Energy Systems: Hardware and Control
Lecture Notes for Control Part
at University of Freiburg

Michael Erhard, Gianluca Frison and Moritz Diehl

Preliminary Version of
November 20, 2017

Contents

Preface

These notes are based on the notes originally written by Michael Erhard and Moritz Diehl for the control part of the course ”Energy Systems: Hardware and Control” (part of REM master), academic year 2016-2017. The notes are expanded with new material covered in the control part of the course ”Energy Systems: Hardware and Control” (part of REM master), academic year 2017-2018. The aim of the control part of the course is to make its attendants familiar with concepts of state space control that include linear quadratic regulator (LQR), the Kalman filter and Model Predictive Control (MPC).

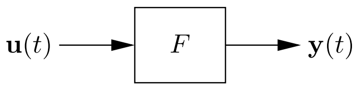
About the notation used in these lecture notes, a matrix $\mathbf{A}$ is capitalized and denoted using bold and roman style, a vector $\mathbf{x}$ is lower case and denoted using bold and roman style, a scalar x is lower case and denoted using italic style.

Sections marked with (*) are not covered in the course, but are left as reference for the curious reader.

Chapter 1

Background on Dynamic Systems in State Space

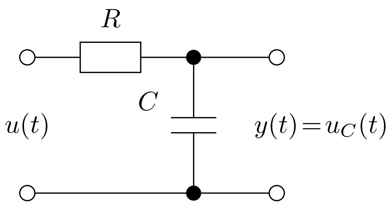
A dynamic system responds to an input signal $\mathbf{u}(t)$ with an output signal $\mathbf{y}(t)$ as depicted in the following block diagram

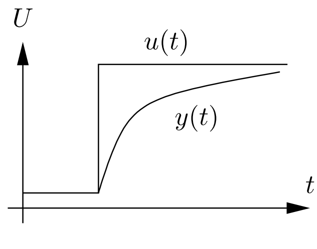


This behavior could be regarded as a 'mapping in time domain' of a function $\mathbf{u} : t \mapsto \mathbf{u}(t)$ to a function $\mathbf{y} : t \mapsto \mathbf{y}(t)$,

$$\mathbf{u} \mapsto \mathbf{y} = F\{\mathbf{u}\} \tag{1.1}$$

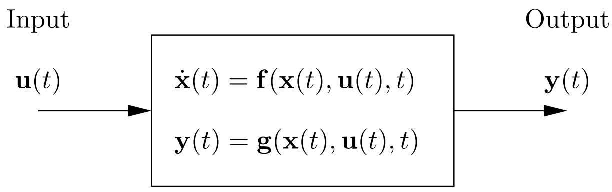
An example is a RC-lowpass circuit and its response to a step input signal





1.1 System dynamics given by Ordinary Differential Equations

If the system dynamics is given by ordinary differential equations (ODE), the system can be represented as follows



- $\mathbf{x}$ is the n-dimensional internal state of the system. It can be regarded as 'memory' of the system.

- The dynamics is given by the equations of motion in form of an ODE

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{1.2}$$

  called 'state equation' (or 'system equation'). It determines the time evolution of the state $\mathbf{x}(t)$ by an ordinary differential equation.

- The second equation

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{1.3}$$

  is called 'output equation' and maps the state (and input) to the output vector $\mathbf{y}(t)$. Note that the output, state and input vectors can have a different dimensions.

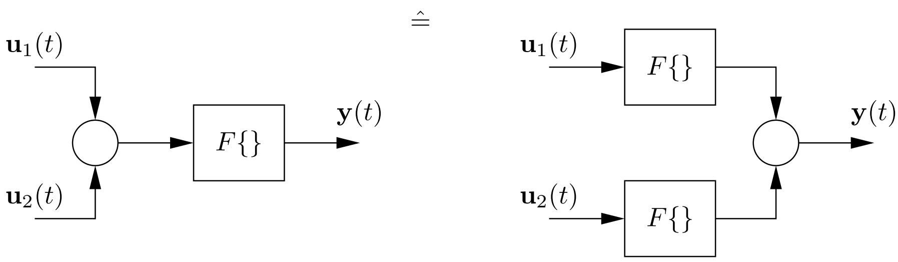
1.2 Linear Time-Invariant (LTI) System

A dynamical system F is called *linear* if the following conditions are fulfilled:

1. Superposition principle

$$F\{\mathbf{u}_1 + \mathbf{u}_2\} = F\{\mathbf{u}_1\} + F\{\mathbf{u}_2\} \tag{1.4}$$

   which can be illustrated as follows



2. Principle of amplification

$$F\{c\mathbf{u}\} = cF\{\mathbf{u}\} \tag{1.5}$$

   depicted as follows

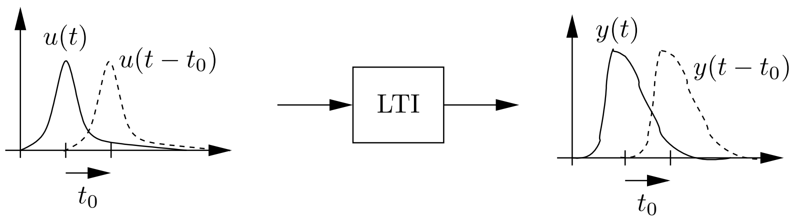
A dynamical system F is called *time-invariant*, if for any function $\mathbf{u}(t)$

$$\mathbf{y} \doteq F\{\mathbf{u}\} \tag{1.6}$$

the equation

$$\mathbf{y}_0 = F\{\mathbf{u}_0\} \tag{1.7}$$

is valid for all t_0, where the function definitions $\mathbf{u}_0 : t \mapsto \mathbf{u}_0(t) \doteq \mathbf{u}(t - t_0)$ and $\mathbf{y}_0 : t \mapsto \mathbf{y}_0(t) \doteq \mathbf{y}(t - t_0)$ are introduced. This can be illustrated by

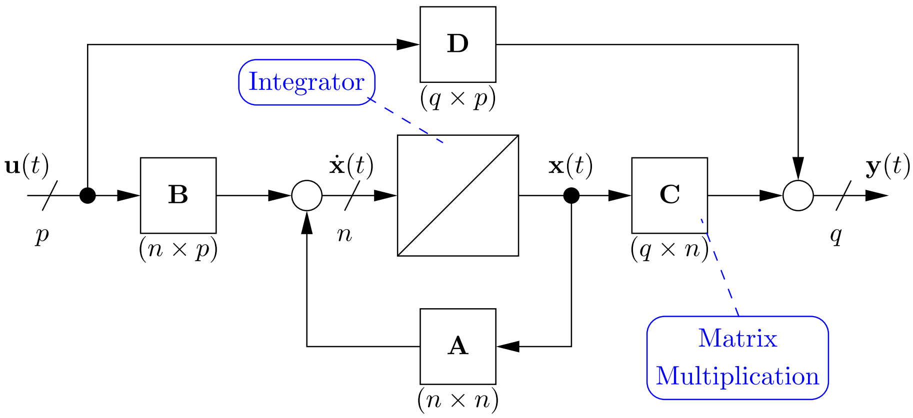


Note: For time invariance, the initial (internal) states of the system have to be $\mathbf{0}$ (zero state).

