto one dimension, i.e. the numerical rank of Ψ_C is taken as $r_C = 1$. Carrying out the scaling procedure described above yields

$$B_w^\infty = \begin{bmatrix} 0 & 0 \\ 0.130 & 3.676 \\ 0 & 0 \\ 17.701 & -0.027 \end{bmatrix}, \qquad C_z^\infty = \begin{bmatrix} 18.079 & 0.095 & 0.014 & -0.024 \end{bmatrix}, \qquad \Delta_i = \begin{bmatrix} \delta_1^i \\ \delta_2^i \end{bmatrix}$$

To assess the conservatism of this uncertainty representation, (21.5) is solved for Δ_i in all operating points under consideration, and the result is plotted in Figure 21.8. Stability is guaranteed inside the unit disc, and the fact that it is largely filled shows that the representation is not too conservative.

The remaining system matrices are

$$A_0 = \begin{bmatrix} 0 & 0 & 0 & -0.154 \\ 4.086 & -0.491 & -0.099 & 0.219 \\ 0 & 1000 & -20 & 0 \\ 417.870 & 1.454 & 0.346 & -0.665 \end{bmatrix}, \qquad B_u = \begin{bmatrix} 0 \\ 0 \\ 1000 \\ 0 \end{bmatrix}$$

and $C_y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$.

LMI-Based Controller Design

With A_0, B_w^∞, B_u, C_z^∞ and C_y as above, the robust design problem can be solved as an LMI problem in the form of (20.2). Design parameters are (a) the weights for the H_2 constraint and (b) the parameters $(\alpha_l, \alpha_r, \beta)$ of the LMI region in Figure 18.3 for the closed-loop poles of the nominal system.

We will treat this problem as a deterministic design problem. This means that in the H_2 performance index no noise inputs are assumed and the input channel w_2 is omitted;
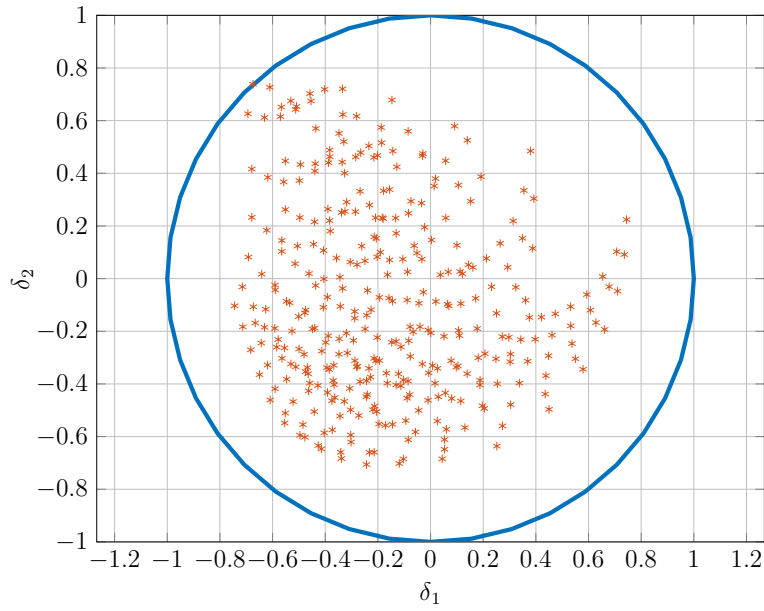
Figure 21.8: Uncertainty representation after scaling

therefore the matrices B_w^2 and D_{vw}^2 in (21.4) are removed from the model. The remaining H_2 weights are chosen as

$$C_z^2 = \begin{bmatrix} C_y \\ 0 \end{bmatrix}, \qquad D_{zu}^2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

This H_2 constraint can be interpreted in terms of linear quadratic optimal control: the 2-norm of $z_2(t)$ is given by

$$\|z_2(t)\|_2^2 = \int_0^\infty (x^T C_y^T C_y x + u^T u)\, dt = \int_0^\infty (y^T y + u^T u)\, dt$$

With the choice $\alpha_l = -13$, $\alpha_r = -0.5$ and $\beta = 6^o$ for the pole region, the solution to the LMI problem is a controller

$$K(s) = k \frac{(1 + sT_1)(1 + sT_2)(1 + sT_3)}{(1 + sT_4)(1 + sT_5)(s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

with values given in the table below.

k	T_1	T_2	T_3	T_4	T_5	ζ	ω_n
4.014	2.032	0.203	0.084	0.503	0.077	0.14	72.6

The closed-loop pole distribution is shown in Figure 21.9, and Figure 21.10 shows the step response at three different operating points.

The pole region parameters α_l, α_r and β can be used to tune the performance of the control system. For example, moving the right boundary α_r of the pole region to the left leads to a smaller peak of the regulation error in response to a disturbance step, and to a larger peak of the control signal. In this way, the pole region parameter α_r can be used as a "tuning knob" to trade control effort against peak regulation error.
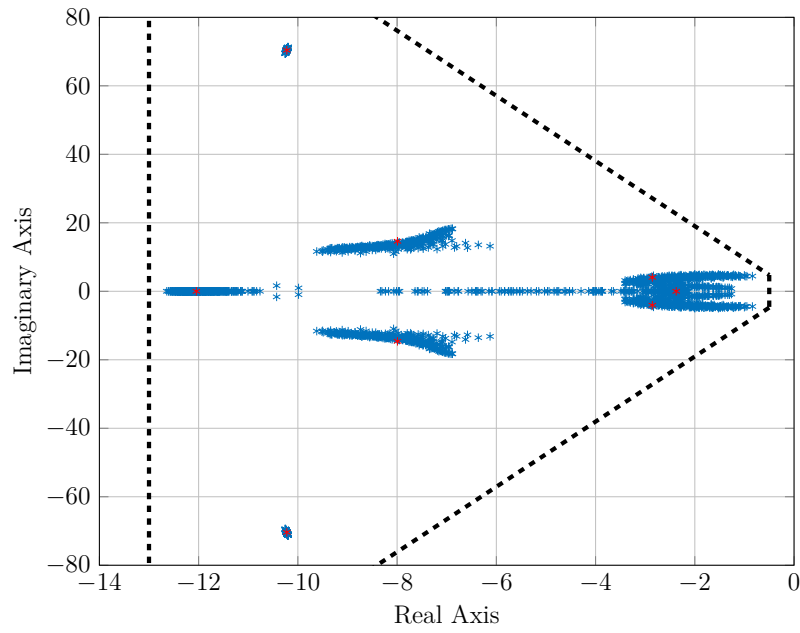
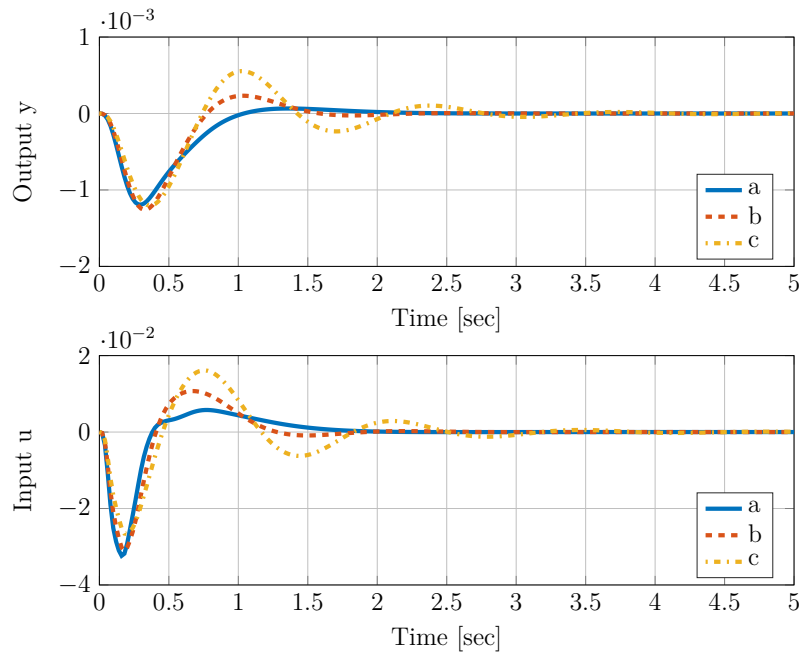Figure 21.9: Closed-loop eigenvalues (dashed: LMI region, red: nominal plant)

Figure 21.10: System response to a 5% disturbance step at different operating conditions. (a) $P = 0.8$, $Q = 0.4$, $X = 0.2$, (b) $P = 0.8$, $Q = 0.0$, $X = 0.6$, (c) $P = 1.0$, $Q = 0.54$, $X = 0.7$

User-Friendly Design Procedure

The design approach presented in this chapter can be carried out as an automated, interactive procedure with a simple user interface. The data input consists of a family of

linear models - which represent parameter uncertainty - and the design specifications, which are expressed in terms of pole region parameters α_l, α_r and β. Practical experience suggests that one can fix the weights in the H_2 cost function and use only the pole region for tuning. The H_∞ constraint guarantees that for any choice of pole region parameters the system is stable in all admissible conditions. This fact may be used to develop an automated search strategy for tuning the controller.

Exercises

Problem 21.1   ACC Benchmark Problem

Consider the system shown in Figure 21.11. Two bodies with masses m_1 and m_2 are connected by a spring with stiffness k. The position of the second body is to be controlled (desired position $y = 0$). The controller generates a force acting on the first body, and only the position $y(t)$ of the second body is available for feedback. A state space model of the system is

$$
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k/m_1 & k/m_1 & 0 & 0 \\ k/m_2 & -k/m_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/m_1 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/m_1 & 0 \\ 0 & 1/m_2 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}
$$

$$
y = x_2 \tag{21.6}
$$

Numerical values of the nominal system are $m_1 = m_2 = 1$ and $k = 1$. However, the stiffness of the spring can take values in the range $0.5 \le k \le 2.0$. A controller is to be designed that brings the second body back to $|y(t)| < 0.1 \quad \forall \quad t > 15$ after a unit disturbance impulse $d_1(t) = \delta(t)$ or $d_2(t) = \delta(t)$. The control input must satisfy $|u(t)| \le 1$ at all times.
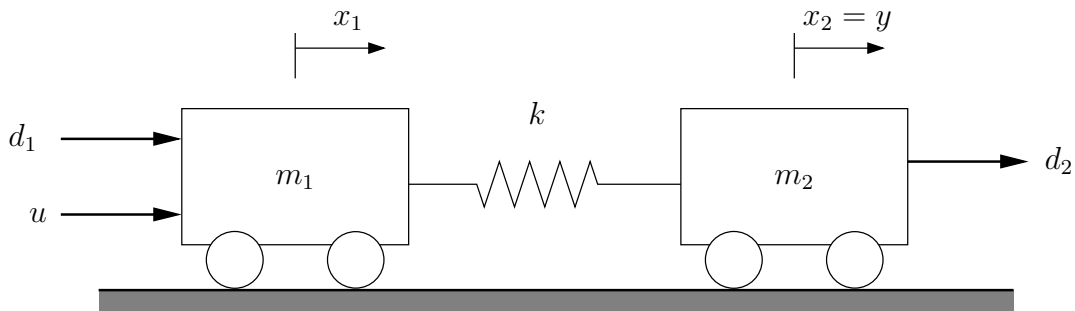

Figure 21.11: Two-mass-spring system

a)   For $k_i = 0.5$, 0.8, 1.1, 1.4, 1.7, 2.0, compute the system matrices A_i, and construct

the matrices (B_w, C_z) for an uncertainty model. Also, derive these matrices directly from the physical model (21.6) and compare the results.

b) Design a controller that guarantees stability for $0.5 \leq k \leq 2.0$, while satisfying the constraints on actuator and settling time.

c) Modify the controller such that the maximum possible range $k_{\min} \leq k \leq k_{\max}$ for robust stability is achieved. The actuator constraint must still be satisfied, but the settling time can be longer than 15.

d) Analyze the controller in frequency domain. Why is it not reasonable to choose this controller for solving the control task? Changing the setup to a four-block design improves the frequency response of the controller. Design a controller using the four-block approach and give a reason for the improvement.

Chapter 22

Model Order Reduction

Modern design techniques - like those discussed in this course - produce controllers of the same dynamic order as the generalized plant. Thus, when the system to be controlled is of high dynamic order, the controller may be too complex to be acceptable for implementation. This is an important issue when LMI techniques are used for the design of controllers, because solving a high-order LMI problem takes more computation time than solving the Riccati equations associated with the same problem. In such cases, the plant model should be approximated by a simplified model. This chapter gives a brief introduction to the topic of model order reduction.

Consider a stable system with transfer function $G(s)$ and state space realization

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

with n dimensional state vector x. If the number of states n is very large, one could try to find a model of lower order that behaves "similar" to the original system. For example, if some of the state variables do not have much effect on the system behaviour, one might consider removing these states from the model. Thus, we need to know which states are "important" for the model and which ones are not. The controllability Gramian and the observability Gramian - discussed in Chapter 5 - turn out to be helpful for answering this question.

Controllability Gramian

Recall the definition of the controllability Gramian

$$W_c = \int_0^\infty e^{At} BB^T e^{A^T t} \, dt$$

A useful geometrical interpretation of W_c - given here without proof - is the following. Define the set S_c as the set of all points in the state space to which the state vector $x(t)$ of the system can be driven from the origin with an input signal $u(\tau)$, $0 \leq \tau \leq t$, that

satisfies $\|u(\tau)\|_2 \leq 1$ (assume that $u(\tau) = 0$ for $\tau > t$). Compute the singular value decomposition of the controllability Gramian

$$W_c = V \Sigma V^T$$

where

$$\Sigma = \text{diag}\left(\sigma_1,\ \sigma_2, \ldots \sigma_n\right) \qquad \text{and} \qquad V = \begin{bmatrix} v_1 & v_2 & \ldots & v_n \end{bmatrix}$$

The singular values are ordered such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$ as usual. The set S_c of points in the state space reachable with input energy 1 is then a hyper-ellipsoid centered at the origin, with semi-axes in the directions of the columns v_i of V, and length given by the square root of the corresponding singular value. This is illustrated in Figure 22.1 for a second order system. Note that v_1 is the direction that is most easily controlled, whereas small singular values indicate directions which are difficult to control.
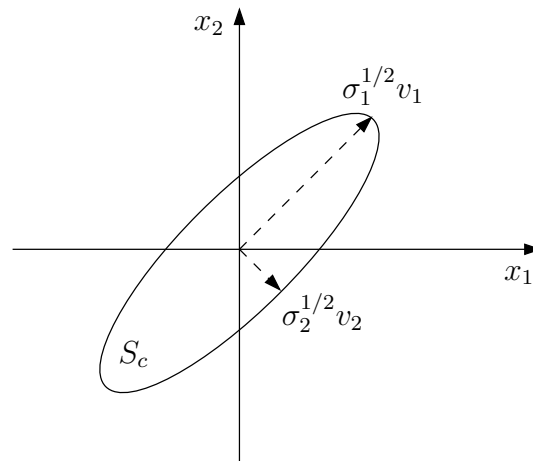
Figure 22.1: Interpretation of the controllability Gramian

Observability Gramian

A similar interpretation can be given for the observability Gramian

$$W_o = \int_0^\infty e^{A^T t} C^T C e^{At}\, dt$$

Define the set S_o as the set of all points in the state space which - when taken as initial conditions $x(0)$ - lead to output signals $y(t)$ that satisfy $\|y(t)\|_2 \leq 1$. Again the singular value decomposition

$$W_o = V \Sigma V^T$$

where

$$\Sigma = \text{diag}\left(\sigma_1,\ \sigma_2, \ldots \sigma_n\right) \qquad \text{and} \qquad V = \begin{bmatrix} v_1 & v_2 & \ldots & v_n \end{bmatrix}$$

determines the set S_o - it is a hyper-ellipsoid centered at the origin with semi-axes given by $v_i/\sqrt{\sigma_i}$. This set is illustrated for a second order system in Figure 22.2. Note that the axes are long in directions with small singular values, indicating directions that have little effect on the output. These directions are difficult to observe.
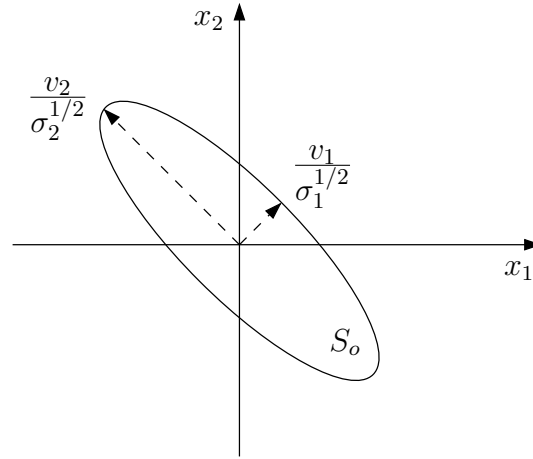

Figure 22.2: Interpretation of the observability Gramian

Balanced Realization

The question posed at the beginning of the chapter was: which state variables are important for the system and which ones are not? The singular value decomposition of the Gramians tells us which states show only a weak response to a control input (the ones associated with small singular values of W_c), and which ones have only weak influence on the observed output (the ones where the singular values of W_o are small). Now it would be unwise to remove a state variable from the model only because it shows little response to control inputs - the same state variable may have a strong effect on the output. The reverse may be true for states with small singular values of W_o. To find out which states have little influence both in terms of controllability *and* observability, we will use a special state space realization of the plant model which is known as *balanced realization.*

Initially we assumed a state space realization of $G(s)$ with system matrices A, B, C and D. Applying a similarity transformation T leads to a different state space model

$$(A, B, C, D) \qquad \rightarrow \qquad (T^{-1}AT, T^{-1}B, CT, D)$$

for the same plant. The eigenvalues and the input/output behaviour of both state space models are the same, because they are realizations of the same transfer function. The controllability and observability Gramians however are different: it is straightforward to check that if W_c and W_o are the Gramians of the first model, then $T^{-1}W_cT^{-T}$ and T^TW_oT are the Gramians of the second model.

A balanced realization of $G(s)$ has the property that its controllability and observability Gramians are equal and diagonal

$$W_c = W_o = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}$$

Using Cholesky factorization and singular value decomposition, it is always possible to find a similarity transformation T that brings a given state space model into this form - in MATLAB one can use the function `balreal()` for this task. The diagonal entries of the Gramian are called the *Hankel singular values* of the system. In the coordinate base of the state space associated with the balanced realization, a small Hankel singular value indicates that a state has little influence both in terms of controllability and observability. Therefore, this realization is well suited for model reduction by removing "unimportant" state variables. We will discuss two different ways of doing this.