The general LTI system in state space can be written as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{1.8}$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{1.9}$$

This set of equations including dimensions of vectors and matrices can be drawn in the following block diagram



SISO and MIMO systems In a LTI system, the state vector $\mathbf{x} \in \mathbb{R}^n$ has dimension n, the input vector $\mathbf{u} \in \mathbb{R}^p$ has dimension p and the output vector $\mathbf{y} \in \mathbb{R}^q$ had dimension q. Therefore, the state space matrices have dimension: $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times p}$, $\mathbf{C} \in \mathbb{R}^{q \times n}$ and $\mathbf{D} \in \mathbb{R}^{q \times p}$.

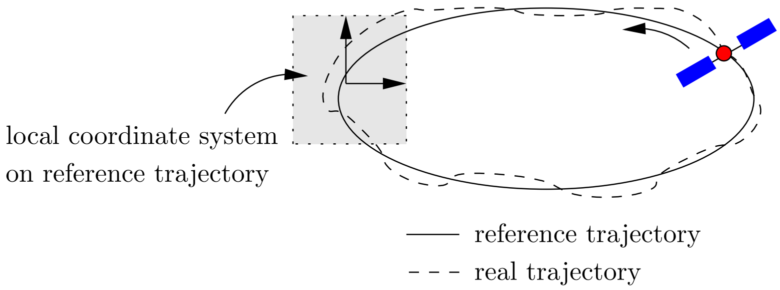
As a special case, we can consider a LTI system with only one input and one output, $p = 1$ and $q = 1$. This kind of system is called Single-Input Single-Output (SISO) and it is formulated as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) \tag{1.10}$$
$$y(t) = \mathbf{c}^\top \mathbf{x}(t) + du(t) \tag{1.11}$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{b} \in \mathbb{R}^n$, $\mathbf{c} \in \mathbb{R}^n$ and $d \in \mathbb{R}$.

The generic case where $p \geq 1$ and $q \geq 1$ is denoted as Multiple-Input Multiple-Output (MIMO).

Linearization The idea is to consider the behavior of a system around a reference or steady-state point by linearization of the ODE. As example we consider trajectory control of a satellite on an orbit



In absence of disturbances and with zero steering input, the satellite would fly on the orbit, denoted as solid trajectory. By introduction of a local (orthogonal) coordinate system, we only consider deviations from this reference trajectory. $\mathbf{x} = \mathbf{0}$ would then describe a satellite flying on the reference trajectory. As deviations are expected to be small compared to the overall trajectory, linearization of the spherical coordinate system is an adequate modelling approach.

In the subsequent sections of this course, only linear systems will be considered. Although almost all real world problems lead to nonlinear ODE, linearization is a powerful tool, which can be applied in many cases. The following procedure is applied

1. Set up general ODE.
2. Linearize system around equilibrium point.
3. Design controller.
4. Validate control design with general (nonlinear) ODE in numerical simulations.

1.3 Setup of State Space Equations

In this section, we consider a SISO system. The dynamics is assumed to be given by a linear differential equation

$$y^{\langle n\rangle} + a_{n-1}y^{\langle n-1\rangle} + \cdots + a_1\dot{y} + a_0 y = b_{n-1}u^{\langle n-1\rangle} + \cdots + b_1\dot{u} + b_0 u \tag{1.12}$$

The superscript $^{\langle n\rangle}$ denotes the n^{th} time derivative, the $a_i, b_i \in \mathbb{R}$ are constant real coefficients. For sake of simplicity, we dropped the time dependencies of u and y. We also assumed $b_n = 0$ (i.e., $D = 0$ in state space form) for simplicity.

Control Canonical Form In the following, this system shall be described as LTI system in state space. The derivation is done in two steps.

Step 1 Solve for the $u(t)$ term on the right hand side (RHS) of the ODE, i.e. consider

$$y^{\langle n\rangle} + a_{n-1}y^{\langle n-1\rangle} + \cdots + a_1\dot{y} + a_0 y = u \tag{1.13}$$

This system of n^{th} order is transformed into a 1^{st} order system by introduction of the state $\mathbf{x} = [x_1, \ldots, x_n]^\top$ and the definitions

$$\begin{aligned}
x_1 &\doteq y && (1.14)\\
x_2 &\doteq \dot{y} = \dot{x}_1 && (1.15)\\
x_3 &\doteq \ddot{y} = \dot{x}_2 && (1.16)\\
&\vdots && (1.17)\\
x_n &\doteq y^{\langle n-1\rangle} = \dot{x}_{n-1} && (1.18)
\end{aligned}$$

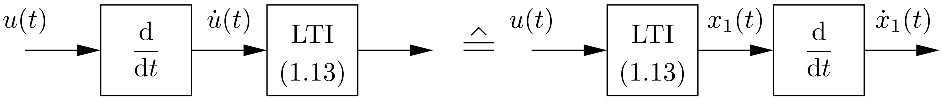
The ODE (1.13) can then be written as

$$\begin{aligned}
\dot{x}_n &= \frac{\mathrm{d}}{\mathrm{d}t}y^{\langle n-1\rangle} = y^{\langle n\rangle} = -a_{n-1}y^{\langle n-1\rangle} - \cdots - a_1\dot{y} - a_0 y + u\\
&= -a_{n-1}x_n - \cdots - a_1 x_2 - a_0 x_1 + u && (1.19)
\end{aligned}$$

or in matrix representation

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & & & & \\ & 0 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & \\ & & & & 0 & 1 \\ -a_0 & -a_1 & \cdots & \cdots & \cdots & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \tag{1.20}$$