Let (A, B, C, D) be a balanced realization of $G(s)$ with n state variables. Assume the inspection of the Hankel singular values indicates that only r states are significant and that the last $n - r$ Hankel singular values are small enough to be neglected. Partition the state space model as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \qquad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \qquad C = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \tag{22.1}$$

where $A_{11} \in \mathbb{R}^{r \times r}$, $A_{22} \in \mathbb{R}^{(n-r) \times (n-r)}$ etc.

Balanced Truncation

The subsystem (A_{11}, B_1, C_1, D) of the partitioned model (22.1) contains the states with significantly large Hankel singular values. One approach to model order reduction is to use this system with r state variables as an approximation of the full order model $G(s)$. This approach is known as *balanced truncation*, because the parts of the model associated with insignificant states are simply ignored.

An important property of the resulting reduced order model

$$G_{\text{tr}}(s) = \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & D \end{array} \right]$$

is that it satisfies

$$G_{\text{tr}}(j\infty) = G(j\infty) = D$$

The direct feedthrough terms of full order and truncated model are the same. This indicates that both models will exhibit similar high-frequency behaviour.

Balanced Residualization

An alternative approach is not to ignore the insignificant states, but to assume that they are constant and take them into account in the reduced order model. This method is known as *balanced residualization*. In the partitioned model

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u$$

$$y = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + Du$$

we make the assumption

$$\dot{x}_2 = 0$$

and eliminate x_2 from the model. It is straightforward to check that the resulting reduced order model is

$$G_{\mathrm{res}}(s) = \left[\begin{array}{c|c} A_{11} - A_{12} A_{22}^{-1} A_{21} & B_1 - A_{12} A_{22}^{-1} B_2 \\ \hline C_1 - C_2 A_{22}^{-1} A_{21} & D - C_2 A_{22}^{-1} B_2 \end{array} \right]$$

The feedthrough term is different from that of the full order model, indicating that the high-frequency behaviour will not be the same. In fact the reduced model was arrived at by assuming that derivatives of some state variables are zero. This is true in steady state, so we would expect similar behaviour of full order and reduced model at low frequencies. One can indeed verify that the steady state gains of both models are the same, i.e.

$$G(0) = G_{\mathrm{res}}(0) = D - CA^{-1}B$$

Example 22.1

A linearized model of a High Voltage DC transmission system with 31 state variables is approximated by a second order model. Figure 22.3 shows the magnitude frequency response when balanced truncation is used. Figure 22.4 shows the result of balanced residualization. The results illustrate that balanced truncation attempts to capture the high frequency behaviour of the full order model, whereas balanced residualization captures its low frequency behaviour. The MATLAB routines `sresid()` and `balmr()` were used for residualization and truncation, respectively.

Approximation Error

A useful property shared by both reduction techniques discussed in this chapter - here stated without proof - is the following. Let $G(s)$ denote the full order plant model and $\hat{G}(s)$ a reduced order approximation obtained either by balanced truncation or by balanced residualization. The H_∞ norm of the approximation error satisfies

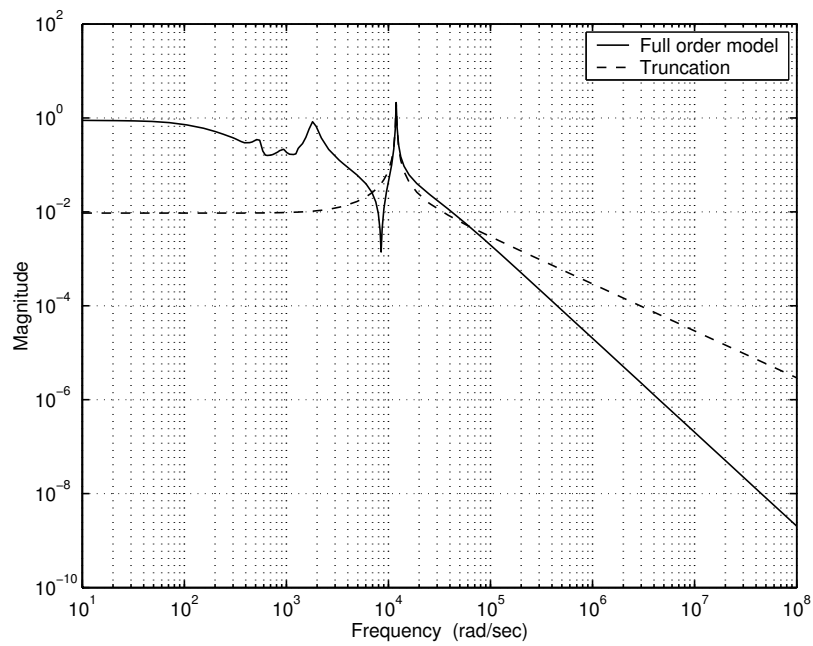$$\|G(s) - \hat{G}(s)\|_\infty \le 2 \sum_{i=r+1}^{n} \sigma_i$$

Figure 22.3: Balanced truncation: the reduced model captures - as far as possible with a second order model - the resonant peak and high-frequency roll-off
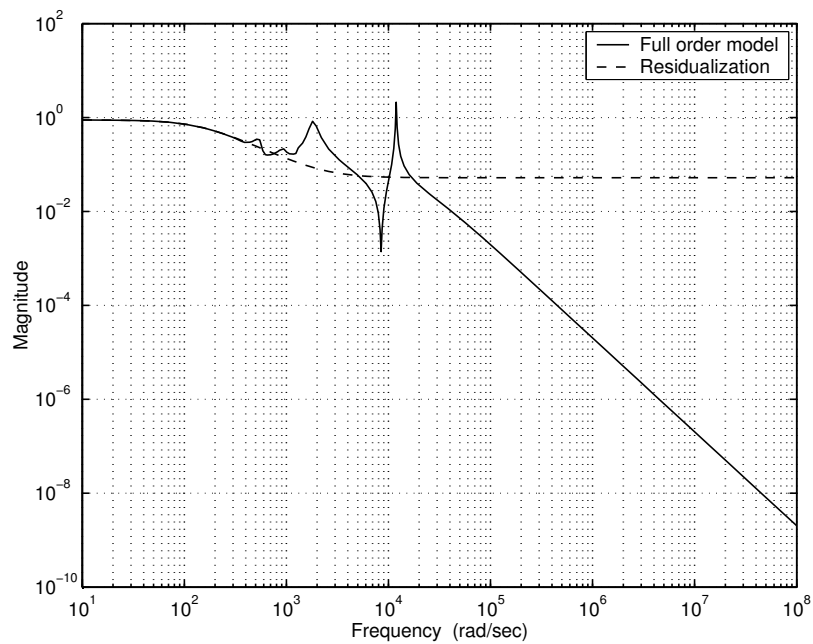

Figure 22.4: Balanced residualization: the reduced model captures the low frequency behaviour

In other words: the maximum error in the magnitude response is less than twice the sum of the Hankel singular values of the states that are being removed from the model.

Order Reduction for Unstable Models

So far we assumed that the full order model is stable - a balanced realization is defined only for stable systems. If the reduced order model is to be used for controller design, it would be unwise to ignore any unstable pole of the plant. One way of dealing with an unstable high order system is to split it up into a stable and an anti-stable part, i.e.

$$G(s) = G_s(s) + G_u(s)$$

where $G_s(s)$ is stable and $G_u(s)$ has all poles in the right half plane. The stable part of the system can then be reduced by balanced truncation or residualization to $\hat{G}_s(s)$, and a reduced model of the plant that retains all unstable modes is

$$\hat{G}(s) = \hat{G}_s(s) + G_u(s)$$

This approach is implemented in the MATLAB function `balmr()`.

Bibliography - Optimal and Robust Control

[5] V. Balakrishnan, S. Boyd, L. El Ghaoui, and E. Feron, "Linear matrix inequalities in systems and control," *Studies in Applied Mathematics*, vol. 15, 1994.

[6] P. Gahinet, A. Nemirovskii, A. J. Laub, and M. Chilali, "The lmi control toolbox," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, IEEE, vol. 3, 1994, pp. 2038–2041.

[7] J. M. Maciejowski, *Multivariable Feedback Design*. Amsterdam: Addison-Wesley, 1989, ISBN: 978-0-201-18243-9.

[8] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control - Analysis and Design*. New York: Wiley, 2005, ISBN: 978-0-470-01168-3.

[9] G. Strang, *Linear Algebra and Its Applications*. Stanford: Elsevier Science, 2014, ISBN: 978-1-483-26511-7.

[10] K. Zhou and J. C. Doyle, *Essentials of Robust Control -*. London: Prentice Hall, 1998, ISBN: 978-0-135-25833-0.

Part III

APPENDICES

Appendix A

Solutions to Exercises - Linear Quadratic Optimal Control

Solution to Problem 1.1   DC motor

Since for the desired steady-state angular velocity ω_0 a steady-state input signal u_0 is needed to translate the system's equilibrium. For the purpose define $\tilde{\omega} = \omega - \omega_0$ and $\tilde{u} = u - u_0$. Further because there is only one state and one input $Q = q \geq 0$, $R = \rho > 0$ and $S = s \geq 0$. Then the problem can be formulated as:
Find control input $\tilde{u}(t)$ that minimises the performance index

$$\int_{t_0}^{T} \left(q\tilde{\omega}^2(t) + \rho\tilde{u}^2(t) \right) dt + s\tilde{\omega}^2(T)$$

Then the control signal that has to be applied to the DC motor is $u(t) = u_0 + \tilde{u}(t)$. The coefficients q, ρ and s can be then used to tune the controller to the design requirements.

Solution to Problem 3.1   Hamilton-Jacobi equation

For the problem given $f(x, u, t) = u$, $l(x, u, \tau) = x^2 + u^2 + x^4/2$ and $m(x) = 0$. Denoting with $\bar{u}$ the minimising input and substituting in (3.3) and (3.4) leads to

$$\frac{\partial V^*}{\partial t}(x(t), t) = -(x^2 + \bar{u}^2 + \frac{1}{2}x^4) - \left[\frac{\partial V^*}{\partial x}\right]^T \bar{u}(t)$$

$$V^*(x(T), T) = 0$$

From

$$V^*(x(t), t) = \int_0^T (x^2 + \bar{u}^2 + \frac{1}{2}x^4) \, dt$$

follows

$$\frac{\partial V^*(x(t), t)}{\partial x} = \int_0^T (2x + 2x^3) \, dt.$$

and finally

$$\frac{\partial V^*}{\partial t}(x(t), t) = -(x^2 + \bar{u}^2 + \frac{1}{2}x^4) - \int_0^T (2x + 2x^3) \, dt \, \bar{u}(t)$$
$$V^*(x(T), T) = 0$$

Solution to Problem 4.1   Solution of Riccati equation

Taking the standard system description and cost function

$$V = \int_{t_0}^T \left(x^T Q x + u^T R u \right) dt + x^T S x$$

leads to $A = 0$, $B = 1$, $Q = 1$, $R = 1$, $S = 0$ which reduces the Riccati differential equation

$$-\dot{P} = Q + PA + A^T P - PBR^{-1}B^T P, \qquad\qquad P(T) = S$$
$$\text{to} \qquad -\dot{P} = 1 - P^2, \qquad\qquad\qquad\qquad P(T) = 0$$

The differential equation can be solved using variable separation method

$$\dot{P} = \frac{dP}{dt} = P^2 - 1 \qquad \Rightarrow \qquad \frac{dP}{P^2 - 1} = dt$$

Which after integration leads to

$$\int \frac{dP}{P^2 - 1} = \int dt \quad \Rightarrow \quad \frac{1}{2} \ln\left(\frac{P-1}{P+1}\right) = t + \tilde{C} \quad \Rightarrow \quad \frac{P-1}{P+1} = e^{2t + 2\tilde{C}} = Ce^{2t}$$

Then, solving for P leads to

$$P(t) = \frac{1 + Ce^{2t}}{1 - Ce^{2t}}$$

and then using that $P(T) = 0 \quad \Rightarrow \quad Ce^{2t} = -1 \quad \Rightarrow \quad C = -e^{-2T}$ leads to

$$P(t) = \frac{1 - e^{2(t-T)}}{1 + e^{2(t-T)}}$$

Thus the optimal control is

$$u^*(t) = -R^{-1}B^T P(t) x(t)$$
$$u^*(t) = -\frac{1 - e^{2(t-T)}}{1 + e^{2(t-T)}} x(t)$$

Solution to Problem 4.2   Schur complement

Recall that X is positive definite ($X > 0$), if $X = X^T$ and $y^T X y > 0$ holds for all $y \neq 0$, or equivalently if the eigenvalues of X are positive. Let $T \in \mathbb{R}^{n \times n}$ be any full-column matrix. Then

$$T^T X T > 0 \quad \Rightarrow \quad q^T T^T X T q > 0, \quad q \neq 0$$

Defining $y = Tq$ leads to $y^T X y > 0$ and thus to $X > 0$.

Then $X > 0$ holds if and only if

$$\begin{bmatrix} I & -LN^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} M & L \\ L^T & N \end{bmatrix} \begin{bmatrix} I & 0 \\ -N^{-1}L^T & I \end{bmatrix} > 0$$

since clearly the selected T matrix is full-column rank (its lower triangular with 1's on the diagonal). Performing the multiplication leads to

$$\begin{bmatrix} M - LN^{-1}L^T & 0 \\ 0 & N \end{bmatrix} > 0$$

which is positive definite only when $N > 0$ and $M - LN^{-1}L^T > 0$.

To prove $X > 0 \quad \Leftrightarrow \quad M > 0, \quad N - L^T M^{-1} L > 0$ select

$$T = \begin{bmatrix} I & -M^{-1}L \\ 0 & I \end{bmatrix}$$

Solution to Problem 4.3   Completing squares

a)   Rewrite

$$\begin{bmatrix} M & L \\ L^T & N \end{bmatrix} = T^{-T} T^T \begin{bmatrix} M & L \\ L^T & N \end{bmatrix} T T^{-1}$$

where T is selected from the Schur complement

$$T = \begin{bmatrix} I & 0 \\ -N^{-1}L^T & I \end{bmatrix}$$

Then

$$W(x, u) = \begin{bmatrix} x^T & u^T \end{bmatrix} T^{-T} \begin{bmatrix} M - LN^{-1}L^T & 0 \\ 0 & N \end{bmatrix} T^{-1} \begin{bmatrix} x \\ u \end{bmatrix}$$

Using that

$$\begin{bmatrix} I & 0 \\ X & I \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ -X & I \end{bmatrix}$$

follows

$$W(x, u) = \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} I & LN^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} M - LN^{-1}L^T & 0 \\ 0 & N \end{bmatrix} \begin{bmatrix} I & 0 \\ N^{-1}L^T & I \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}$$

$$= \begin{bmatrix} x^T & x^T LN^{-1} + u^T \end{bmatrix} \begin{bmatrix} M - LN^{-1}L^T & 0 \\ 0 & N \end{bmatrix} \begin{bmatrix} x \\ N^{-1}L^T x + u \end{bmatrix}$$

$$= x^T \left(M - LN^{-1}L^T \right) x + \left(x^T LN^{-1} + u^T \right) N \left(N^{-1}L^T x + u \right)$$

$$\min_u W(x, u) = \min_u \left(x^T \left(M - LN^{-1}L^T \right) x + \left(N^{-1}L^T x + u \right)^T N \left(N^{-1}L^T x + u \right) \right)$$

The first term is independent of u and the second is quadratic in u and non-negative (since $N > 0$). The control signal that gets the second term to zero (and therefore minimises $W(x, u)$) is

$$\bar{u} = -N^{-1}L^T x$$

b)   Rewrite the term to be minimised from Hamilton-Jacobi equation (4.3) as

$$W(x, u) = x^T Q x + u^T R u + x^T (PA + A^T P)x + x^T PBu + u^T B^T P x$$

$$= \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} Q + PA + A^T P & PB \\ B^T P & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}$$

Then using the above results follows $\bar{u} = -R^{-1}B^T P x$,   i.e. (4.4).

Solution to Problem 4.4   Riccati Differential Equation

Define an arbitrary control input $u(t)$

$$u(t) = -R^{-1}B^T P(t)x(t) + v(t),$$

where $P(t)$ and $v(t)$ are arbitrarily chosen.

Substitution of $u(t)$ in the state equation yields

$$\dot{x} = (A - BR^{-1}B^T P)x + Bv. \tag{A.1}$$

In the cost function

$$V = \int_{t_0}^{T} (x^T Q x + u^T R u) \, dt + x^T(T)Sx(T),$$

we can rewrite the term $u^T R u$ as

$$u^T R u = (-R^{-1}B^T P x + v)^T R(-R^{-1}B^T P x + v)$$

$$= x^T PBR^{-1}B^T P x + v^T R v - v^T B^T P x - x^T PBv. \tag{A.2}$$

From equation (A.1) we have

$$Bv = \dot{x} - Ax + BR^{-1}B^T P x.$$

The left multiplication of Bv with $x^T P$ yields

$$x^T P B v = x^T P \dot{x} - x^T P A x + x^T P B R^{-1} B^T P x. \tag{A.3}$$

Replacing the term $u^T R u$ in V with equation (A.2) and (A.3) yields

$$
\begin{aligned}
V = \int_{t_0}^{T} \Big(& x^T Q x + x^T P B R^{-1} B^T P x + v^T R v - x^T P \dot{x} + x^T P A x \\
& - x^T P B R^{-1} B^T P x - \dot{x}^T P x + x^T A^T P x - x^T P B R^{-1} B^T P x \Big) dt \\
& + x_T^T (S - P_T) x_T + x_T^T P_T x_T - x_0^T P_0 x_0 + x_0^T P_0 x_0,
\end{aligned}
$$

where

$$
\begin{aligned}
x_T &= x(T) \\
P_T &= P(T) \\
x_0 &= x(t_0) \\
P_0 &= P(t_0),
\end{aligned}
$$

and

$$
\begin{aligned}
x_T^T P_T x_T - x_0^T P_0 x_0 &= \int_{t_0}^{T} \frac{d}{dt}(x^T P x) dt \\
&= \int_{t_0}^{T} (\dot{x}^T P x + x^T \dot{P} x + x^T P \dot{x}) dt.
\end{aligned}
$$

After the cancellation and rearrangement of V,

$$
\begin{aligned}
V = \int_{t_0}^{T} & x^T (Q + P A + A^T P - P B R^{-1} B^T P + \dot{P}) x \, dt \\
& + \int_{t_0}^{T} v^T R v \, dt + x_T^T (S - P_T) x_T + x_0^T P_0 x_0.
\end{aligned}
$$

When $P(t)$ satisfies

$$
\begin{aligned}
Q + P A + A^T P - P B R^{-1} B^T P + \dot{P} &= 0 \\
S - P_T &= 0.
\end{aligned}
$$

the following terms in V remain:

$$V = \int_{t_0}^{T} v^T R v \, dt + x_0^T P_0 x_0$$

Since $R > 0$, choosing $v(t) \equiv 0$ leads to the optimal control input $u(t) = u^*(t)$. We obtain the minimal performance index

$$V^* = x_0^T P_0 x_0.$$

Solution to Problem 5.1   Hamiltonian state-space model

a)   The transfer function y to v is the transfer function of the lower dashed block. The state-space matrices of the block are $\bar{A} = -A^T$, $\bar{B} = -C^T$, $\bar{C} = -B^T$ and $\bar{D} = 0$ and hence

$$G_{vy}(s) = \bar{C}(sI - \bar{A})^{-1}\bar{B} + \bar{D} = B^T(sI + A^T)^{-1}C^T$$

The eigenvalues of $G_{vy}(s)$ are the negative eigenvalues of $G(s)$.

b)   The state-space model is

$$\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} A & -BR^{-1}B^T \\ -C^TC & -A^T \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} r$$

$$y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix}$$

Note that by $Q = C^TC$ the system's A matrix is the Hamiltonian matrix.

Solution to Problem 5.2   **DC motor** Construct the Hamiltonian matrix with $A = -a$, $B = b$, $R = \rho$ and $Q = 1$:

$$H = \begin{bmatrix} -a & -b\frac{1}{\rho}b \\ -1 & a \end{bmatrix} = \begin{bmatrix} -0.5 & -\frac{150^2}{\rho} \\ -1 & 0.5 \end{bmatrix}$$

As transformation matrix U can be used the right eigenvectors of H. In MATLAB: `[U,L] = eig(H)`. MATLAB arranges the real eigenvalues in increasing order, therefore the first one is the stable and the second the unstable one.

For a) $\rho = 100$, $s = 0$

$$H = \begin{bmatrix} -0.5 & -225 \\ -1 & 0.5 \end{bmatrix} \quad \Rightarrow \quad U = \begin{bmatrix} 1 & 1 \\ 0.0645 & -0.0689 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} -15.0083 & 0 \\ 0 & 15.0083 \end{bmatrix}$$

$$G = -(u_{22} - su_{12})^{-1}(u_{21} - su_{11}) = -\frac{u_{21}}{u_{22}} = 0.9355$$

Substituting in (5.7) leads to

$$P(t) = \frac{0.0645 - 0.0689e^{-15.0083(T-t)}0.9355e^{-15.0083(T-t)}}{1 + e^{-15.0083(T-t)}0.9355e^{-15.0083(T-t)}} \approx 0.06\frac{1 - e^{-30(1-t)}}{1 + 0.94e^{-30(1-t)}}$$

Then the feedback gain is $f(t) = -\frac{1}{\rho}bP(t)$.

a) $\rho = 100$, $s = 0$ $P(t) \approx 0.06 \dfrac{1 - e^{-30(1-t)}}{1 + 0.94e^{-30(1-t)}}$ $f(t) \approx -0.1 \dfrac{1 - e^{-30(1-t)}}{1 + 0.94e^{-30(1-t)}}$

b) $\rho = 10^3$, $s = 0$ $P(t) \approx 0.2 \dfrac{1 - e^{-10(1-t)}}{1 + 0.8e^{-10(1-t)}}$ $f(t) \approx -0.03 \dfrac{1 - e^{-10(1-t)}}{1 + 0.8e^{-10(1-t)}}$

c) $\rho = 10^4$, $s = 0$ $P(t) \approx 0.5 \dfrac{1 - e^{-3(1-t)}}{1 + 0.5e^{-3(1-t)}}$ $f(t) \approx -0.0075 \dfrac{1 - e^{-3(1-t)}}{1 + 0.5e^{-3(1-t)}}$

d) $\rho = 10^3$, $s = 0.19$ $P(t) \approx 0.2 \dfrac{10^4 + 5e^{-10(1-t)}}{10^4 - 6e^{-10(1-t)}}$ $f(t) \approx -0.03 \dfrac{10^4 + 5e^{-10(1-t)}}{10^4 - 6e^{-10(1-t)}}$

The plots of $f(t)$, $u(t) = f(t)\omega(t)$ and $\omega(t)$ in the interval under the initial condition $\omega(0) = 1$ are shown in Fig. A.1 and Fig. A.2-left.
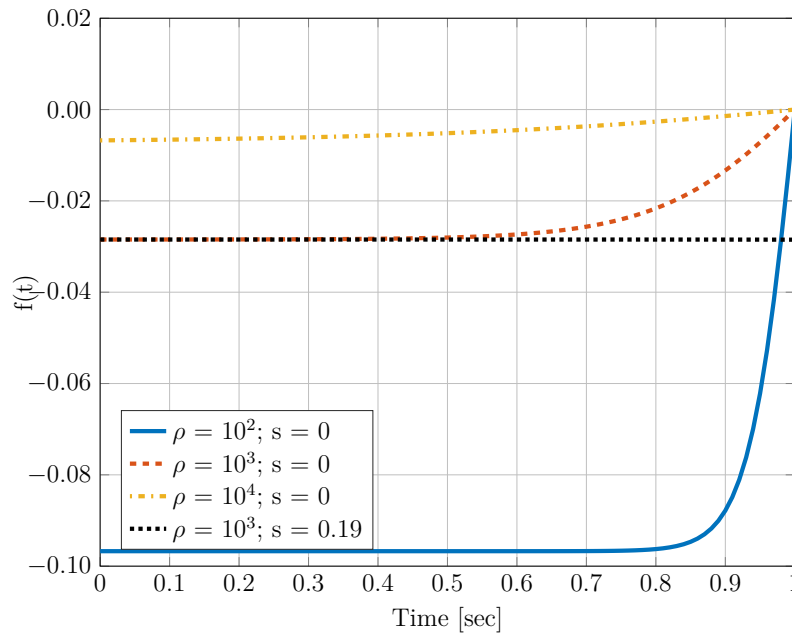
Figure A.1: Feedback gain $f(t)$

From the expressions for $f(t)$ (and the figures) a), b) and c) is obvious that by increasing ρ the argument of the exponents in $P(t)$ gets smaller, corresponding to a larger time-constant and therefore to slower control. The feedback gains in b) and d) are similar for small t and have difference only at the end of the time period where the later converges to a finite value ($P(1) = s = 0.19$). To see a difference in the control signal and the angular velocity in Fig. A.2-right the signals are showed "zoomed-in" near $t = 1$ sec.

MATLAB code: `orc05_DCmotor.m`, `orc05_DCmotor_sim.mdl`.

Solution to Problem 6.1   **DC motor - LQR problem**
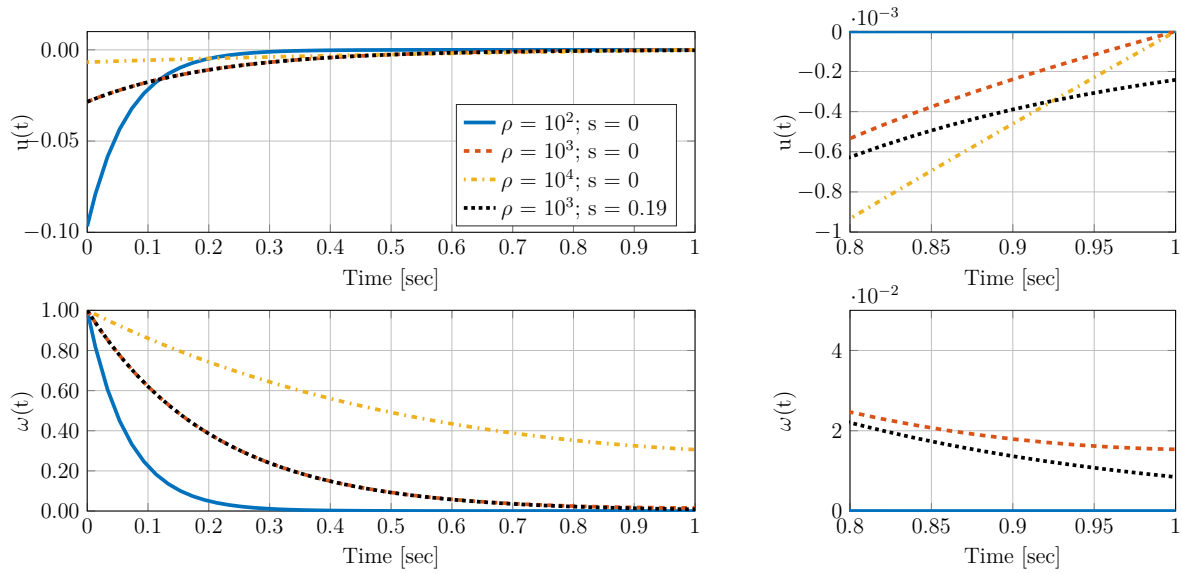
Figure A.2: Control signal $u(t)$ and angular velocity $\omega(t)$

Note that now the infinite horizon control problem is considered, for which $P(t)$ and $F(t)$ are constant matrices. Then the solution can be easily obtained using the LQR function in MATLAB:

```
F=-lqr(A,B,Q,R)
```

See the MATLAB files: `orc05_DCmotorLQR.m`, `orc05_DCmotor_sim.mdl`

Solution to Problem 6.2   LQR control

From the Figure 6.2 the control signal can be derived

$$u = -f_1(r - x_1) + (f_2 x_2 + f_3 x_3)$$
$$u = -f_1 r + (f_1 x_1 + f_2 x_2 + f_3 x_3) = f\,x - f_1 r$$

a)   In determining the optimal control, we assume that the input is zero. Let us determine the state-feedback gain matrix f, where

$$f = \begin{bmatrix} f_1 & f_2 & f_3 \end{bmatrix}$$

such that the following performance index is minimized:

$$V = \int_0^\infty \left(x^T Q x + u^T R u \right) dt$$

To get the fast response, q_{11} must be sufficiently large compared with q_{22}, q_{33} and R, because $y(t) = x_1(t)$. The optimal control gain is obtained using the `lqr` MATLAB function. The controller can be further tuned with respect to some non-zero initial conditions x_0. For the purpose of simulation the function `initial` can be used.

b)   The state-space equation for the closed-loop system is

$$\dot{x} = Ax + bu = Ax + b(fx - f_1 r) = (A + bf)x - bf_1 r$$
$$y = cx = x_1$$

To obtain the unit step-response, following code can be used.

```
Gcl = ss(A+b*f, -b*f(1), c, d);
[y,x,t] = step(Gcl, 10);
```

MATLAB code: `orc05_tuneLQR.m`

Solution to Problem 6.3   Infinite horizon LQR

Construct the Hamiltonian matrix and find its eigenvalues λ and eigenvectors U. Select the eigenvectors corresponding to the stable (with negative real part) eigenvalues and stack them together in a matrix $\begin{bmatrix} U_{11} \\ U_{21} \end{bmatrix}$. Now $P = U_{21} U_{11}^{-1}$ and

$$F = \begin{bmatrix} -3.6103 & -1.5514 \end{bmatrix}$$

This lead to the closed-loop eigenvalues $-2.3271 \pm j0.6446$.

MATLAB files: `orc06_inf_horizon_LQR.m`

Solution to Problem 7.1   Mass-spring system

a) Using Newton's laws, the equation of motion is

$$m\ddot{q}(t) + kq(t) = u(t)$$

Letting $x_1(t) = q(t)$ denote the position of mass and $x_2(t) = \dot{q}(t)$ its velocity, the dynamics can be expressed in the state space as

$$\dot{x}_1(t) = x_2(t); \qquad \dot{x}_2(t) = -\frac{k}{m}x_1(t) + \frac{1}{m}u(t)$$

or

$$\dot{x}(t) = Ax(t) + bu(t)$$

with

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \qquad A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & 0 \end{bmatrix}; \qquad b = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}$$

For simplicity assume that $m = 1$ and $k = 1$, where appropriate units are used.

In the absence of control, initial conditions will produce a persistent sinusoidal motion of mass. As such, we seek a control law that regulates the position of the mass to its equilibrium value of $q(t) = 0$. Thus, we define the state of interest, z as $x_1(t)$.

$$z(t) = cx(t) \qquad \text{with} \qquad c = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

b*) Since the control is scalar, the cost function takes the following form

$$V = \int_0^\infty \left(z^2(t) + \rho u^2(t) \right) dt \qquad \text{with} \qquad \rho > 0$$

where ρ is the control weighting parameter. The optimal LQR control law takes the form $u(t) = fx(t)$ where the feedback gain f is a row vector $f = \begin{bmatrix} f_1 & f_2 \end{bmatrix}$. To determine the value of gain, we must solve the algebraic Riccati equation. Recalling that we want a symmetric solution to the Riccati equation, letting

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix} \tag{A.4}$$

and using the values of A, b and c given above, the Riccati equation becomes

$$0 = \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}$$
$$- \frac{1}{\rho} \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix}$$

Carrying out the matrix multiplication leads to the following three equations in the three unknown P_{ij} from Equation (A.4)

$$0 = 2P_{12} - 1 + \frac{1}{\rho}P_{12}^2$$

$$0 = -P_{11} + P_{22} + \frac{1}{\rho}P_{12}P_{22} \tag{A.5}$$

$$0 = -2P_{12} + \frac{1}{\rho}P_{22}^2$$

Solving the first equation for P_{12} and simplifying yields $P_{12} = -\rho \pm \rho\sqrt{1 + \dfrac{1}{\rho}}$.

Both the positive and negative choices for P_{12} are valid solution for Riccati equation. While we are only interested in the positive semidefinite solution, we still need more information to resolve which choice of P_{12} leads to the unique choice of $P \geq 0$. Rewriting the 3rd equation

$$2P_{12} = \frac{1}{\rho}P_{22}^2$$

indicates that P_{12} must be positive, since the right hand side of the equation must always be positive. The last equation indicates that there will also be a $\pm$ sign ambiguity in selecting the appropriate P_{22}. To resolve the ambiguity we use Sylvester's Test, which says that for $P \geq 0$ both

$$P_{11} \geq 0 \qquad \text{and} \qquad P_{11}P_{22} - P_{12}^2 \geq 0$$

Solving Equation (A.5) using the above mentioned relations, which clearly show that $P_{22} > 0$, gives the remaining elements of P

$$P_{11} = \rho\sqrt{2(1 + \frac{1}{\rho})(\sqrt{1 + \frac{1}{\rho}} - 1)}; \quad P_{12} = -\rho + \rho\sqrt{1 + \frac{1}{\rho}}$$

$$P_{22} = \rho\sqrt{2(\sqrt{1 + \frac{1}{\rho}} - 1)}$$

The final step in computing the controller is to evaluate the control gains $f = -b^T/\rho P$

$$f_1 = -\sqrt{1 + \frac{1}{\rho}} + 1; \qquad f_2 = -\sqrt{2(\sqrt{1 + \frac{1}{\rho}} - 1)}$$

Evaluating the eigenvalues of the closed-loop dynamics leads to a pair of complex conjugate closed-loop poles

$$\lambda_{1,2} = -\frac{1}{2}\sqrt{2(\sqrt{1 + \frac{1}{\rho}} - 1)} \pm j\frac{1}{2}\sqrt{2(\sqrt{1 + \frac{1}{\rho}} + 1)}$$

c) Given the control gains as a function of the control weighting ρ it is useful to examine the locus of the closed-loop poles for the system as ρ varies over $0 < \rho < \infty$. For large ρ

$$\rho \to \infty, \quad f_1 \to 0, \quad f_2 \to 0, \quad \lambda_{1,2} = \pm j1$$

Thus when the control is expensive there is practically no feedback, and the closed-loop eigenvalues are the same as the open-loop eigenvalues. For small ρ

$$\rho \to 0, \quad f_1 \to \infty, \quad f_2 \to \infty, \quad \lambda_{1,2} \to -\infty \pm j\infty$$

This is consistent with how ρ influences the cost. Larger values of ρ place a heavy penalty on the control and lead to low gains with slow transients (poles close to the imaginary axis), while small values of ρ allow larger control action, move the poles further to the left from the imaginary axis, thus leading to fast transients.

d)   MATLAB files: `orc05_mass_spring.m`

Solution to Problem 7.2   Spectral factorisation

The transfer function u to y is

$$G_{yu} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & -1 \\ -5 & s \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 3 \end{bmatrix} = \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & 1 \\ 5 & s \end{bmatrix} \begin{bmatrix} 0 \\ 3 \end{bmatrix}}{s^2 - 5} = \frac{3}{s^2 - 5} = \frac{b(s)}{a(s)}$$

Then

$$p(s) = a(s)a(-s) + \frac{1}{\rho}b(s)b(-s) = ((-s)^2 - 5)(s^2 - 5) + \frac{1}{1}9 = s^4 - 10s^2 + 34$$

The four roots are $\pm 2.3271 \pm j0.6446$, thus the closed-loop eigenvalues are $-2.3271 \pm j0.6446$ and the closed-loop characteristic polynomial is

$$a_c(s) = (s + 2.3271 + j0.6446)(s + 2.3271 - j0.6446) = s^2 + 4.6542s + 5.831$$

Recall that

$$a_c(s) = \det(sI - A - bf) = \det\left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 5 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 3 \end{bmatrix} \begin{bmatrix} f_1 & f_2 \end{bmatrix} \right) = \det \begin{bmatrix} s & -1 \\ -5 - 3f_1 & s - 3f_2 \end{bmatrix}$$

or

$$a_c(s) = s^2 + 4.6542s + 5.831 = s^2 - 3f_2 s - (5 + 3f_1)$$

By comparing the coefficients follows $f_2 = -1.5514$, $f_1 = -3.6103$, which is exactly the same result obtained in Problem 6.3.

Solution to Problem 7.3   Optimal state feedback

Check the closed-loop characteristic polynomial

$$a_c(s) = \det \begin{bmatrix} s & -1 \\ 1 & s + q \end{bmatrix} = s^2 + qs + 1$$

Thus closed-loop stability requires $q > 0$ (recall that for first and second order system sufficient condition for stability is that all coefficients are positive).