Step 2 : As the system is linear, we can solve (1.13) for $\dot{u}(t)$, $\ddot{u}(t), \dots$ on the RHS separately and then add the results to obtain the solution for the complete system. For the solution of (1.13) for $\dot{u}(t)$ on the RHS, the possibility of swapping LTI systems is exploited as follows



Hence, instead of solving for $\dot{u}(t)$, we solve for $u(t)$ and take the solution $\dot{x}_1(t) = x_2(t)$ instead of $x_1(t)$. Applying this principle to higher orders and utilizing (1.14–1.18) yields

$$y(t) = [b_0, b_1, \dots, b_{n-1}]\,\mathbf{x}(t) \tag{1.21}$$

The result can be summarized as

Control Canonical Form

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 & & & & \\ & 0 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & \\ & & & & 0 & 1 \\ -a_0 & -a_1 & \cdots & \cdots & \cdots & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \tag{1.22}$$

$$y(t) = [b_0, b_1, \dots, b_{n-1}]\,\mathbf{x}(t) \tag{1.23}$$

Similar considerations lead to the following alternative form, which shall be given without derivation

Observer Canonical Form

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 & -a_0 \\ 1 & \ddots & & & & -a_1 \\ & 1 & \ddots & & & \vdots \\ & & \ddots & \ddots & & \vdots \\ & & & \ddots & 0 & \vdots \\ & & & & 1 & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ \vdots \\ \vdots \\ b_{n-1} \end{bmatrix} u(t) \tag{1.24}$$

$$y(t) = [0, \dots, 0, 1]\,\mathbf{x}(t) \tag{1.25}$$

It should be remarked that the state space representation for a given ODE (1.12) is not unique. A transformation will be discussed later in Sect. 1.5.

1.4 Solution of the State Space ODE

In the following, the equation

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{1.26}$$

with $\mathbf{x}(t_0) = \mathbf{x}_0$ as initial condition will be solved.

Homogeneous solution

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}_0 \tag{1.27}$$

is the solution for

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) \tag{1.28}$$

which is the homogeneous part of (1.26). We used the *matrix exponential function*, which is defined by

$$e^{\mathbf{A}(t-t_0)} \doteq \sum_{\nu=0}^{\infty} \frac{\mathbf{A}^\nu (t-t_0)^\nu}{\nu!} \tag{1.29}$$

The time derivative reads

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} e^{\mathbf{A}(t-t_0)} &= \frac{\mathrm{d}}{\mathrm{d}t} \sum_{\nu=0}^{\infty} \frac{\mathbf{A}^\nu (t-t_0)^\nu}{\nu!} = \sum_{\nu=1}^{\infty} \frac{\mathbf{A}^\nu \nu (t-t_0)^{\nu-1}}{\nu!} \\ &= \mathbf{A} \sum_{\nu=1}^{\infty} \frac{\mathbf{A}^{\nu-1} (t-t_0)^{\nu-1}}{(\nu-1)!} = \mathbf{A} e^{\mathbf{A}(t-t_0)} \end{aligned} \tag{1.30}$$

Computing the time derivative of the solution (1.27) yields

$$\dot{\mathbf{x}}(t) = \mathbf{A} \underbrace{e^{\mathbf{A}(t-t_0)}\mathbf{x}_0}_{\mathbf{x}(t)} = \mathbf{A}\mathbf{x}(t) \tag{1.31}$$

and proves that the solution fulfills (1.28).

General Solution The general solution reads

$$\mathbf{x}(t) = \mathbf{\Phi}(t, t_0)\mathbf{x}_0 + \int_{t_0}^{t} \mathbf{\Phi}(t, \tau)\mathbf{B}\mathbf{u}(\tau)\mathrm{d}\tau \tag{1.32}$$

with

$$\mathbf{\Phi}(t, t_0) \doteq e^{\mathbf{A}(t-t_0)} \tag{1.33}$$

Note that the first term is the homogeneous solution due to the initial condition $\mathbf{x}_0$ and the second term is a convolution integral of input $\mathbf{u}(t)$. In order to show that (1.32) is a solution, we compute $\dot{\mathbf{x}}(t)$ by deriving (1.32) using the Leibniz integral rule

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{\Phi}(t, t_0)\mathbf{x}_0 + \underbrace{\mathbf{\Phi}(t, t)}_{=\mathbf{I}}\mathbf{B}\mathbf{u}(t) + \int_{t_0}^{t} \underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{\Phi}(t, \tau)}_{\mathbf{A}\mathbf{\Phi}(t,\tau)}\mathbf{B}\mathbf{u}(\tau)\,\mathrm{d}\tau \\ &= \underbrace{\mathbf{A}\left(\mathbf{\Phi}(t, t_0)\mathbf{x}_0 + \int_{t_0}^{t} \mathbf{\Phi}(t, \tau)\mathbf{B}\mathbf{u}(\tau)\,\mathrm{d}\tau\right)}_{=\mathbf{x}(t),\ \text{compare (1.32)}} + \mathbf{B}\mathbf{u}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \end{aligned} \tag{1.34}$$

□

1.5 Diagonalization and Modal Canonical Form

Repetition Eigenvalues and Eigenvectors

- Eigenvalue λ and eigenvector $\mathbf{v} \neq \mathbf{0}$ are defined by the following equation:

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \tag{1.35}$$

- The *characteristic polynomial* of $\mathbf{A}$ is defined by[a]

$$p(\lambda) \doteq \det(\lambda\mathbf{I} - \mathbf{A}) \tag{1.36}$$

- Eigenvalues λ_i are the *roots* of the characteristic polynomial, i.e., solution of the *characteristic equation* $p(\lambda) = 0$.

[a] An alternative definition would be $p(\lambda) \doteq \det(\mathbf{A} - \lambda\mathbf{I})$. We use the definition above to obtain $p(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \cdots$.

In the following, we assume for simplicity that the eigenvalues of $\mathbf{A}$ are different, i.e. $\lambda_i \neq \lambda_j$ for $i \neq j$. As a consequence, there exists a matrix $\mathbf{V}$ such that

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} \qquad \leftrightarrow \qquad \mathbf{\Lambda} = \mathbf{V}^{-1}\mathbf{A}\mathbf{V} \tag{1.37}$$

with

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \tag{1.38}$$

The matrix $\mathbf{V}$ is composed of the (right) eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$ by

$$\mathbf{V} \doteq [\mathbf{v}_1, \ldots, \mathbf{v}_n] \tag{1.39}$$

The left eigenvectors $\mathbf{w}_1^\top, \ldots, \mathbf{w}_n^\top$ are the rows of the inverse matrix as follows

$$\begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_n^\top \end{bmatrix} \doteq \mathbf{V}^{-1} \tag{1.40}$$

Considering $\mathbf{V}^{-1}\mathbf{V} = \mathbf{I}$ element-wise yields the relation between the left and right eigenvectors

$$\mathbf{w}_i^\top \mathbf{v}_j = \delta_{i,j} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{otherwise} \end{cases} \tag{1.41}$$

The matrix $\mathbf{A}$ can then be written

$$\mathbf{A} = [\mathbf{v}_1, \ldots, \mathbf{v}_n] \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_n^\top \end{bmatrix} \tag{1.42}$$

Now, the matrix exponential reads (assume $t_0 = 0$)

$$\begin{aligned} e^{\mathbf{A}t} &= e^{\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}t} = \sum_{\nu=0}^{\infty} \frac{1}{\nu!}(\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1})^\nu t^\nu \\ &\vdots \quad \text{with } (\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1})^\nu = \mathbf{V}\mathbf{\Lambda}\underbrace{\mathbf{V}^{-1}\mathbf{V}}_{=\mathbf{I}}\mathbf{\Lambda}\mathbf{V}^{-1} \cdots \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} = \mathbf{V}\mathbf{\Lambda}^\nu\mathbf{V}^{-1} \\ &= \mathbf{V}\left(\sum_{\nu=0}^{\infty} \frac{1}{\nu!}\mathbf{\Lambda}^\nu t^\nu\right)\mathbf{V}^{-1} = \mathbf{V}e^{\mathbf{\Lambda}t}\mathbf{V}^{-1} \end{aligned} \tag{1.43}$$

with

$$e^{\mathbf{\Lambda}t} = \sum_{\nu=0}^{\infty} \frac{\mathbf{\Lambda}^{\nu}}{\nu!} t^{\nu} = \begin{bmatrix} e^{\lambda_1 t} & & \\ & \ddots & \\ & & e^{\lambda_n t} \end{bmatrix} \tag{1.44}$$

For $\mathbf{\Phi}(\cdot)$ follows

$$\mathbf{\Phi}(t) = e^{\mathbf{A}t} = \sum_{i=1}^{n} e^{\lambda_i t} \mathbf{v}_i \mathbf{w}_i^{\top} \tag{1.45}$$

The expressions $e^{\lambda_i t}\mathbf{v}_i$ can be regarded as dynamic modes of the system. The homogeneous solution now reads

$$\mathbf{x}(t) = \mathbf{\Phi}(t)\mathbf{x}_0 = \sum_{i=1}^{n} e^{\lambda_i t} \mathbf{v}_i (\mathbf{w}_i^{\top} \mathbf{x}_0) \tag{1.46}$$

where the term $(\mathbf{w}_i^{\top}\mathbf{x}_0)$ could be interpreted as 'excitation' amplitude of mode i due to the initial condition given by $\mathbf{x}_0$.

Regarding the state space system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{1.47}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{1.48}$$

and by definition of the new state variables $\mathbf{z}(t)$ by

$$\mathbf{z}(t) \doteq \mathbf{V}^{-1}\mathbf{x}(t) \tag{1.49}$$

we get for (1.47) by multiplication with $\mathbf{V}^{-1}$ from the left

$$\mathbf{V}^{-1}\dot{\mathbf{x}}(t) = \mathbf{V}^{-1}\mathbf{A}\underbrace{\mathbf{V}\mathbf{V}^{-1}}_{=\mathbf{I}}\mathbf{x}(t) + \mathbf{V}^{-1}\mathbf{B}\mathbf{u}(t) = \mathbf{\Lambda}\mathbf{V}^{-1}\mathbf{x}(t) + \mathbf{V}^{-1}\mathbf{B}\mathbf{u}(t) \tag{1.50}$$

As a result the ODE reads

$$\dot{\mathbf{z}}(t) = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \mathbf{z}(t) + \mathbf{V}^{-1}\mathbf{B}\mathbf{u}(t) \tag{1.51}$$

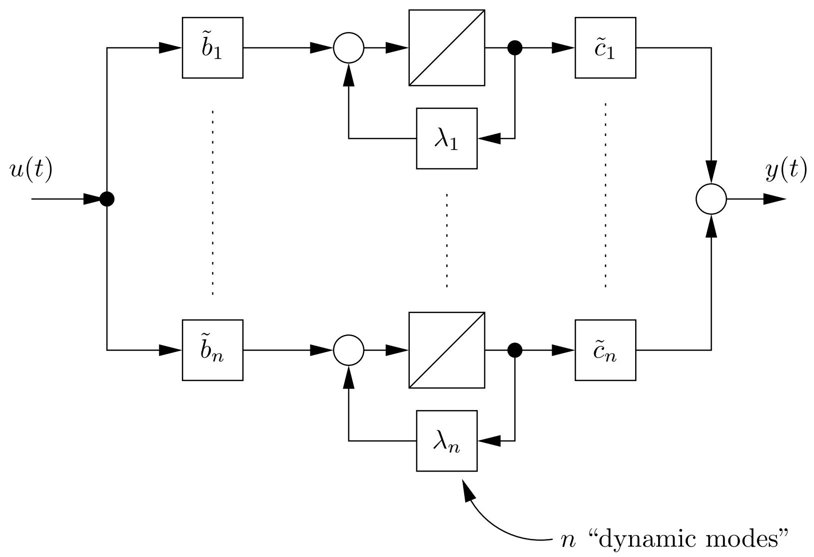
$$\mathbf{y}(t) = \mathbf{C}\mathbf{V}\mathbf{z}(t) + \mathbf{D}\mathbf{u}(t) \tag{1.52}$$