The optimal state feedback loop gain $L(s)$ is given by

$$L(s) = -f(sI - A)^{-1}b = \frac{1 + qs}{s^2}$$

and Kalman's identity, a necessary and sufficient condition for LQR optimality for a stable control law, requires the satisfaction of inequality (7.4) or what is equivalent

$$|1 + L(j\omega)| \geq 1 \qquad \forall \omega.$$

Substituting $L(j\omega)$ and squaring both sides (doesn't change the inequality) leads to

$$\left|\frac{1 - \omega^2 + jq\omega}{-\omega^2}\right|^2 \geq 1, \quad \Rightarrow \quad \frac{(1 - \omega^2)^2 + q^2\omega^2}{\omega^4} \geq 1, \quad \Rightarrow \quad 1 + (q^2 - 2)\omega^2 \geq 0$$

The last inequality can be satisfied for all ω if and only if $q \geq \sqrt{2}$. Thus, for example, the stable gain matrix $f = \begin{bmatrix} -1 & -2 \end{bmatrix}$ is optimal with respect to some LQR problem, but the stable gain matrix $f = \begin{bmatrix} -1 & -1 \end{bmatrix}$ is not optimal with respect to any LQR problem.

Solution to Problem 7.4   Pendulum

a)   With the state vector $\begin{bmatrix} \psi & \dot\psi & \theta & \dot\theta \end{bmatrix}$ the state-space matrices are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 2.5 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2.5 & 0 & -2 & 0 \end{bmatrix}; \qquad b = \begin{bmatrix} 0 \\ 1 \\ 0 \\ -1 \end{bmatrix}; \qquad c = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix}.$$

b)   The open-loop characteristic equation

$$a(s) = \det(sI - A) = s^4 - 0.5s^2 - 2.5$$

has eigenvalues ± 1.361 and $\pm j1.162$. To obtain a time-constant of $1/3$ the dominant closed-loop pole(s) should have a real part of -3. We could choose four poles consistent with this constraint, but this still leaves a lot of choice. By means of spectral factorisation theorem, we can make a reasonable choice of the eigenvalues. The open-loop transfer function is

$$\frac{b(s)}{a(s)} = \frac{1}{s^4 - 0.5s^2 - 2.5}$$

By spectral factorisation, the $2n$ degree polynomial is

$$p(s) = a(s)a(-s) + \frac{1}{\rho}b(s)b(-s) = (s^4 - 0.5s^2 - 2.5)^2 + \frac{1}{\rho}$$

Note that the characteristic polynomial of the negative closed-loop of

$$\tilde{L}(s) = \frac{1}{\rho}G(s)G(-s) = \frac{1}{\rho}\frac{b(s)b(-s)}{a(s)a(-s)}$$

is equal to $p(s)$. Thus, in order to determine ρ leading to the desired time constant, construct the system $G(s)G(-s)$ and use standard root-locus software (e.g., `rltool` in MATLAB) to find a value of $k = 1/\rho$ leading to the desired closed-loop pole locations in the left-half plane - see Fig. A.3 (this is known as "symmetric root-locus" because by the properties of the Hamiltonian each pole in the left half-plane has a mirror in the right half-plane).

Selecting $1/\rho = k = 1.5\,10^7$ leads to the optimal closed-loop poles $-3.0238 \pm j7.2742$ and $-7.3034 \pm j3.0117$.
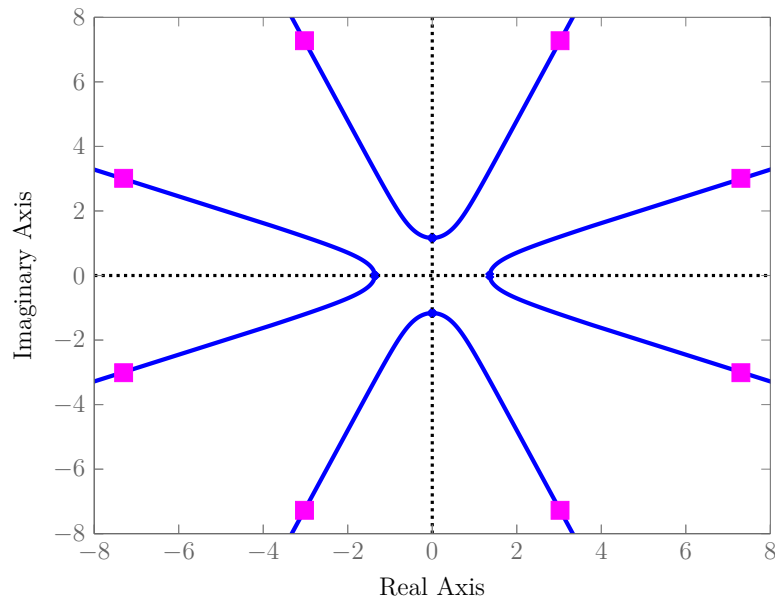
Figure A.3: Symmetric root-locus for LQR design for pendulum

c)   Knowing the optimal closed-loop locations, the problem now is to find the feedback
     gain matrix f. One method is to find $a_c(s) = \det(sI - A - bf)$ as function of
     f and solve for f (as in Problem 7.2). However, since this is a tedious task one
     can instead use a pole-placement - the desired closed-loop poles are known and are
     optimal! Thus   `f = -place(A, b, stabp)`   where `stabps` are the stable poles
     of $p(s)$ leads to

$$f = \begin{bmatrix} -3875.5 & -1283.9 & -3662.2 & -1263.2 \end{bmatrix}.$$

MATLAB files: `orc07_pendelum.m`

Solution to Problem 9.1   Autonomous system

We start with the solution to the state equation

$$\dot{x} = Ax; \qquad x(t) = e^{At}x_0; \qquad x(0) = x_0$$

Substituting for $x(t)$ in the cost function, we get

$$V = \int_0^\infty (x_0^T e^{A^T t} Q e^{At} x_0)\, dt = x_0^T \int_0^\infty (e^{A^T t} Q e^{At}\, dt) x_0 = x_0^T P x_0$$

where

$$P = \int_0^\infty (e^{A^T t} Q e^{At}\, dt)$$

To show P is the solution to $PA + A^T P + Q = 0$, substitute for P and use that A is stable $\left(\lim_{t \to \infty} e^{At} \to 0\right)$:

$$\int_0^\infty (e^{A^T t} Q e^{At} A + A^T e^{A^T t} Q e^{At}) \, dt + Q = 0$$

$$\int_0^\infty \frac{d}{dt}(e^{A^T t} Q e^{At}) dt + Q = 0$$

$$e^{A^T t} Q e^{At}\Big|_0^\infty + Q = 0$$

$$0 - Q + Q = 0$$

which proves the result.

Solution to Problem 9.2   LQG control

a)   The LQG controller can be seen in Fig. 9.1. Using that $A = 1$, $b = 1$ and $c = 1$ simplifies to the controller in Fig. A.4 with transfer function

$$K(s) = F(sI - A - BF - F_e C)^{-1} F_e = \frac{f f_e}{s - 1 - f - f_e}$$
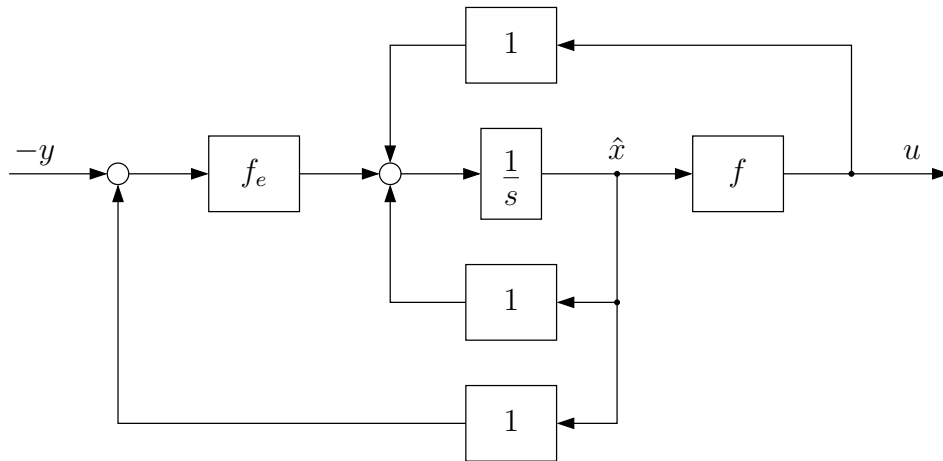
Figure A.4: LQG controller

b)   Writing the filter algebraic Riccati equation (FARE):

$$P_e A^T + A P_e - P_e C^T R_e^{-1} C P_e + Q_e = 0.$$

$$2p_e - \frac{1}{r_e} p_e^2 + 1 = 0 \quad \to \quad p_{e\,1,2} = r_e \pm \sqrt{r_e^2 + r_e}$$

The stable solution is the positive solution: $p_e = r_e + \sqrt{r_e^2 + r_e} > 0$, therefore

$$f_e = -P_e C^T R_e^{-1} = -p_e \frac{1}{r_e} = -1 - \sqrt{1 + \frac{1}{r_e}}$$

c) By duality

$$f = -1 - \sqrt{1 + \frac{1}{\rho}} = -3 \qquad \rightarrow \qquad \rho = \frac{1}{3}$$

d) From $r_e = 0.125$ we have $f_e = -4$, and with $f = -3$

$$\dot{x} = (A + bf)x + w - bf\varepsilon = (1 - 3)x + w + 3\varepsilon = -2x + w + 3\varepsilon;$$
$$\dot{\varepsilon} = (A + f_e c)\varepsilon + w + f_e v = (1 - 4)\varepsilon + w - 4v = -3\varepsilon + w - 4v$$

The error dynamics is not much faster than the plant. To make it five times faster than the plant we need

$$\dot{\varepsilon} = -10\varepsilon + w + f_e v = (1 + f_e) + w + f_e v \qquad \rightarrow \qquad f_e = -11$$

$$f_e = -1 - \sqrt{1 + \frac{1}{\tilde{r}_e}} = -11 \qquad \rightarrow \qquad \tilde{r}_e = 0.01$$

e) From d), $\alpha_e = -10$ and

$$\xi(t) = w(t) + f_e v(t) = w(t) - 11v(t).$$

We also have

$$E[\xi(t)\xi(t + \tau)] = E[w(t)w(t + \tau)] + f_e^2 E[v(t)v(t + \tau)]$$
$$= (q_e + r_e f_e^2)\delta(\tau) = 2.21\delta(\tau)$$

Solution to Problem 9.3   LQG and disturbance

Disturbance d enters the loop at a point inside the controller, which is not a realistic assumption. The error dynamics is not controllable from d, so the transfer function from d to y is same as that under state feedback. From Figure A.5 we get the transfer function
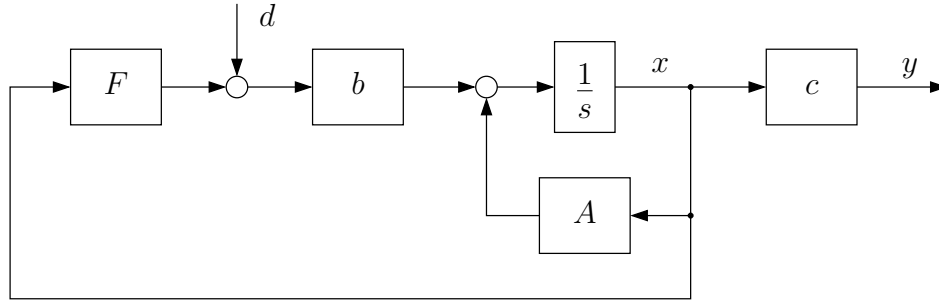
Figure A.5: LQG controller

from d to y as

$$\frac{Y(s)}{D(s)} = c(sI - A - bf)^{-1}b = \frac{cb}{s - A - bf} = \frac{1}{s + 2}.$$

Next if the disturbance enters the loop at a point outside the controller, at the plant input, then the estimator dynamics do affect the controlled output. The controller transfer function (from $-y$ to u) is

$$K(s) = f(sI - A - bf - f_ec)^{-1}f_e$$

and the plant transfer function (from u to y) is

$$G(s) = c(sI - A)^{-1}b = \frac{cb}{s - a}$$

Substituting $f = -3$ and $f_e = -4$ in the transfer function from d to y gives

$$\frac{Y(s)}{D(s)} = \frac{G}{1 + GK} = \frac{(bc)(s - a - bf - f_ec)}{(s - a)(s - a - bf - f_ec) + cbff_e} = \frac{s + 6}{s^2 + 5s + 6} = \frac{s + 6}{(s + 2)(s + 3)}$$

Appendix B

Solutions to Exercises - Optimal and Robust Control

Solution to Problem 11.1   LQG control of an aircraft

One way to construct the generalised plant $P(s)$ is (manually) constructing its matrices. Since the only states of the generalised plant in Fig. 11.3 are the states of the aircraft $A_P = A$. The rest can be constructed by direct comparison with Fig. 11.3 - note that the input to $P(s)$ is $\begin{bmatrix} r^T & w_1^T & w_2^T & u^T \end{bmatrix}^T$.

$$
P(s) = \left[\begin{array}{c|cccc} A & 0 & Q_e^{1/2} & 0 & B \\ \hline Q^{1/2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R^{1/2} \\ -C & I & 0 & -R_e^{1/2} & 0 \end{array} \right] = \left[\begin{array}{c|cccc} A & 0 & B & 0 & B \\ \hline C & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R^{1/2} \\ -C & I & 0 & -R_e^{1/2} & 0 \end{array} \right]
$$

where the second expression holds for the initial values $Q = C^T C$ and $Q_e = BB^T$ (which is often a good initial guess:the disturbance enters the same way as the input signal u and we are interested in the output signals y). Set initially $R = I$, $R_e = I$. The generalised plant can now be constructed by: `Gplant = mk_LQG_gplant(Gairc,R,Re)`.

Next the LQG controller can be designed: `K = h2syn ( Gplant, 3, 3 )` (Alternatively one can use the `lqg` function in MATLAB).

Finally simulate the response closed-loop system to the desired reference step command and update R and R_e (and Q and Q_e if necessary) to meet the design requirements.

MATLAB files: `orc11_LQG.m`, `aircraft_model.m`, `aircraft_sim.mdl`, `mk_LQG_gplant.m`

Solution to Problem 12.1   Matrix norms

From the definition of the induced norm we get

$$\|AB\| = \max_{x \neq 0} \frac{\|ABx\|}{\|x\|} = \max_{x \neq 0} \frac{\|ABx\|\|Bx\|}{\|Bx\|\|x\|} \leq \max_{y \neq 0} \frac{\|Ay\|}{\|y\|} \max_{x \neq 0} \frac{\|Bx\|}{\|x\|} = \|A\|\|B\|$$

In the above proof the inequality

$$\max_{x \neq 0} \frac{\|ABx\|}{\|Bx\|} \leq \max_{y \neq 0} \frac{\|Ay\|}{\|y\|}$$

has been applied. We know that Bx is always a vector in the column space of the matrix B and the column rank of this matrix might not be full. In this case we cannot produce any arbitrary vector y through Bx which implies the correctness of this inequality.

Solution to Problem 13.1   Singular value decomposition

Let A be an $n \times m$ matrix with a singular value decomposition $U \Sigma V^T = A$, and the first r singular values non-zero. Recall that the columns of U and V define bases for the various matrix spaces - see Fig. B.1.
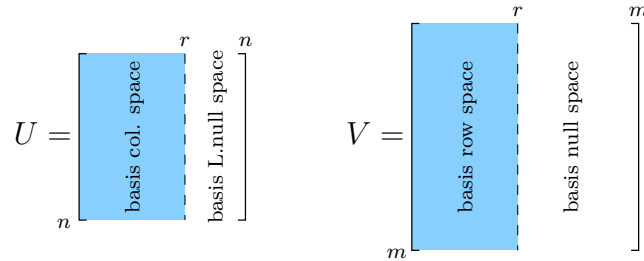
Figure B.1: Basis of matrix spaces using SVD

For the first matrix $\sigma_3 = 0$ and thus the matrix has rank=2. The second matrix is diagonal and the singular values are the magnitudes of the diagonal elements. The third matrix is a vector, in which case the only singular value is the length of the vector. In this case the first column of U is a normalised vector in the same direction.

MATLAB file: `orc13_SVD.m`

Solution to Problem 13.2   Rank of a matrix

The columns (and rows) of the first matrix are linearly dependent and thus it has rank 1. This is also confirmed, from the SVD, since $\sigma_2 = 0$. Same is true for the second matrix. The third matrix has rank=2, but $\sigma_2 = 0$. This is so because from numerical point of view the matrix is still singular. Thus SVD provides more reliable computation of the (numerical) rank of a matrix.

MATLAB file: `orc13_SVD.m`

Solution to Problem 13.3   SVD of an image

The SVD returns the following singular values

$$\begin{bmatrix} 1231.4 & 104 & 5.1 & 3.8 & 3.1 & 2.3 & 2.1 & 1.8 & 0.4 & 0.1 \end{bmatrix}$$

Because $\sigma_7 \gg \sigma_8$ one can truncate the last 2. This would however not reduce the number of bits transmitted. Since $\sigma_2 \gg \sigma_3$ one can leave only the first 2 singular values, in which case 2 matrices has to be transmitted: U_{12} and $S_{12} = \Sigma_{12} V_{12}^T$ correspondingly with $2 \cdot 10 = 20$ and $2 \cdot 10 = 20$ elements or all-together $(10 + 10) \cdot 2 \cdot 8 = 320$ bits. The receiver has to perform the multiplication $U_{12} S_{12}$ to recover the image.

MATLAB file: `orc13_SVD.m`.   See `orc13_SVD_photo.m` for more realistic example.