This representation is denoted *modal canonical form.*

For a SISO system we can define the vectors

$$\begin{bmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_n \end{bmatrix} \doteq \mathbf{V}^{-1}\mathbf{b} \qquad [\tilde{c}_1, \ldots, \tilde{c}_n] \doteq \mathbf{c}^{\top}\mathbf{V} \tag{1.53}$$

and draw the following block diagram



1.6 Dynamics and Stability

For consideration of stability, we examine the time evolution of the modes in equation (1.46)

$$\mathbf{\Phi}(t) = \sum_{i=1}^{n} \underbrace{e^{\lambda_i t}\mathbf{v}_i}_{\text{mode}} \mathbf{w}_i^\top \tag{1.54}$$

As simple example, we consider a system with $n = 2$ states in the form

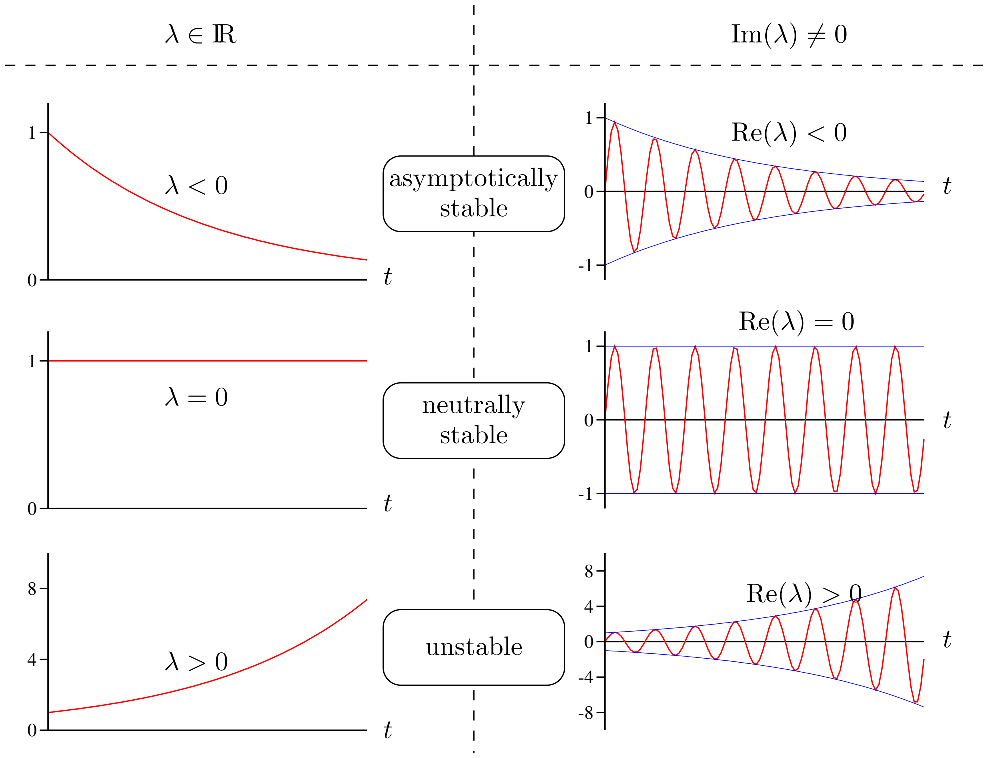
$$\mathbf{\Phi}(t) = e^{\lambda_1 t}\mathbf{v}_1\mathbf{w}_1^\top + e^{\lambda_2 t}\mathbf{v}_2\mathbf{w}_2^\top \tag{1.55}$$

For an initial value problem ($u(t) = 0$) with $\mathbf{x}_0 = c\mathbf{v}_2$ (c is a constant), the solution reads

$$\mathbf{x}(t) = \Phi(t)\mathbf{x}_0 = e^{\lambda_1 t}\mathbf{v}_1 \underbrace{\mathbf{w}_1^\top c\mathbf{v}_2}_{0} + e^{\lambda_2 t}\mathbf{v}_2 \underbrace{\mathbf{w}_2^\top c\mathbf{v}_2}_{c} = ce^{\lambda_2 t}\mathbf{v}_2 \tag{1.56}$$

This corresponds to an excitation of mode λ_2.

The λ_i in the exponential determines the time evolution of mode i as depicted in the following figures

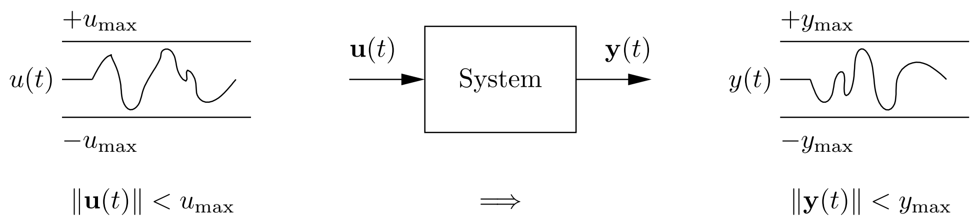


There are two comments on eigenvalues with imaginary part $\mathrm{Im}(\lambda) \neq 0$

- If the coefficients of the characteristic polynomial are real values, which is usually the case for physical systems, and an eigenvalue λ has an imaginary part $\mathrm{Im}(\lambda) \neq 0$, the complex conjugated value $\lambda^\star$ is also an eigenvalue. In other words: complex eigenvalues occur as pairs. The same holds for the eigenvectors.
- The oscillating parts in the time evolution of the state variables are always composed of the two modes λ and $\lambda^\star$ resulting in real values, e.g.

$$e^{\lambda t} + e^{\lambda^\star t} = 2e^{\mathrm{Re}(\lambda)t}\cos(\mathrm{Im}(\lambda)t) \tag{1.57}$$

BIBO Stability A system is said to have bounded input bounded output (BIBO) stability if every bounded input results in a bounded output.



Asymptotic stability $\Rightarrow$ BIBO Stability

An LTI system is BIBO stable (and internally stable) if $\mathrm{Re}(\lambda_i) < 0$ for all eigenvalues λ_i.