Solution to Problem 14.1   Singular value plot

The desired singular value plot (sigma plot) can be obtained through `sigma(Gyr)`, where `Gyr` is the closed-loop transfer function with the LQG controller obtained in Problem 11.1. For each frequency $G_{yr}(j\omega)$ is just a (complex) matrix for which SVD can be performed and the singular values plotted over ω. The plot is shown in Fig. B.2.
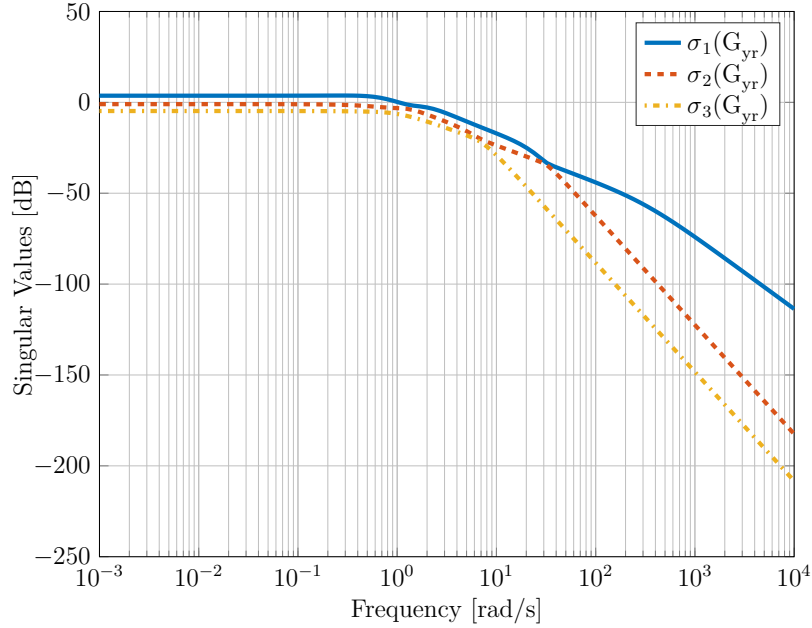

Figure B.2: Singular value plot of the aircraft with LQG controller

The fact that for low frequencies all singular values are close to 0 dB, means that the closed-loop system has static gain (also low-frequency gain) of about 1, which means that good reference tracking can be expected. At high frequencies the singular values drop, which means that noise signals will be suppressed. If the maximum singular value is too large for high frequencies the controller should be redesigned and if this does not help, better sensors needs to be selected for the channels corresponding to its input direction.

Solution to Problem 15.1 H_2 and H_∞ norm

In general the H_2 and H_∞ norms should be computed between the external inputs w and the fictitious outputs z of the closed-loop system $T(s)$, where `T=lft(G,K)`, where `G` is the generalised plant and `K` is the controller. However for the purpose of illustration one can also compute the norms for $G_{yr}(s)$.

The H_2 norm of $T(s)$ is 2.5040 (will vary depending on the chosen matrices Q, R, ...). By using breakpoints in Matlab, one can trace that the function `norm(T,2)` calls the function `normh2.m`, which first solves (15.8) and than uses (15.6) to compute the norm.

The H_∞ norm of $T(s)$ turns out to be 2.4939. Again using breakpoints one can find that the computation is performed in `norminf.m`. Two methods for computing H_∞ were discussed in the script: (A) gridding over frequencies and computing the singular value decomposition for each frequency and (B) using the Hamiltonian matrix M_γ and performing bisection steps over γ. The method implemented in Matlab uses a fast algorithm, proposed by Bruisma and Steinbuch, that combines the above two.

Solution to Problem 15.2   Two LQG designs

The design procedure is similar to the one in Problem 11.1, but no reference signal is considered in the generalised plant. Having constructed the two generalised plants corresponding to the two designs the function `h2syn.m` can be used to design the controllers. The H_2 norm returned by the function is equal to the LQG performance index V. For the first design it is 9.45 and for the second 792.5. The singular value plots of the closed-loop transfer function from r to y with the two controllers is given in Fig. B.3.
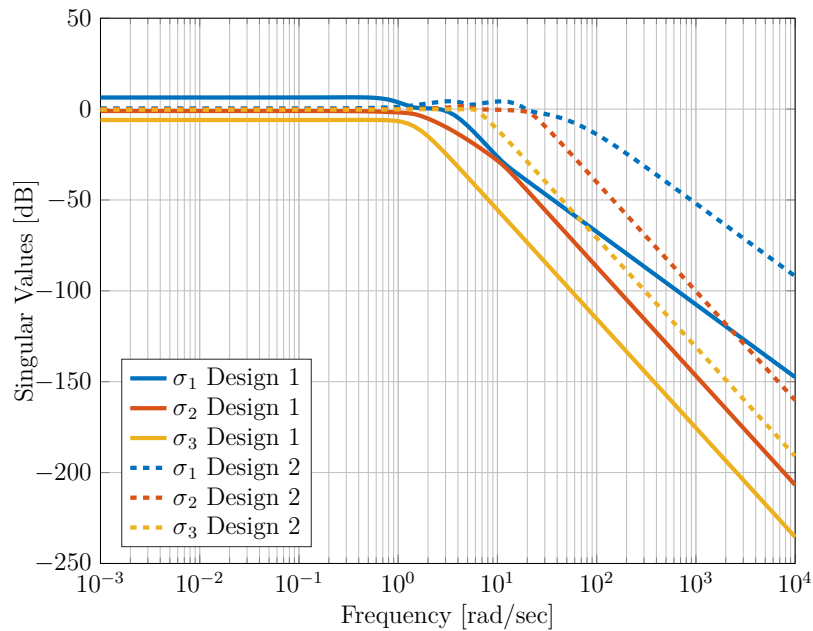
Figure B.3: Singular value plot of the aircraft with two LQG controllers

At low frequency the singular values of $G_{yr2}(s)$ lie very close to the 0 dB line, where as with the first design they are on ± 5 dB. Thus we expect the second design to have better reference tracking. At high frequencies the singular values with the second design start decreasing later than with the first design showing higher bandwidth, but also higher sensitivity to noise. Further at frequencies between 10 and 100 rad/s the increase above 1 of all singular values with the second design indicate that noise signals in this frequency range will be actually amplified. In summary we should expect faster system response and better reference tracking with the second design, but also higher sensitivity to high

frequency noise. This can be validated by performing closed-loop simulations.

Matlab file: `orc15_LQGsigma.m`

Solution to Problem 15.3   Computing the H_2 norm

The observability and controllability gramians of $T(s)$ can be computed by `Wo = gram(T,'o')` and `Wc = gram(T,'c')` respectively. Then the H_2 norm can be simply obtained by `H2o = sqrt(trace(B'*Wo*B))` or `H2c = sqrt(trace(C*Wc*C'))`.

When using the impulse response it is important that the integration is performed correctly. The easiest way to do so is to use a fixed step size when computing the impulse response, e.g. `[g,t] = impulse(T,0:0.01:2)` and later multiply the sum by the step size (i.e., 0.01).

Matlab file: `orc15_H2.m`

Solution to Problem 15.4   Computing the H_∞ norm

As discussed in the solution of Problem 15.1 there are several ways to compute the H_∞ norm using M_γ. The easiest to implement is the "brute-force" approach, where one starts from high γ and with a fixed step size decreases it until M_γ becomes with imaginary eigenvalues. A much faster results can be obtained using the bisection method, summarised in the following algorithm:

1. Choose $\gamma_{\max}$, $\gamma_{\min}$ and tolerance γ_{tol} (e.g. $\gamma_{\max} = 1000$, $\gamma_{\min} = 0$, $\gamma_{tol} = 10^{-6}$)

2. Set $\gamma = (\gamma_{\max} + \gamma_{\min})/2$

3. If $\gamma_{\max} - \gamma_{\min} < \gamma_{tol}$ Stop!

4. Construct M_γ and compute its eigenvalues

5. If M_γ has eigenvalues on the imaginary axis set $\gamma_{\min} = \gamma$, else set $\gamma_{\max} = \gamma$

6. Return to step 2

The H_∞ norm is equal to γ ($\pm\gamma_{tol}/2$).

Matlab file: `orc15_Hinf.m`

Solution to Problem 16.1   Three block approximation

From the definition of the H_∞ norm

$$\left\| \begin{bmatrix} W_S S \\ W_T T \\ W_K K S \end{bmatrix} \right\|_\infty < 1 \quad \text{is equivalent to} \quad \sup_\omega \bar{\sigma} \left(\begin{bmatrix} W_S(j\omega)S(j\omega) \\ W_T(j\omega)T(j\omega) \\ W_K(j\omega)K(j\omega)S(j\omega) \end{bmatrix} \right) < 1$$

Consider the SISO case. Then using that for a vector

$$\bar{\sigma}(x) = \sqrt{\lambda_{\max}(x^H x)} = \sqrt{x^H x} = \sqrt{|x_1|^2 + \cdots + |x_n|^2}$$

follows

$$\sup_\omega \sqrt{|W_S S|^2 + |W_T T|^2 + |W_K K S|^2} < 1$$

which has a maximum approximation error when all three terms are equal, i.e.

$$\sup_\omega \sqrt{3|W_S S|^2} < 1$$

or $\sup_\omega \sqrt{|W_S S|^2} = \|W_S S\| < 1/\sqrt{3} = 0.577$, which results in -4.8 dB approximation error.

Solution to Problem 16.2   Sensitivity and input directions

The singular value plot of $S(s)$ for the LQG design from Problem 14.1 is shown in Fig. B.4. One can clearly see that in one of the input directions (corresponding to the minimum singular value) the low-frequency sensitivity is low, meaning good reference tracking. However the other two singular values are large (one even close to 2 dB) showing poor disturbance rejection.

For example for $\omega = 10^{-3}$ rad/s the maximum and minimum singular values are 0.9802 and 0.0002 and the corresponding input directions

$$v_1 = \begin{bmatrix} -0.0007 & -0.0015 + 0.5704j & -0.0025 + 0.8213j \end{bmatrix}^T$$
$$v_3 = \begin{bmatrix} 1.0000 & 0 + 0.0001j & 0 + 0.0008j \end{bmatrix}^T$$

Thus disturbances entering through the first input have less effect, than combination of disturbances acting on all inputs.

Matlab file: `orc16_sensitivity.m`

Solution to Problem 16.3   Waterbed effect

The two systems visualise the waterbed effect. Note that the integrator in $G_1(s)$ is only there to give zero-steady state error for the step responses - the waterbed effect will be also evident, for example $G(s) = \dfrac{2}{(s+1)(s+2)}$
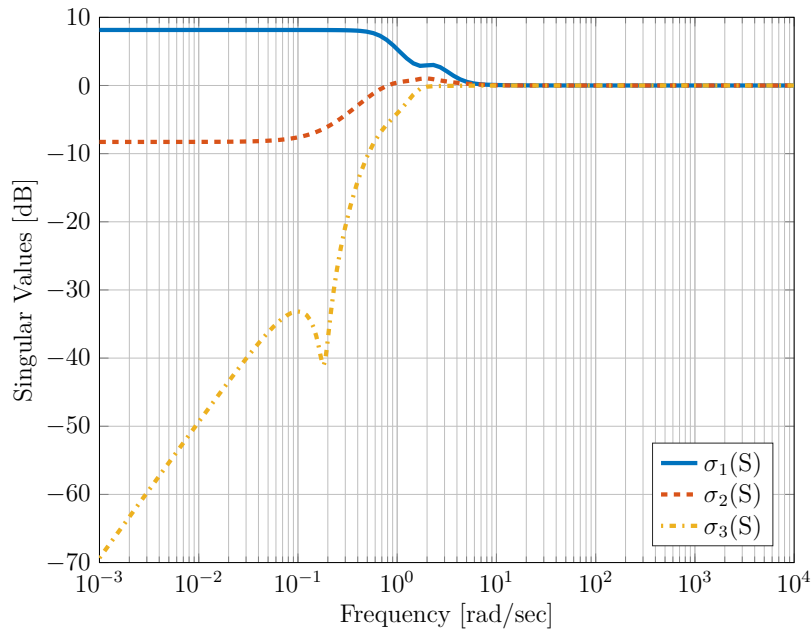
Matlab file: `orc16_waterbed.m`

Figure B.4: Singular value plot of the sensitivity of the aircraft with LQG controller

Solution to Problem 17.1   Mixed sensitivity design

The fact that $G(s)$ and $G_d(s)$ share common dynamics means that the disturbance is not acting on the plant's output, but rather enters inside the plant, as apparent from Fig. 17.25. The block diagram can be re-arranged to the one in Fig. B.5.
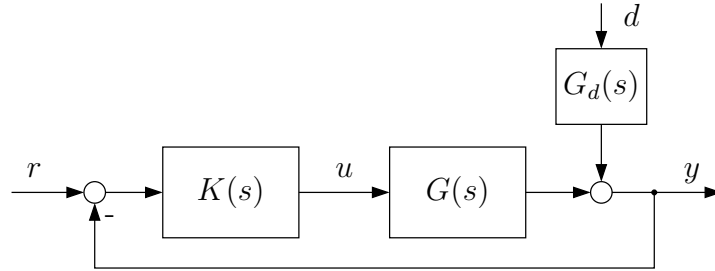

Figure B.5: System block diagram

At a first time we can "neglect" the disturbance input during the design process and concentrate only on the reference tracking. However the disturbance rejection properties with the obtained controller have to be analysed and changes to the design might be needed.

For the purpose of reference tracking the suggested problem formulation with sensitivity and control sensitivity filters can be used. The generalised plant is shown in Fig. B.6.

To construct the generalised plant one needs to select the $W_S(s)$ and $W_K(s)$ filters. To
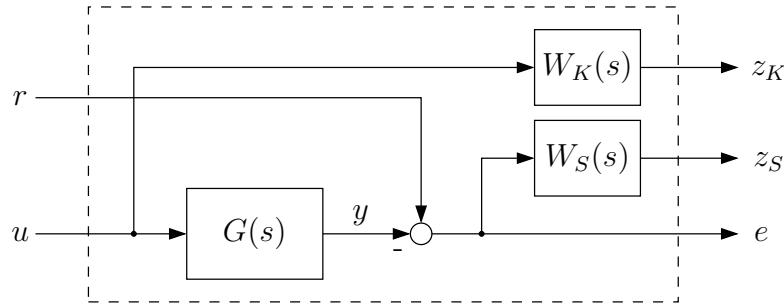
Figure B.6: Generalised plant for the H_∞ design

meet the design requirements several iterations over/adjustments of the filters might be needed. From the requirements for reference tracking we know that the sensitivity $S(s)$ should be low at low frequencies. Since the H_∞ design algorithm will aim on achieving $\|W_S(s)S(s)\|_\infty < 1$ this means that the gain of $W_S(s)$ should be high at low frequencies. Further it is known, that $S(s)$ cannot be kept low at all frequencies, i.e. for high frequencies the gain of $W_S(s)$ should become low. A reasonable initial filter $W_S(s)$ is a first order (low-pass) system

$$W_S(s) = N_S \frac{w_S}{s + w_S} = \frac{1}{M_S} \frac{w_S}{s + w_S}$$

Notice that N_S directly sets the gain of $W_S(s)$ for low frequencies, or alternatively M_S sets an upper-boundary on $S(s)$ for low frequencies.

In order to select M_S and w_S one can have a look the sigma plot of $G_d(s)$, shown in Fig. B.7. The disturbance is a low-frequency one, has low-frequency gain of 10 and a crossover frequency of 0.1 rad/s. Thus select $M_S = 1/10$ and $w_S = 0.1$ rad/s. A good initial guess for the control sensitivity is $W_K(s) = 1$.

The generalised plant can be constructed with the following code

```
systemnames = 'G Ws Wk';
inputvar    = '[r; u]';
input_to_G  = '[u]';
input_to_Wk = '[u]';
input_to_Ws = '[r-G]';
outputvar   = '[Wk; Ws; r-G]';
GSYS = sysic;
```

The H_∞ controller can be designed with the function `hinfsyn`
`[K, Gcl, hinf] = hinfsyn(GSYS, nmeas, ncont)`

where **nmeas** and **ncont** are the number of measured outputs and controlled inputs (in this case both equal 1); `Gcl` is the closed-loop of the generalised plant and the controller; `hinf` is the H_∞ norm of `Gcl`, i.e. $\texttt{hinf} = \left\| \begin{bmatrix} W_S S \\ W_K K S \end{bmatrix} \right\|_\infty$
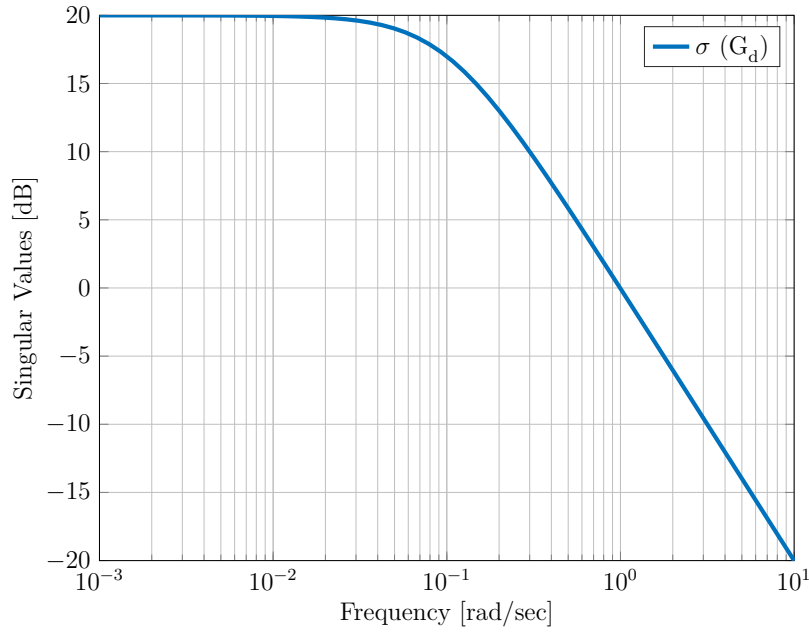
Figure B.7: Sigma plot of $G_d(s)$

Simulating the closed-loop response with the obtained controller shows that the system response to a reference step is slow and the control input is not efficiently used $u(t) <$ 0.3. In order to improve these one can further reduce M_S to $M_S = 0.01$, construct the generalised plant and repeat the design. Now the reference response satisfies the design requirements, but the disturbance rejection is poor. "Pushing down" the sensitivity by selecting $M_S = 0.001$ and setting $w_S = 0.01$ rad/s yields a controller satisfying the design requirements posed in the problem statement. The sensitivity and control sensitivity are shown in Fig. B.8 together with the sigma plots of the inverse filters. The fact that `hinf` is greater than 1 indicates that in the filters are "imposing" a performance criteria that cannot be met. By appropriately re-scaling the filters the same controller can be obtained, but with norm equal to 1.

Matlab file: `orc17_Hinf_design.m`

The disturbance rejection of the obtained controller are however not satisfactory for the stronger disturbance $G_d(s) = 100/(10s + 1)$. By selecting a stronger sensitivity filter (smaller M_S) the disturbance rejection can be improved by at the price of violating the actuator constraints.

One approach to simultaneously address both the reference tracking and the disturbance rejection is to take into account both the transfer functions from reference to output and control action, and from disturbance input to output and control action. For this purpose, the generalised plant in Figure B.9 can be used.

Combining all the individual transfer functions to a single one and minimising its H_∞
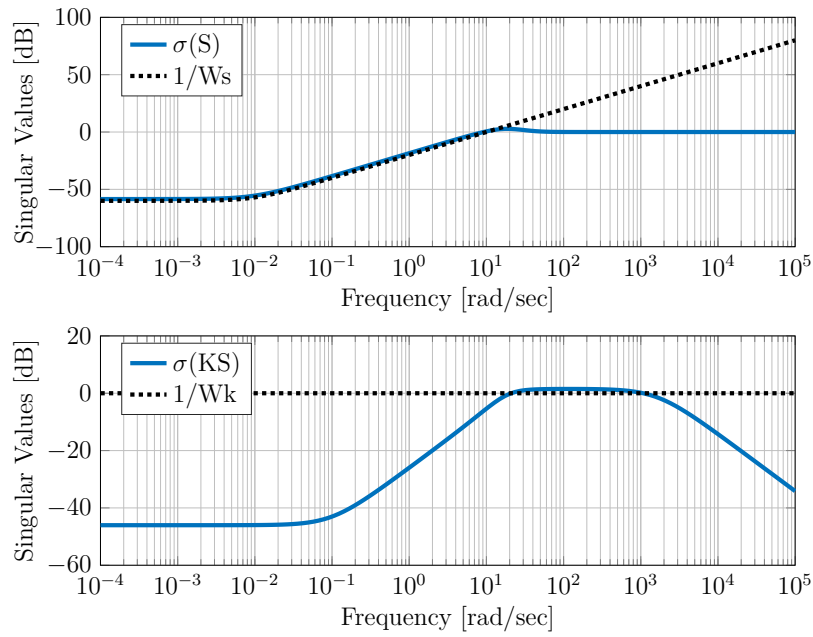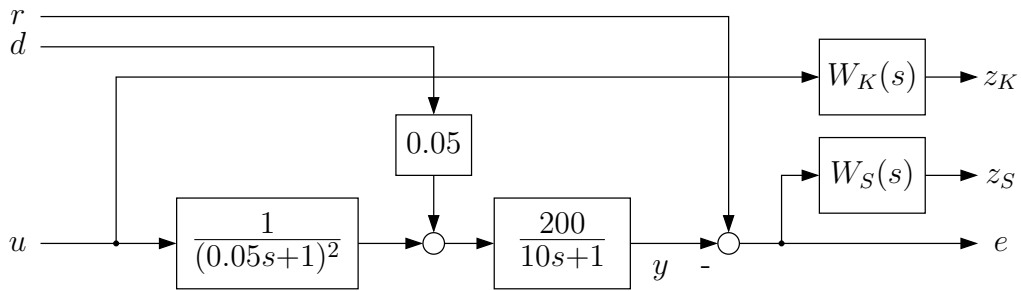
Figure B.8: Sigma plot of $S(s)$ and $K(s)S(s)$ with the obtained controller

Figure B.9: Generalised plant with reference and disturbance inputs

norm results in the so called 4-block design

$$\left\| \begin{bmatrix} W_S S & W_S S G_d \\ W_K K S & W_K K S G_d \end{bmatrix} \right\|_\infty \tag{B.1}$$

This can be easily seen by closing the loop $u = Ke$, computing the transfer function and using that $\| - G(s)\|_\infty = \|G(s)\|_\infty$

$$\begin{bmatrix} z_S \\ z_K \end{bmatrix} = \begin{bmatrix} W_S S & -W_S S G_d \\ W_K K S & -W_K K S G_d \end{bmatrix} \begin{bmatrix} r \\ d \end{bmatrix}$$

Performing the design with this new generalised plant, without re-configuring the filters results in a controller that has excellent disturbance rejection, but high overshoot and control signal to reference tracking - Fig. B.10. The singular value plots of the four transfer functions in (B.1) are shown in Fig. B.11. As can be seen both S and SG_d have
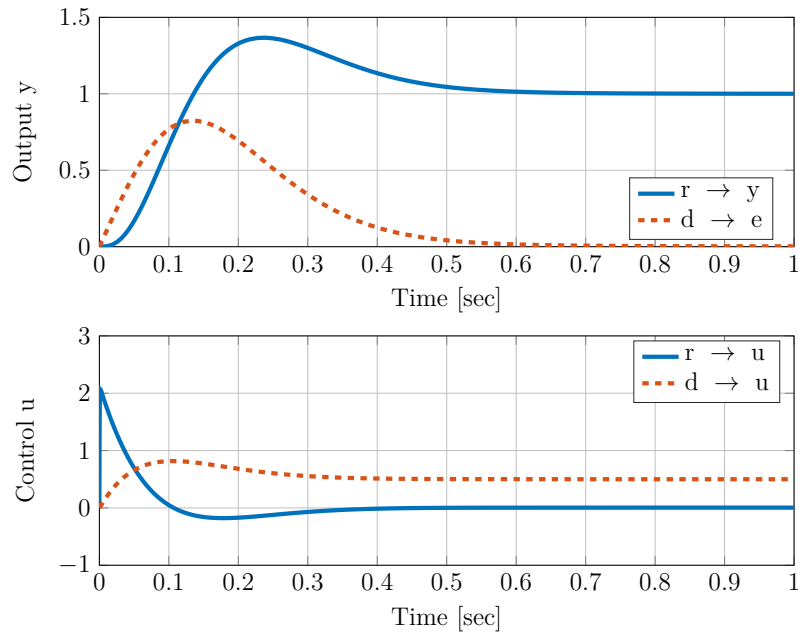
Figure B.10: Step and disturbance responses $W_K = 1$, $W_S(s) = \frac{1}{0.001} \frac{1}{s/0.01+1}$

peak values over 1 at frequency of approx. 10 rad/s, which is known to be the reason for the overshoot in the reference step response. One way to reduce this peak values of S and SG_d is by using the $W_K(s)$ filter. Another option is to use $W_S(s)$ filter and impose an upper limit for the sensitivity also for high frequencies.
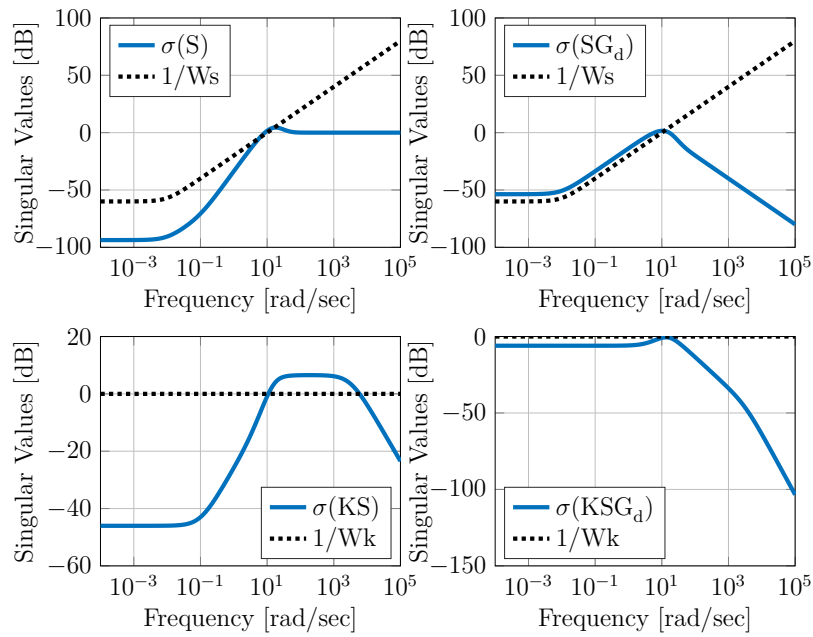

Figure B.11: Sigma plots by the 4-block design

For example using the bi-proper sensitivity filter

$$W_S(s) = \frac{1}{M_S} \frac{s/(c\omega_s) + 1}{s/\omega_s + 1}$$

with $\omega_s = 0.01$ rad/s, $M_S = 0.02$ and $c = 80$ obtains a controller which achieves closed-loop performance close to the required - see Figures B.12 and B.13.
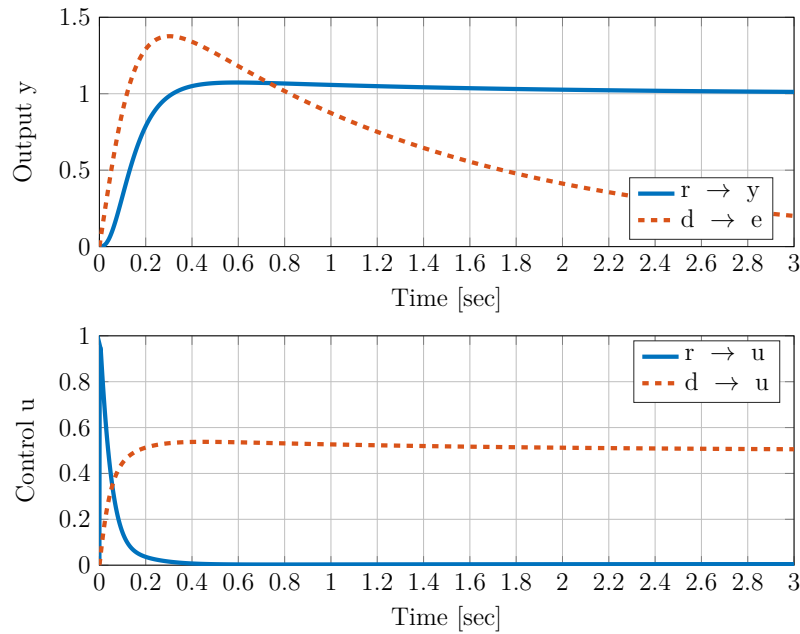

Figure B.12: Step and disturbance responses for the second 4-block design

Matlab file: `orc17_Hinf_design_2.m`

However close, meeting the design requirements remains a difficult task. Partially the problem is due to the presence of 4 transfer functions that have to be shaped, but only 2 design parameters $W_S(s)$ and $W_K(s)$. This problem can be easily remedied using prefilters for the reference signal and the disturbance signal, as shown in Fig. B.14.
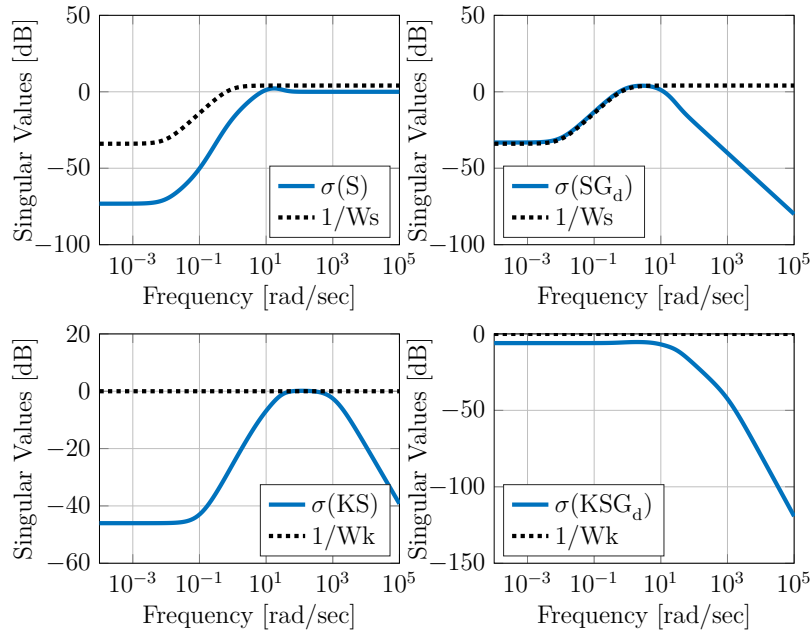
Figure B.13: Sigma plots for the second 4-block design
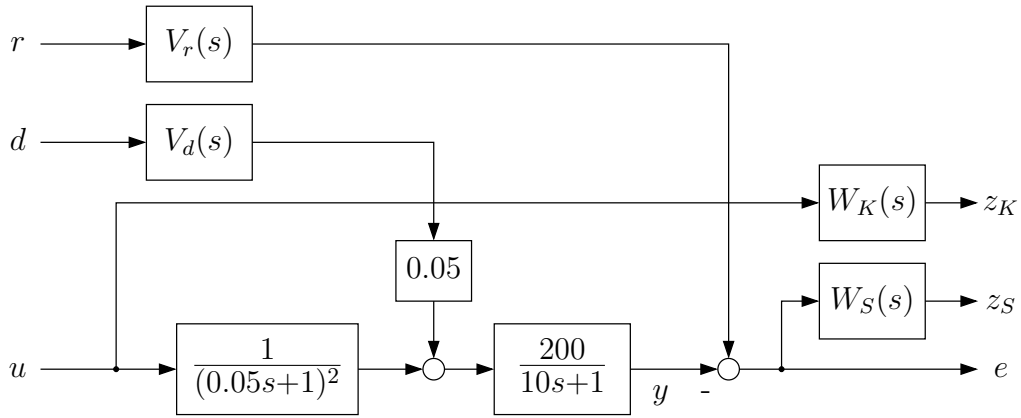
In this case, the 4-block design becomes

Figure B.14: Generalised plant with prefilters for the reference and disturbance inputs

$$\left\| \begin{bmatrix} W_S S V_r & W_S S G_d V_d \\ W_K K S V_r & W_K K S G_d V_d \end{bmatrix} \right\|_\infty \tag{B.2}$$

Matlab file: `orc17_Hinf_design_3.m`

In order to meet the design requirements one can try using higher order shaping filters and shaping the complementary sensitivity.

Matlab file: `orc17_Hinf_design_4.m`

During the preparation of this solution no controller was found meeting all of the design requirements. Thus the questions whether such controller exists and how to find it are still open.

Solution to Problem 17.2 H_∞ control of an aircraft

The H_∞ controller designs presented in the lecture and some improved designs are available in the Matlab solution file. In order to achieve both good reference tracking and disturbance rejection a 4-block design *similar* to the one presented in the solution of Problem 17.1 will be needed. This time, however the inputs to the generalised plant are the reference signal r and the *input disturbance d*, see Fig. B.15.

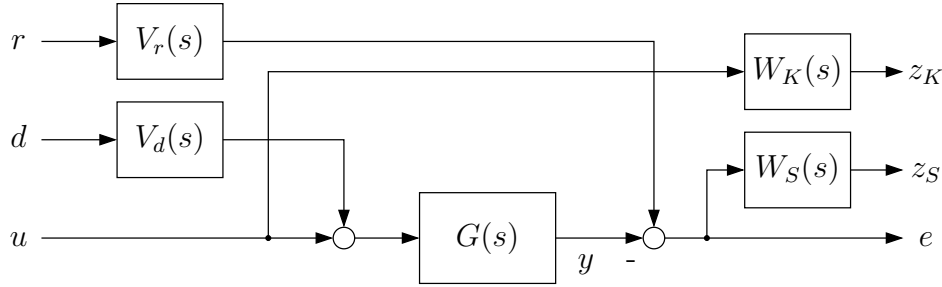Figure B.15: Generalised plant with reference input and input disturbance

Matlab file: `orc17_Hinf_aircraft.m`

a) Ignoring the input disturbance and the noise effects, Design 4 from Chapter 17 can be modified to satisfy the actuator constraints by slightly reducing the bandwidth requirements (reducing ω_S). The step response is shown in Fig. B.16, with settling time for the relative altitude (y_1) of approximately 1.2 seconds.

Design 6:

ω_S	M_S	ω_K	M_K
$2.5 \cdot 10^{-4}$	10^{-4}	10^2	10^2

Note, that since the design requirements do not specify actuator constraints for inputs u_1 and u_2 one can set the $W_K(s)$ as

$$W_K(s) = \begin{bmatrix} \delta & 0 & 0 \\ 0 & \delta & 0 \\ 0 & 0 & \frac{1}{M_K}\frac{s/\omega_K+1}{s/(c\omega_K)+1} \end{bmatrix}$$

where $\delta \geq 0$ is a small number. Then the settling time can be reduced to approx. 1 sec, but at the expense of (unrealistically) high control in the first two input channels.

b) The system response to a sinusoidal measurement noise with amplitude of 0.1 and frequency of 10^4 rad/s (injected in all outputs) is shown in Fig. B.17.