Proof (*) (sketch): assume $\mathbf{x}_0 = 0$ (LTI system, $D = 0$)

$$y(t) = C\int_0^t \mathbf{\Phi}(t,\tau)\mathbf{u}(t)\mathrm{d}\tau \tag{1.58}$$

Hence, the following holds

$$\|y(t)\| \leq u_{\max} \left\| C \int_0^t \mathbf{\Phi}(t,\tau)\mathrm{d}\tau \right\| \tag{1.59}$$

It remains to show that $\|\cdot\|$ on the RHS is bounded. This is done by considering that $\mathbf{\Phi}(\cdot)$ involves terms

$$\int_0^t \tau^l e^{\lambda\tau}\mathrm{d}\tau = \left[\frac{\tau^l}{\lambda}e^{\lambda\tau}\right]_0^t - \frac{l}{\lambda}\int_0^t \tau^{l-1}e^{\lambda\tau}\mathrm{d}\tau \tag{1.60}$$

The $[\cdot]$ term is bounded if $\mathrm{Re}(\lambda) < 0$. The integral on the RHS is considered by induction $l \to (l-1)$ until $l = 0$.

□

Note that for $\mathrm{Re}(\lambda) = 0$, the system is neutrally or marginally stable, but *not* BIBO stable as a bounded input function leading to an unbounded output can be found.

Methods to examine stability

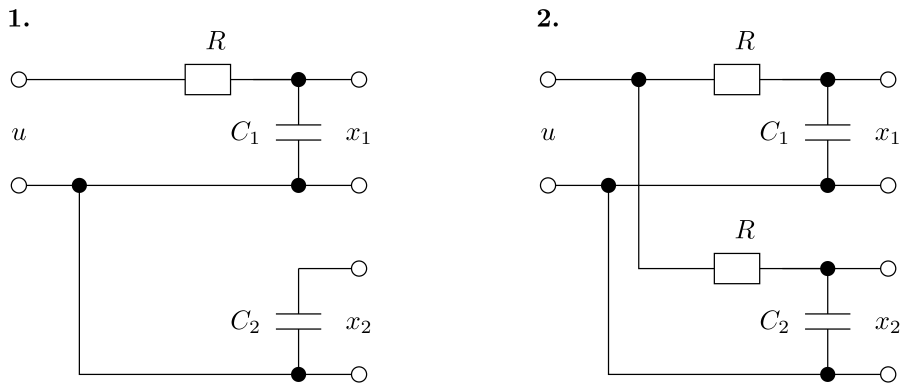
- Compute eigenvalues explicitly and check whether $\mathrm{Re}(\lambda_i) < 0$. Nowadays this is easy to do on a computer.
- Utilize algebraic criteria on the characteristic polynomial, e.g. Hurwitz or Routh. This is particularly useful if no computer is available.

Chapter 2

Controllability

2.1 Controllability

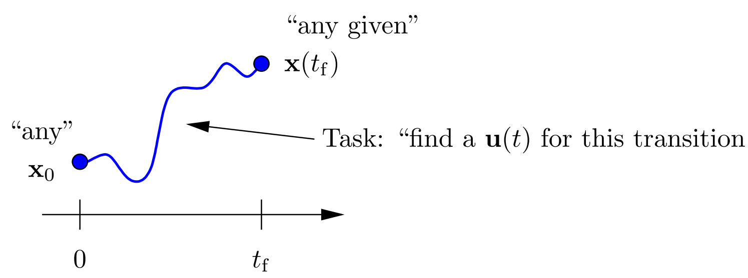
We consider controllability, i.e., control of the state $\mathbf{x} = [x_1, x_2]^\top$ by input u for the following introductory examples



1. Is not controllable as x_2 is 'disconnected'.
2. Here we have to distinguish two cases. For $C_1 = C_2$ the subsystems would behave equally, hence the states can not be manipulated separately. System not controllable. For $C_1 \neq C_2$ any state can be generated by appropriate choice of $u(t)$, hence system controllable.

> **Controllability**
>
> A system is controllable, if in finite time t_f any initial state $\mathbf{x}(0)$ can be driven to any given final state $\mathbf{x}(t_\mathrm{f})$ by appropriate choice of the control signal $\mathbf{u}(t)$ for $0 \leq t \leq t_\mathrm{f}$.

This can be depicted as follows



By consideration of the solution of the state space ODE

$$\mathbf{x}(t_\mathrm{f}) = e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}(0) + \int_0^{t_\mathrm{f}} e^{\mathbf{A}(t_\mathrm{f}-\tau)}\mathbf{B}\mathbf{u}(\tau)\,\mathrm{d}\tau \tag{2.1}$$

we get

$$\underbrace{\mathbf{x}(t_\mathrm{f}) - e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}(0)}_{\doteq -e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}_\mathrm{i}} = \int_0^{t_\mathrm{f}} e^{\mathbf{A}(t_\mathrm{f}-\tau)}\mathbf{B}\mathbf{u}(\tau)\,\mathrm{d}\tau \tag{2.2}$$

The value $\mathbf{x}_\mathrm{i}$ is defined by setting the LHS equal to $-e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}_\mathrm{i}$. As the equation has to be valid for any $\mathbf{x}(t_\mathrm{f})$ and any $\mathbf{x}(0)$, the following equation has to hold for all $\mathbf{x}_\mathrm{i} \in \mathbb{R}^n$.

$$-e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}_\mathrm{i} = \int_0^{t_\mathrm{f}} e^{\mathbf{A}(t_\mathrm{f}-\tau)}\mathbf{B}\mathbf{u}(\tau)\,\mathrm{d}\tau \tag{2.3}$$

The system is controllable, if for any $\mathbf{x}_\mathrm{i} \in \mathbb{R}^n$, a <u>finite</u> t_f and a control input $\mathbf{u}(t)$ for $0 \le t \le t_\mathrm{f}$ can be found, such that (2.3) holds. In other words: by appropriate choice of $\mathbf{u}(t)$, the system can be driven from any initial state $\mathbf{x}_\mathrm{i}$ to the zero state in finite time t_f.

Controllability for SISO Systems

Criterion by Kalman (1960). Define controllability matrix

$$\mathcal{C} \doteq \left[\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \ldots, \mathbf{A}^{n-1}\mathbf{b}\right] \tag{2.4}$$

The system $(\mathbf{A}, \mathbf{b})$ is controllable, if $\mathcal{C}$ has full rank n. i.e. $\det(\mathcal{C}) \neq 0$.

Proof: consider

$$-e^{\mathbf{A}t_\mathrm{f}}\mathbf{x}_\mathrm{i} = \int_0^{t_\mathrm{f}} e^{\mathbf{A}(t_\mathrm{f}-\tau)}\mathbf{b}u(\tau)\,\mathrm{d}\tau \tag{2.5}$$

Hence

$$\begin{aligned} -\mathbf{x}_\mathrm{i} &= \int_0^{t_\mathrm{f}} e^{-\mathbf{A}\tau}\mathbf{b}u(\tau)\,\mathrm{d}\tau = \int_0^{t_\mathrm{f}} \left(\sum_{\nu=0}^{\infty} \frac{(-\mathbf{A})^\nu \tau^\nu}{\nu!}\right)\mathbf{b}u(\tau)\,\mathrm{d}\tau \\ &= \sum_{\nu=0}^{\infty} \mathbf{A}^\nu \mathbf{b} \underbrace{\int_0^{t_\mathrm{f}} \frac{(-1)^\nu \tau^\nu}{\nu!} u(\tau)\,\mathrm{d}\tau}_{u_\nu \doteq} \end{aligned} \tag{2.6}$$

Thus we get for $\mathbf{x}_\mathrm{i}$

$$\mathbf{x}_\mathrm{i} = -\sum_{\nu=0}^{\infty} \mathbf{A}^\nu \mathbf{b} u_\nu \tag{2.7}$$

This equation has a solution for any $\mathbf{x}_\mathrm{i}$, if $\mathbf{A}^\nu\mathbf{b}$ span up the complete vector space, such that any $\mathbf{x}_\mathrm{i}$ can be composed by appropriate choice of the u_ν coefficients.

It remains to show that $\mathbf{A}^\nu\mathbf{b}$ with $\nu = 0, \ldots, \infty$ span up the complete vector space if $\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \ldots, \mathbf{A}^{n-1}\mathbf{b}$ are linearly independent, i.e. $\mathcal{C}$ is non-singular. The argument is based on the theorem of Cayley-Hamilton for the characteristic polynomial $p(\mathbf{A}) = 0$, which states that $\mathbf{A}^n$ can be written as linear combination (LC) of $\mathbf{A}^0, \ldots, \mathbf{A}^{n-1}$. Hence, $\mathbf{A}^{n+1} = \mathbf{A}\mathbf{A}^n$ can be written as LC of $\mathbf{A}^0, \ldots, \mathbf{A}^n$ and recursively as LC of $\mathbf{A}^0, \ldots, \mathbf{A}^{n-1}$. As a consequence, it is sufficient to consider $\mathbf{A}^0, \ldots, \mathbf{A}^{n-1}$.

□

Example We consider the introductory example on page 15. The system ODE read

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -\frac{1}{RC_1} & 0 \\ 0 & -\frac{1}{RC_2} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} \frac{1}{RC_1} \\ \frac{1}{RC_2} \end{bmatrix} u(t) \tag{2.8}$$

The controllability matrix is then

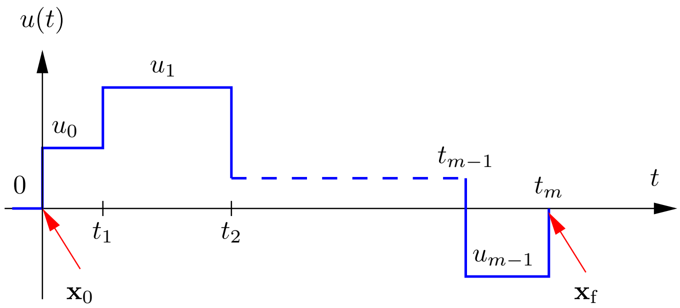
$$\mathcal{C} = [\mathbf{b}, \mathbf{Ab}] = \begin{bmatrix} \frac{1}{RC_1} & -\frac{1}{(RC_1})^2 \\ \frac{1}{RC_2} & -\frac{1}{(RC_2})^2 \end{bmatrix} \tag{2.9}$$

Hence the system is controllable if

$$\det(\mathcal{C}) = \frac{1}{(RC_1)(RC_2)} \left(-\frac{1}{RC_2} + \frac{1}{RC_1} \right) \neq 0 \tag{2.10}$$

which is equivalent to $C_1 \neq C_2$.

Control Input for State Transition (*) The task is to control the state transition from $\mathbf{x}_0 \to \mathbf{x}_\mathrm{f}$ with a piece-wise constant control input given as follows



Using the solution of the ODE (2.1), we get

$$\mathbf{x}_\mathrm{f} = e^{\mathbf{A}t_m}\mathbf{x}_0 + \sum_{i=0}^{m-1} \left(\int_{t_i}^{t_{i+1}} e^{\mathbf{A}(t_m-\tau)}\mathbf{b}\,\mathrm{d}\tau \right) u_i \tag{2.11}$$

By defining

$$\mathbf{p}_i \doteq \int_{t_i}^{t_{i+1}} e^{\mathbf{A}(t_m-\tau)}\mathbf{b}\,\mathrm{d}\tau \tag{2.12}$$

(2.11) can be written as

$$[\mathbf{p}_0, \ldots, \mathbf{p}_{m-1}] \begin{bmatrix} u_0 \\ \vdots \\ u_{m-1} \end{bmatrix} = \mathbf{x}_\mathrm{f} - e^{\mathbf{A}t_m}\mathbf{x}_0 \tag{2.13}$$

Hence the input amplitudes can be computed by

$$\begin{bmatrix} u_0 \\ \vdots \\ u_{m-1} \end{bmatrix} = [\mathbf{p}_0, \ldots, \mathbf{p}_{m-1}]^{-1} \left(\mathbf{x}_\mathrm{f} - e^{\mathbf{A}t_m}\mathbf{x}_0 \right) \tag{2.14}$$

It should be remarked that due to the dimensions, (at least) n control steps are needed for an n-dimensional state vector. In addition, the times t_i have to be chosen such that the $\mathbf{p}_i$ are linearly independent.

2.2 Extension to MIMO Systems

Having introduced controllability for SISO systems, we now sketch the criteria for MIMO systems.

Controllability for MIMO systems

The controllability matrix can now be defined as

$$\mathcal{C} \doteq \left[\mathbf{B}, \mathbf{A}\mathbf{B}, \mathbf{A}^2\mathbf{B}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right] \tag{2.15}$$

The system $(\mathbf{A}, \mathbf{B})$ is controllable if $\mathrm{rank}(\mathcal{C}) = n$. (Note that $\mathcal{C}$ is a matrix of size $n \times (np)$.)