Note that since the transfer function from noise to output is $S(s)$, and the sensitivity is close to 1 for large frequencies the noise is not attenuated. This is natural and cannot be avoided. However, since the system will not respond to this high-frequency noise, there is not a reason for the controller to attempt to compensate
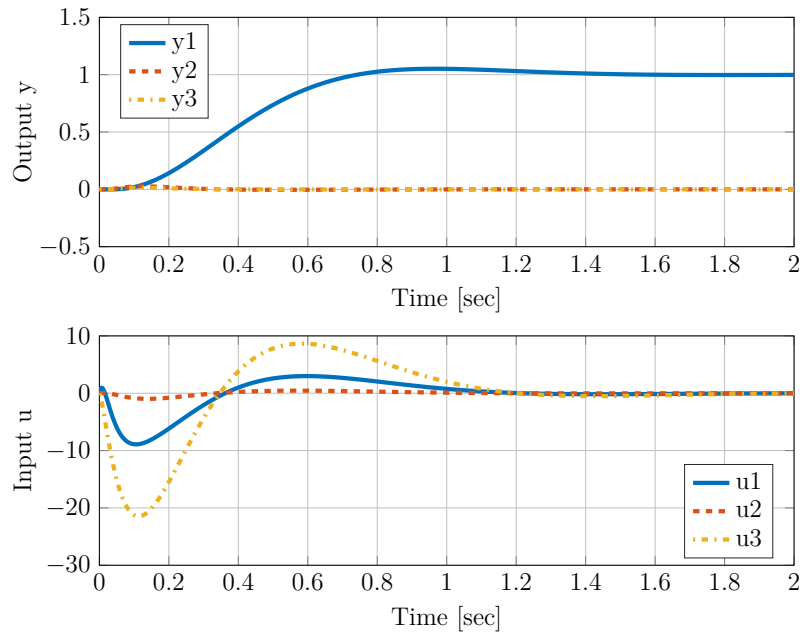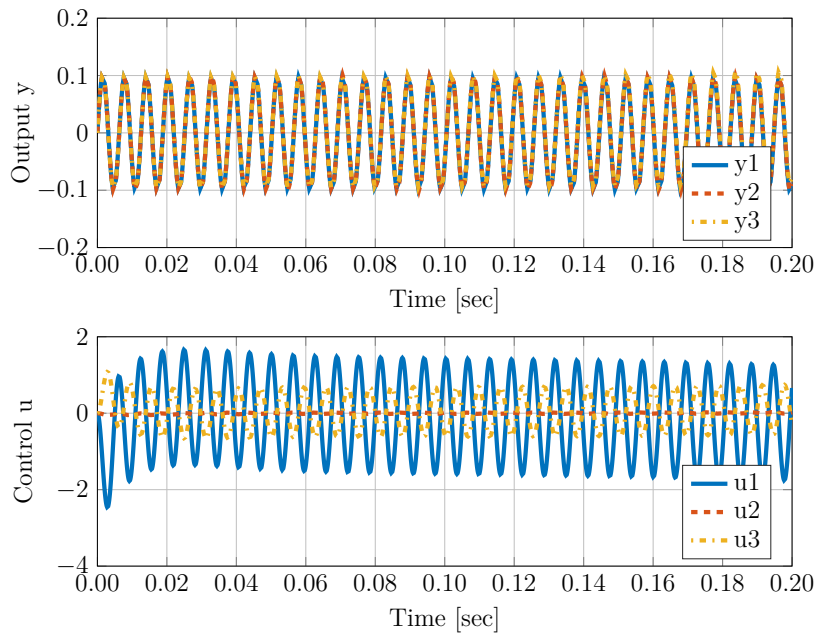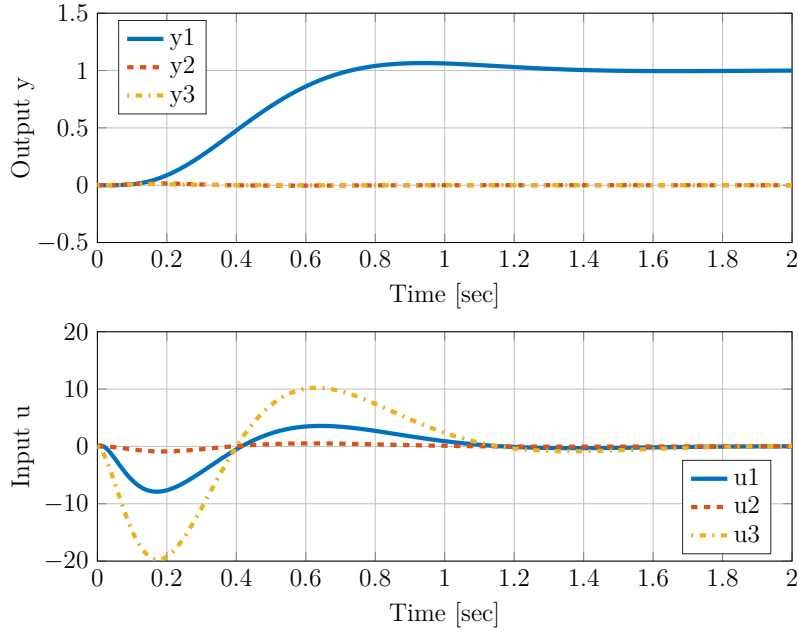
Figure B.16: Design 6: response to $r(t) = [\sigma(t)\ \ 0\ \ 0]^T$

Figure B.17: Design 6: response to a sinusoidal noise injected in all outputs

it. Thus the only improvement of interest is a reduction of the magnitude of the control signals. This can be achieved by reducing ω_K, and compensating the resulting slower-system response by slight increase in ω_S. The tracking performance and the noise response are shown in Fig. B.18 and Fig. B.19.

Design 7:

ω_S	M_S	ω_K	M_K
$3.5 \cdot 10^{-4}$	10^{-4}	10	10^2

Figure B.18: Design 7: response to $r(t) = [\sigma(t) \ \ 0 \ \ 0]^T$

c) The system response to a step input disturbance applied to the spoiler angle (u_1) is shown in Fig. B.20. It is clear, that whereas the disturbance has no big effect on the y_2 and y_3 it leads to significant oscillations in the relative altitude y_1 and to a steady-state error.

To improve the disturbance rejection a generalised plant with only disturbance input (see Fig. 17.19) can be used, but the resulting controllers will have poor reference tracking. To achieve both good reference tracking and disturbance rejection a generalised plant with both r and d as inputs can be constructed, as shown in Fig. B.15, resulting in the 4-block design (2 inputs (r and d) times 2 outputs (z_K and z_S)). The pre-filters $V_r(s)$ and $V_d(s)$ are added to allow for independent shaping of the four transfer functions

$$\left\| \begin{bmatrix} W_S & 0 \\ 0 & W_K \end{bmatrix} \begin{bmatrix} S & SG \\ KS & KSG \end{bmatrix} \begin{bmatrix} V_r & 0 \\ 0 & V_d \end{bmatrix} \right\|_\infty = \left\| \begin{bmatrix} W_S S V_r & W_S SG V_d \\ W_K KS V_r & W_K KSG V_d \end{bmatrix} \right\|_\infty .$$

Designing a controller with this new generalised plant using the parameters from Design 7 and identity gains as pre-filters ($V_r(s) = I_3$, $V_d(s) = I_3$) leads to closed-loop system with large overshoot at reference step, but very good disturbance rejection capabilities, as shown in Fig. B.21 (notice the different time-scale of the disturbance response plots compared to Fig. B.20).
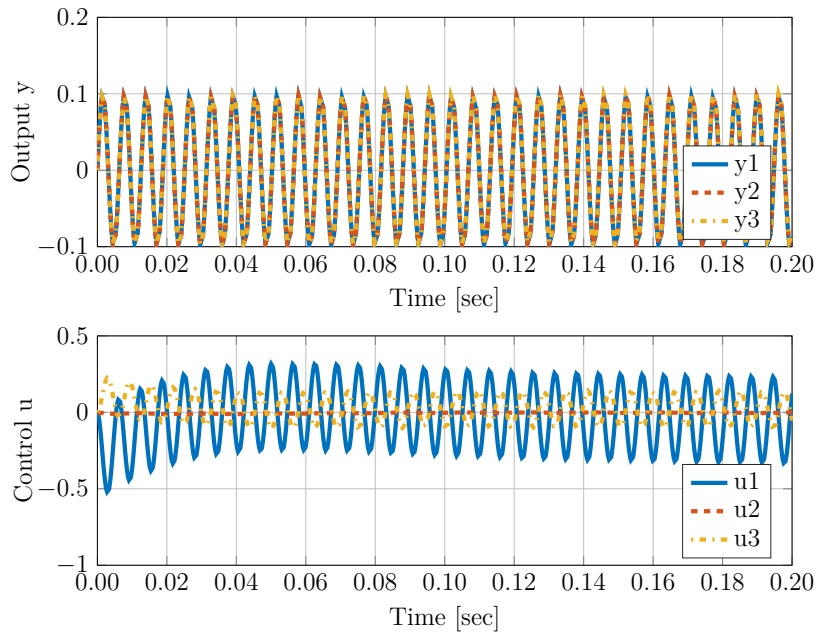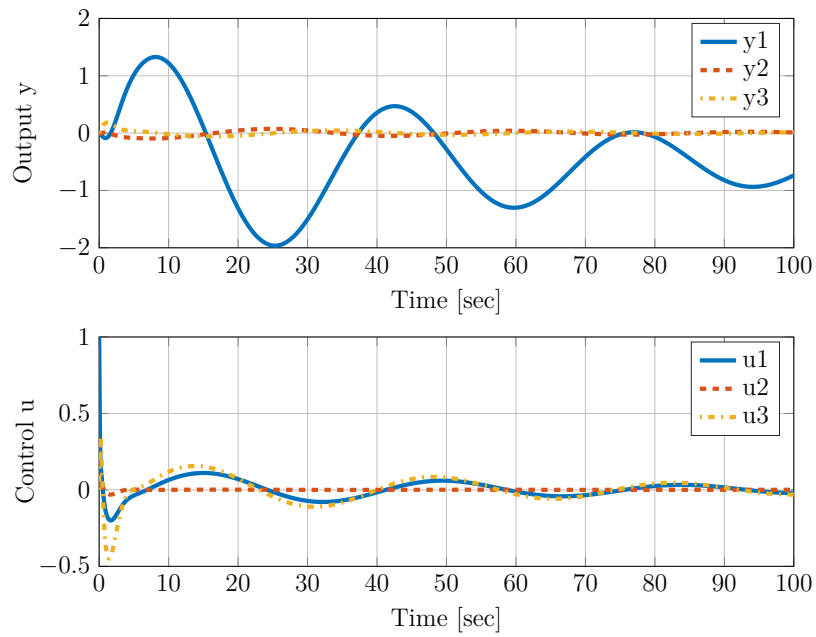
Figure B.19: Design 7: response to a sinusoidal noise injected in all outputs

Figure B.20: Design 7: response to $d(t) = [\sigma(t) \;\; 0 \;\; 0]^T$

The following steps were taken to obtain a better controller

- $V_d(s) = 0.01\,I_3$ - set smaller priority on the disturbance rejection: reduces the reference tracking overshoot (at the expense of slower disturbance rejection), but also results in very slow convergence of y_1 to the reference value.
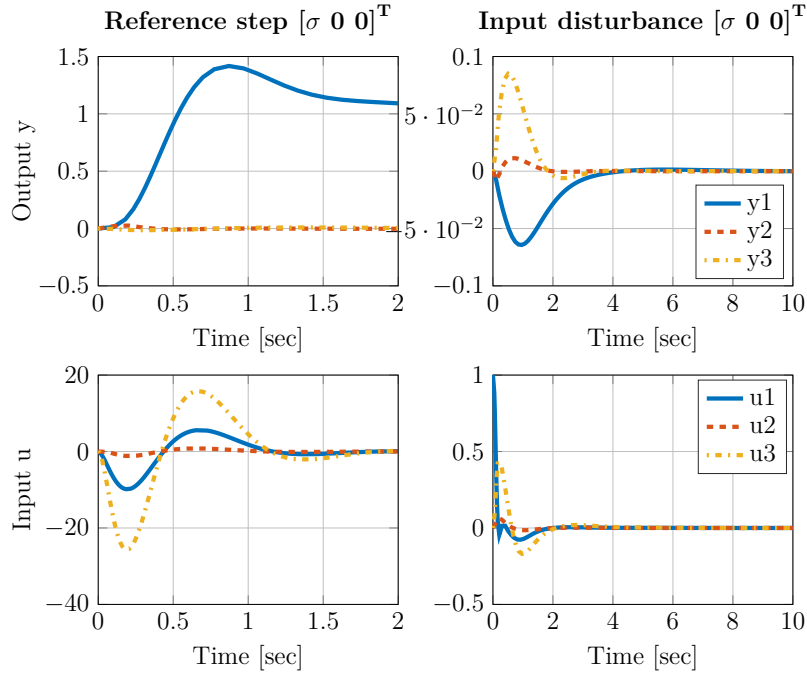
Figure B.21: Design 7: responses after 4-block design

  − $W_S(s) = \begin{bmatrix} 5w_S & 0 & 0 \\ 0 & w_S & 0 \\ 0 & 0 & 10w_S \end{bmatrix}$ - give higher value to the error in the first channel:
results fast tracking, reasonable disturbance rejection (with no steady-state error), but violates $|u_3| \leq 20$.

  − $\omega_S = 3 \cdot 10^{-5}$ - reduce the bandwidth requirement: makes the system response slower, but reduces the 3rd input to satisfy the actuator constraints. Shows cross-coupling between the outputs (reference step in y_1 results in some response in y_2 and y_3).

With the so-obtained design parameters, shown below, the closed-loop system has the response shown in Fig. B.22.

Design 8:

ω_S	M_S	$W_S(s)$			ω_K	M_K	$W_K(s)$			V_r	V_d
$3 \cdot 10^{-5}$	10^{-4}	$\begin{bmatrix} 5w_S & 0 & 0 \\ 0 & w_S & 0 \\ 0 & 0 & 10w_S \end{bmatrix}$			10	10^2	$\begin{bmatrix} w_K & 0 & 0 \\ 0 & w_K & 0 \\ 0 & 0 & w_K \end{bmatrix}$			I_3	$0.01\, I_3$

where

$$w_S(s) = \frac{1}{M_S} \frac{1}{\frac{s}{\omega_S} + 1} = \frac{1}{M_S} \frac{\omega_S}{s + \omega_S}$$

$$w_K(s) = \frac{1}{M_K} \frac{\frac{s}{\omega_K} + 1}{\frac{s}{c\omega_K} + 1} = \frac{c}{M_K} \frac{s + \omega_K}{s + c\omega_K},$$
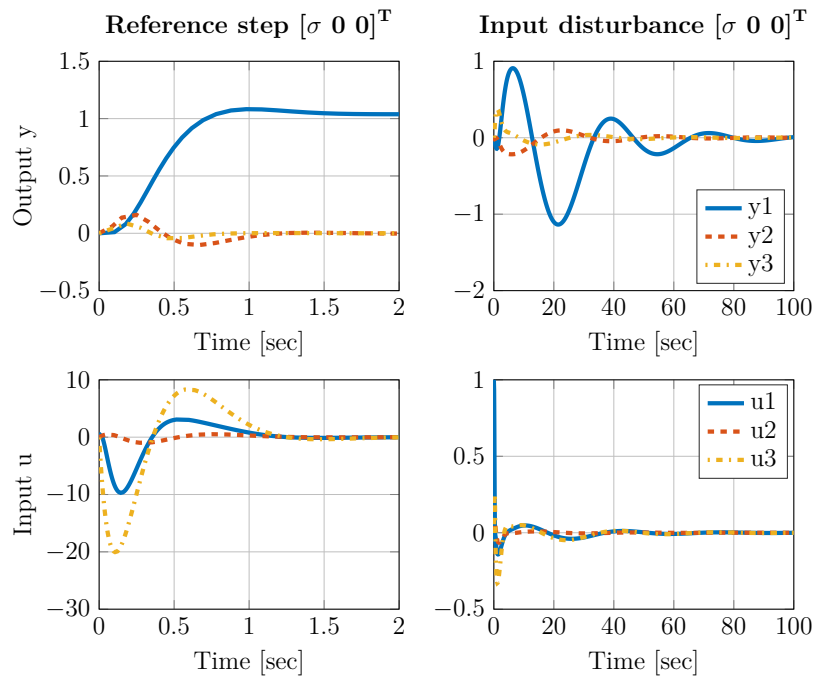
Figure B.22: Design 8: responses after 4-block design

Matlab file: `orc17_Hinf_aircraft_2.m`

Solution to Problem 18.1   Pole regions using LMIs

Start from the requirement that the poles have real part between α_l and α_r, see Fig. B.23 (right). Let $s = x + jy$ be a point inside the region. Then $\alpha_l < x < \alpha_r$. This can be rewritten as 2 inequalities

$$2\alpha_l - 2x < 0 \qquad \text{and} \qquad 2x - 2\alpha_r < 0$$

On another hand taking the 2×2 matrices L and M results in

$$\begin{bmatrix} l_{11} & l_{12} \\ l_{12} & l_{22} \end{bmatrix} + \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}(x + jy) + \begin{bmatrix} m_{11} & m_{21} \\ m_{12} & m_{22} \end{bmatrix}(x - jy) < 0$$

$$\begin{bmatrix} l_{11} + m_{11}2x & l_{12} + m_{12}(x + jy) + m_{21}(x - jy) \\ l_{12} + m_{21}(x + jy) + m_{12}(x - jy) & l_{22} + m_{22}2x \end{bmatrix} < 0$$

The last can be used to express the desired 2 inequalities by selecting $l_{12} = 0$, $m_{12} = m_{21} = 0$ (to get the off-diagonal terms equal to zero), $l_{11} = 2\alpha_l$, $m_{11} = -1$ and $l_{22} = -2\alpha_r$, $m_{22} = 1$, resulting in the matrices

$$L = \begin{bmatrix} 2\alpha_l & 0 \\ 0 & -2\alpha_r \end{bmatrix}, \qquad M = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Figure B.23: LMI regions

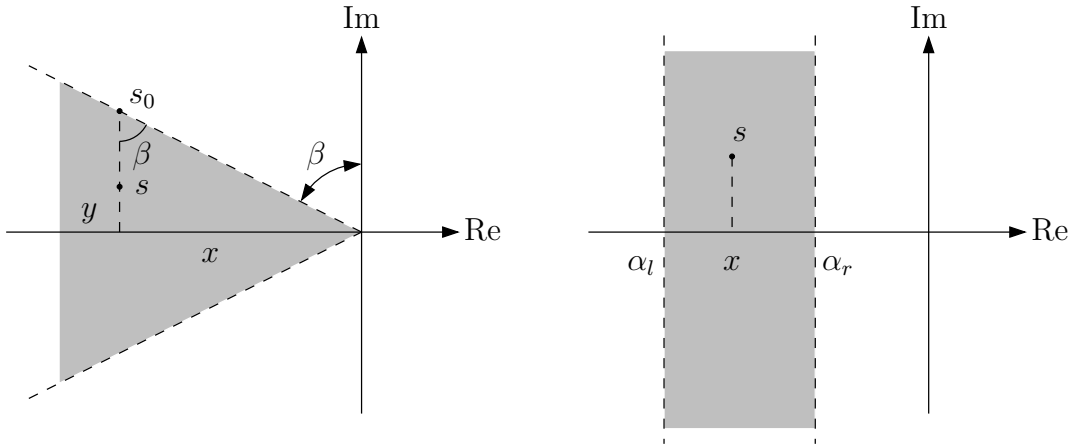Now consider the cone region in Fig. B.23 (left), and a point $s = x + jy$ in it. Let $s_0 = x + jy_0$ be a point at the boundary of the region. Then

$$\tan \beta = \frac{|x|}{|y_0|} < \frac{|x|}{|y|}, \qquad \text{for all } s \in \mathcal{D}$$

This can be rewritten as

$$\tan^2 \beta = \frac{\sin^2 \beta}{\cos^2 \beta} < \frac{x^2}{y^2} \qquad \Rightarrow \qquad x^2 \cos^2 \beta - y^2 \sin^2 \beta > 0$$

Further, $x < 0$ for all $s \in \mathcal{D}$. Since $\beta \in \begin{bmatrix} 0 & 90° \end{bmatrix}$, $\cos \beta > 0$ and thus $x \cos \beta < 0$ for all $s \in \mathcal{D}$. The problem is now write the above 2 conditions in the form

$$K = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} < 0$$

Using Schur complement, K can be written as

$$k_{22} < 0, \qquad k_{11} - k_{12}k_{22}^{-1}k_{21} < 0$$

Multiplying the second equation by k_{22}, which is a negative number, yields $k_{11}k_{22} - k_{12}k_{21} > 0$. The similarity with the desired inequalities becomes noticeable.

$$
\begin{array}{ll}
k_{22} < 0 & x \cos \beta < 0 \\
k_{11}k_{22} - k_{12}k_{21} > 0 \qquad & x^2 \cos^2 \beta - y^2 \sin^2 \beta > 0
\end{array}
$$

To obtain the desired matrices, recall that $s + \bar{s} = 2x$ and $s - \bar{s} = 2jy$. Thus select

$$\begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} = 2 \begin{bmatrix} x \cos \beta & -jy \sin \beta \\ jy \sin \beta & x \cos \beta \end{bmatrix} = \begin{bmatrix} \cos \beta & -\sin \beta \\ \sin \beta & \cos \beta \end{bmatrix} (x + jy) + \begin{bmatrix} \cos \beta & \sin \beta \\ -\sin \beta & \cos \beta \end{bmatrix} (x - jy)$$

and therefore

$$L = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad M = \begin{bmatrix} \cos \beta & -\sin \beta \\ \sin \beta & \cos \beta \end{bmatrix}.$$

Here, the Schur complement is applied to the Hermitian matrix K. In problem 4.2, The Schur complement has been defined for the symmetric matrices and it is straightforward to extend the result of this problem to the case of Hermitian matrices.

Solution to Problem 18.2 $\quad H_2$ and H_∞ norm computation via LMIs

Let's start from the H_∞ norm computation. Theorem 18.3 gives a condition to check, whether the H_∞ norm is smaller than γ. Thus, the smallest γ for which the LMI in (18.22) holds will be equal to the H_∞ norm. Therefore one can compute the norm, by solving the following problem

$$\text{minimise } \gamma \qquad \text{subject to} \quad \begin{bmatrix} A^T P + PA & PB & C^T \\ B^T P & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0, \quad P = P^T > 0.$$

This can be solved using the `mincx` function of the LMI lab toolbox in MATLAB, which can solve problems of the type

$$\text{minimise } c^T p \qquad \text{subject to} \quad M(p) < 0,$$

where $M(p) = \sum_k M_k p_k$ is an LMI in p.

Start by defining the beginning of a set of LMIs: `setlmis([]);` which is always the first command. Next, define the decision variables $P = P^T$ and γ.

```
P = lmivar(1,[nstate 1]);   % symmetric, [n x n    full block]
g = lmivar(1,[1 0]);        % symmetric, [1 x 1       scalar]
```

where n is the dimension of A. If instead of a symmetric a full-block $n \times m$ matrix Q has to be defines, this can be done via `Q = lmivar(2, [n m])`.

Next, define the beginning of a new LMI by `Mk = newlmi();`. One can enumerate the LMIs manually, but it is recommended to do it automatically to avoid mistakes. Now the elements of the $M_k < 0$ can be defined. Since M_k is a symmetric matrix, only one of the off-diagonal elements should be specified, that is either $M_{k_{ij}}$ or $M_{k_{ji}}$, but not both.

The LMIs are specified with the command `lmiterm`, having the following syntax:

```
lmiterm([Mk i j X],A,B,'s');
```

The meaning of the variables is as follows: `Mk` that the term has to be added to the k-th LMI; `i j` indicates the position in the LMI, that is $M_{k_{ij}}$; `X` is variable; `A` and `B` are matrices multiplying `X` by the left and right, that is $M_{k_{ij}} = AXB$. If present `'s'` indicates, that the term is symmetric: $M_{k_{ij}} = AXB + B^T X^T A^T$. Since `lmiterm` assumes, that the LMI is $M_k < 0$, if $0 < M_k$ has to be defined this can be done by writing `-Mk`. Finally a constant term $M_{k_{ij}} = C$ can be defined as `lmiterm([Mk i j 0],C);`.

The LMIs for computing the H_∞ norm can then be defined by

```
M1 = newlmi();
    lmiterm([M1 1 1 P],1,A,'s');        % M11 = (*) + I*P*A
    lmiterm([M1 1 2 P],1,B);            % M12 = I*P*B
    lmiterm([M1 1 3 0],C');             % M13 = C'
    lmiterm([M1 2 2 g],-1,1);           % M22 = -I*g*I
    lmiterm([M1 2 3 0],D');             % M23 = D'
    lmiterm([M1 3 3 g],-1,1);           % M33 = -I*g*I
M2 = newlmi();
    lmiterm([-M2 1 1 P],1,1);           % 0 < P
mylmi = getlmis();
```

where `getlmis()` gathers the 2 LMIs in one.

The last task before solving the minimisation problem is to express it in the form $c^T p$. This can be done by:

```
n = decnbr(mylmi);              % number of decision variables
c = zeros(n,1);                 % prepare the vector
for i = 1:n                     % loop over all decision variables
    gi = defcx(mylmi,i,g);      % g matrix when decision var. i is =1
```

```
                                  %          and all other dec. vars = 0
    c(i) = gi;                    % how c'p depends on gi
end
```

For example, if the objective function is $g + x_0^T P x_0$, the above following code can construct the corresponding `c` vector.

```
n = decnbr(mylmi);                % number of decision variables
c = zeros(n,1);                   % prepare the vector
for i = 1:n                       % loop over all decision variables
    [gi,Pi] = defcx(mylmi,i,g,P); % g and P when only dec. var i is = 1
    c(i) = gi + x0'*Pi*x0;        % how c'p depends on gi
end
```

Finally solve the optimisation problem

```
[cost, par] = mincx(mylmi,c);
```

The H_∞ norm is equal to `cost`. The decision variables `P` and `g` can be extracted as

```
P = dec2mat(mylmi,par,P);
g = dec2mat(mylmi,par,g);
```

Similarly the LMI conditions for computing the H_2 norm can be constructed.

Matlab files: `lmi_norm_hinf.m`, `lmi_norm_hinf.m`, `orc18_norms_via_LMIs.m`

Besides the LMI Lab of the Robust Control toolbox of MATLAB there are many other LMI solvers available. Worth mentioning here is the "Yet Another LMI Parser" (YALMIP) that provides an interface between the definition of the LMIs and many LMI solvers (including the LMI Lab). Two freeware solvers worth mentioning are the SEDUMI and SDPT3 toolboxes. A lengthier list of LMI solvers for MATLAB can be found in the documentation of YALMIP.

Solution to Problem 19.1 H_2 synthesis using LMIs

To synthesise a controller the LMI conditions (19.19), (19.20) and $\begin{bmatrix} Y & I \\ I & X \end{bmatrix} > 0$ are needed as constraints and trace(W) is minimised. Note that in order to impose $D_{zw} + D_{zu}\tilde{D}_K D_{vw} = 0$ the feed-through matrix D_K of the controller is set to zero. This accordingly simplifies the LMI conditions.

Matlab files: `orc19_H2_design_LMI.m`, `lmi_h2syn.m`

Solution to Problem 19.2 H_∞ **state-feedback design with pole-region constraints**

Because this is a state-feedback design and furthermore there are pole-region constraints imposed, the standard MATLAB function `hinfsyn` cannot be used in this problem. The synthesis is a direct implementation of LMIs (19.10), (19.5) and $P = P^T > 0$. Expanding inequality (19.5) in the case of conic sector and vertical strip pole-region constraints leads to

$$
\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2\alpha_l & 0 \\ 0 & 0 & 0 & -2\alpha_r \end{bmatrix} \otimes P + \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 & 0 \\ \sin(\beta) & \cos(\beta) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \otimes (AP + B_uY)
$$

$$
+ \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 & 0 \\ -\sin(\beta) & \cos(\beta) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \otimes (AP + B_uY)^T < 0.
$$

Using the short notations $s = \sin(\beta)$, $c = \cos(\beta)$ and $Q = AP + B_uY$ the above LMI can be rewritten as

$$
\begin{bmatrix} cQ + cQ^T & -sQ + sQ^T & 0 & 0 \\ sQ - sQ^T & cQ + cQ^T & 0 & 0 \\ 0 & 0 & 2\alpha_l P - Q - Q^T & 0 \\ 0 & 0 & 0 & -2\alpha_r P + Q + Q^T \end{bmatrix} < 0.
$$

Matlab files: `orc19_Hinf_design_LMI.m`, `lmi_hinfsyn_SFB.m`

Solution to Problem 21.1   ACC Benchmark Problem

The uncertainty model can be constructed in 3 different ways.

1. Using the physical model of the system. This is possible, only when it is known which are the varying coefficients and what is their range.

    Here only a simplified construction from the physical model is presented. A more detailed examples can be found, for example, in the documentation of the Matlab "μ-Analysis and Synthesis Toolbox" from Balas et.al., and in the book "Robust Control Design with MATLAB" from Gu et.al. (note that both books use older version of the Robust Control Toolbox than the one discussed in the exercises).

    In the provided model the only physical parameter that varies is k, and $k_{\min} \leq k \leq k_{\max}$. It is always possible to rewrite it as $k = k_0 + \tilde{k}\delta$, where $|\delta| \leq 1$, where $k_0 = (k_{\max} + k_{\min})/2$ and $\tilde{k} = (k_{\max} - k_{\min})/2$. Then as signal u multiplied by k can be equivalently represented in a linear-fractional transformation form, as shown in Fig. B.24.
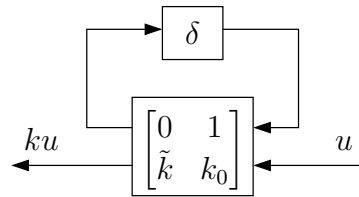

Figure B.24: Linear fractional transformation of an uncertain element.

Using this the system in (21.6) can be augmented with an additional input and output connected to δ, as shown below.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_0/m_1 & k_0/m_1 & 0 & 0 \\ k_0/m_2 & -k_0/m_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\tilde{k} & 1/m_1 & 1/m_1 & 0 \\ \tilde{k} & 0 & 0 & 1/m_2 \end{bmatrix} \begin{bmatrix} w \\ u \\ d_1 \\ d_2 \end{bmatrix}$$

$$z = x_1 - x_2$$
$$y = x_2$$

where $w = \delta z$ (perform the substitution to check the validity!).

The advantage of this approach is the clear connection between the physical parameters and δ. Using such models it becomes also clear, that whenever there are several independent parameters that vary, combining them to a single uncertainty matrix Δ, with $\|\Delta\| \leq 1$ is restrictive, since it includes more variations than the possible ones. This problem is addressed by the μ-synthesis approach (see the above references).

2. Using the uncertain state-space model functions in Matlab (see `help uss`). These functions allow defining uncertain gains, matrices and state-space models and using them in standard Matlab functions.

   For the above system, one can define: `k = ureal('k', 1, 'range', [0.5 2]);` and use it in the construction of the `A` matrix (which in return is an uncertain matrix) and the system model `G` (which turns out as uncertain state-space model).

   Then the function `lftdata` can be used to separate the uncertain model to a fixed part and an uncertain varying part: `[Go, Delta] = lftdata(G)`.

   Note, that the function `lftdata` appears to work correctly only when the uncertain parameters are defined with the middle value (that is k_0 in the case of k as nominal one). That is, `lftdata` will return the correct fixed part `Go`, only when `k` is defined as: `k = ureal('k', 1.25, 'range', [0.5 2])`

3. Using the uncertainty model construction technique presented on page 152. Note, that wherever the previous two approaches can be used only in the case, where it is known what the varying parameter is, this method can be also applied, when one has several models of the same system. For example, one can linearise a non-linear model (or identify a system from measurement data) for different operating conditions, and thus obtain several different models of the same system. Then, using this approach, an uncertainty model can be constructed that captures the variation in the system in the form of uncertainty.

   A vital issue when using these models is the selection of the nominal system (that is the nominal A matrix). For simplicity take only the spring constant (instead of the whole A matrix). It is known that $0.5 \leq k \leq 2$. Consider two different choices for the nominal value: $k_0 = 1$ and $k_0 = 1.25$. Since the uncertainty δ is normalised (recall that $\|\Delta\| \leq 1$ for the small gain theorem) each of the two choices of k_0 will lead to different scaling factors. When $k_0 = 1$ a scaling $\tilde{k} = 1$ will be needed, in order to include $k_{\max} = 2$, but this will also include, for example $k = 0$ which is equivalent to no spring at all - that is the position of the second cart cannot be influenced by moving the first one. When $k_0 = 1.25$, the tightest (for this problem) uncertainty description is achieved (no plants included outside the range of interest) with $\tilde{k} = 0.75$. This is illustrated in Fig. B.25.
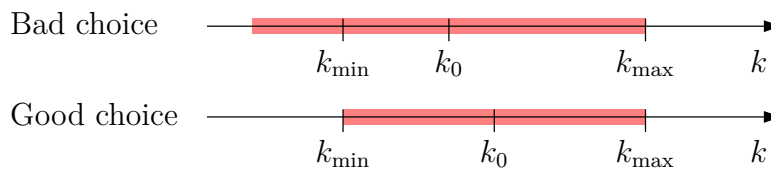

Figure B.25: Uncertainty description depending on the nominal model/value. The filled area represents the values of k captured by the uncertainty model.

The last two approaches are demonstrated in the Matlab solution to this problem.
(`orc21_ACC1_model.m`)

Having once constructed the model with the uncertainty there are different techniques
that one can use for the controller design.

- Nominal Performance [NP] - design a controller for the nominal system. This can be
  done with any of the previously presented techniques (LQR, H_2, mixed-sensitivity,
  etc.). However such a controller will not guarantee the stability of the two-cart
  system under variations in the spring constant.

- Robust Stability [RS] - design a controller that stabilises the closed-loop system
  under all plant variations captured by the uncertainty model. To do this one needs
  to construct a generalised plant as the one shown in Fig. B.26. Note, that besides
  having only the uncertainty input one can add also the disturbance inputs. This
  is demonstrated in `orc21_ACC2_hinfsyn_uss.m` and `orc21_ACC2_hinfsyn_unc.m`
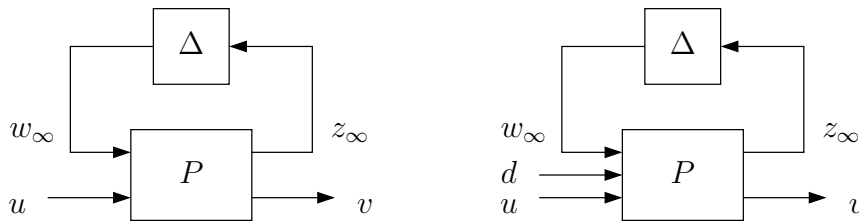 correspondingly for modelling techniques 2 (`uss`) and 3 (`unc`) above.

Figure B.26: Generalised plant for robust stability design.

- Robust Stability and Nominal Performance [RS+NP] - design a controller guaran-
  teeing the robust stability of the system under all plant variations and achieving
  some nominal performance. For example, using LMIs, one can combine the require-
  ment for robust stability under the model uncertainty (that is, the H_∞ norm from
  w_∞ to z_∞ being less than 1) with LQG, H_2, mixed-sensitivity H_∞ or/and pole-
  placement requirements for the nominal system. Note, that a controller satisfying
  this design requirements will guarantee the robust stability, but only the nominal
  performance. This is, however, in many cases sufficient, since the nominal perfor-
  mance often provides some "useful" performance under small changes in the plant.
  These designs are relatively straightforward, since they require combining the nec-
  essary LMI conditions. The pole region constraint approach and the mixed H_2/H_∞
  approach are presented in the Matlab file `orc21_ACC3_othersyn.m`.

- Robust Performance [RP] - design a controller that satisfies the specified perfor-
  mance criteria for all plants modelled by the uncertainty model. This is a much
  stronger requirement than the previous ones and more difficult to satisfy. To per-
  form the design a generalised plant $P(s)$ with both uncertainty channels (z_∞, w_∞)

and "mixed-sensitivity-type" performance channels (z_P, w_P) is constructed - see Fig. B.27. If a controller guarantying H_∞ norm smaller than 1 from w to z is obtained it will guarantee robust performance, i.e. no matter how the plant changes ($\|\Delta\| \leq 1$) the performance requirements will be satisfied. The solution is presented both with modelling techniques 2 (`uss`) and 3 (`unc`) of the above showed ones.
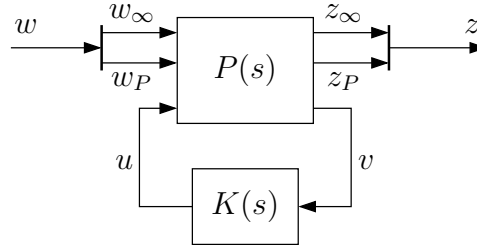

Figure B.27: Generalised plant with both stability and performance channels.

Looking at the frequency response of the controller it can be seen that the controller does not have a sufficient roll-off for high frequencies. The setup used so far weights the transfer functions SG_d and KSG_d. Both of them should decrease for high frequencies, so first order filters seem to be a reasonable choice for shaping the transfer functions. This choice enforces a 20 dB/dec decrease of the transfer functions. For high frequencies S is assumed to be constant. The plant's frequency response to disturbances G_d decreases with 40 dB/dec. Therefore, the requirement of a 20 dB/dec decrease of the transfer functions is already fulfilled without further roll-off in K. K can even increase with 20 dB/dec or at least be constant for high frequencies without violating the requirements given by the shaping filters.

In general, there are two possible improvements to deal with this problem. One possibility would be a higher order shaping filter to enforce sufficient roll-off in K. Above that, the four-block design allows additional weighting of the transfer functions S and KS and leads to an appropriate controller in frequency domain.

For setting up the four-block design a second input channel r acting on the plant's output has to be introduced. In a second step, the shaping filters have to be tuned adequately. As explained before, there should be a sufficient roll-off in the controller transfer function for high frequencies. Above that, the disturbance responses should meet the given requirements. Increasing the tuning parameter M_2 by a factor of four gives satisfying results. For further details please consult the Matlab file `orc21_ACC4_4block.m`.

Matlab files:

```
orc21_ACC1_model.m,        uncertainty_model.m
orc21_ACC2_hinfsyn_unc.m,  orc21_ACC2_hinfsyn_uss.m
orc21_ACC3_othersyn.m
orc21_ACC4_4block.m
```