Proof (sketch): repeat basically the same as above by replacing $\mathbf{b}$ by $\mathbf{B}$ and $u(t)$ by $\mathbf{u}(t)$.

$$\hookrightarrow \cdots \hookrightarrow \mathbf{x}_0 = -\sum_{\nu=0}^{\infty} \mathbf{A}^{\nu}\mathbf{b}\mathbf{u}_{\nu} \tag{2.16}$$

Hence the columns of $\left[\mathbf{B}, \mathbf{A}\mathbf{B}, \mathbf{A}^2\mathbf{B}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right]$ have to span up the vector space $\mathbb{R}^n$. This condition is equal to $\mathrm{rank}(\mathcal{C}) = n$. Note that we can stop the sum at n due to the Cayley-Hamilton theorem.

$\square$

Repetition: Rank of a Matrix

$$\begin{aligned}\mathrm{rank}(\mathbf{M}) &= \text{number of linearly independent column vectors in } \mathbf{M}\\ &\qquad\text{(or alternatively)}\\ &= \text{number of linearly independent row vectors in } \mathbf{M}.\end{aligned}$$

2.3 Gilbert Criterion and Kalman Decomposition

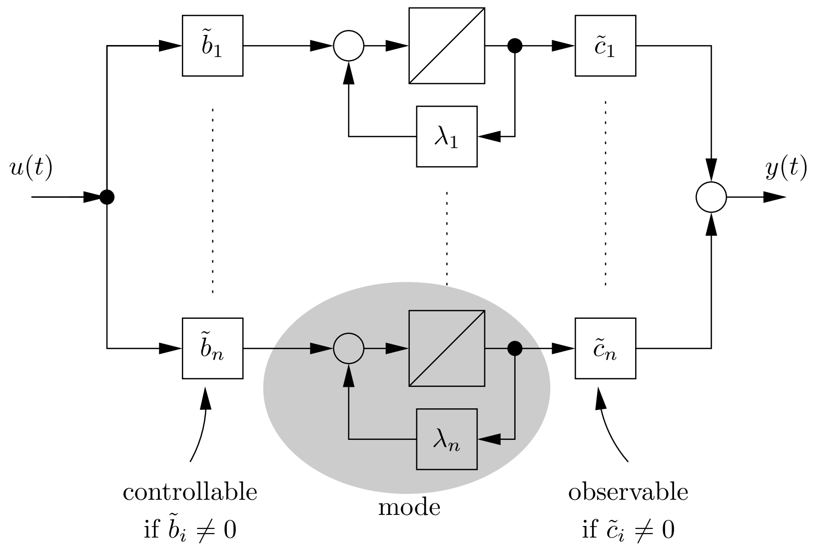
The modal canonical form was introduced by the transformation (1.49)

$$\mathbf{z}(t) \doteq \mathbf{V}^{-1}\mathbf{x}(t) \tag{2.17}$$

which could be regarded as transformation of the system $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}) \to (\mathbf{\Lambda}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}}, \mathbf{D})$ with the matrices

$$\tilde{\mathbf{B}} = \mathbf{V}^{-1}\mathbf{B} \qquad \tilde{\mathbf{C}} = \mathbf{C}\mathbf{V} \qquad \mathbf{\Lambda} = \mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \tag{2.18}$$

Note that $\lambda_i \neq \lambda_j$ for $i \neq j$ to avoid a more theoretical discussion. For a SISO system in modal canonical form, controllability (and observability, a notion that will be introduced later in the course) can be understood by the following descriptive block diagram



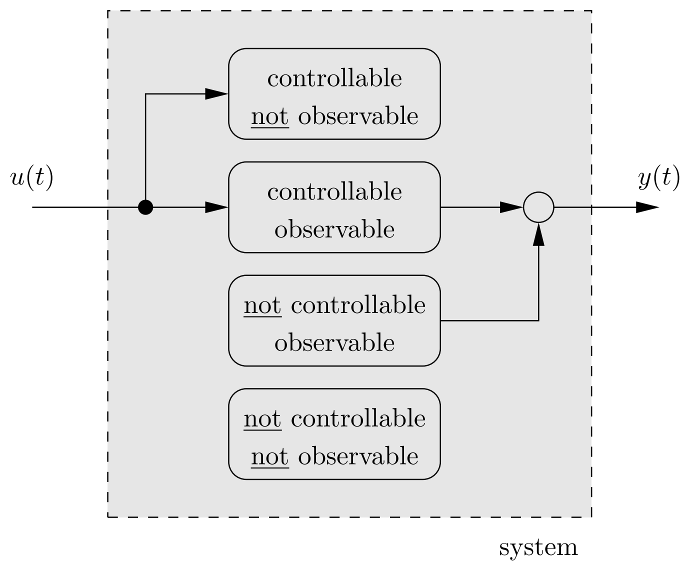
Gilbert Criterion (for SISO system)

The system $(\mathbf{\Lambda}, \tilde{\mathbf{b}}, \tilde{\mathbf{c}}^\top)$ (with $\lambda_i \neq \lambda_j$ for $i \neq j$) is

- controllable if all elements $\tilde{b}_i$ of $\tilde{\mathbf{b}}$ are non-zero.
- observable if all elements $\tilde{c}_i$ of $\tilde{\mathbf{c}}^\top$ are non-zero.

Proof (without).

Finally, it should be remarked that each mode can be attributed the two properties controllability and observability separately. Hence, the modes can be split up into four classes called Kalman decomposition, depicted in the following figure



In these notes, we consider only the part of the system that is both controllable and observable.

2.4 Stabilizability

Stabilizability is a weaker notion than controllability.

Stabilizability

The system $(\mathbf{A}, \mathbf{B})$ is stabilizable if there exist a matrix $\mathbf{K} \in \mathbb{R}^{p\times n}$ such that the matrix $\mathbf{A} - \mathbf{BK}$ is stable.

Recall that in the considered (continuous time) framework, a matrix $\mathbf{M}$ is stable if $\mathrm{Re}(\lambda_i) < 0$ for all eigenvalues λ_i of $\mathbf{M}$.

The idea of stabilizability is that all unstable modes of the system must be controllable, such that all eigenmodes of the matrix $\mathbf{A} - \mathbf{BK}$ can be made stable. That is formalized in the following theorem

Controllability and Stabilizability

If the system $(\mathbf{A}, \mathbf{B})$ is controllable, then it is stabilizable.

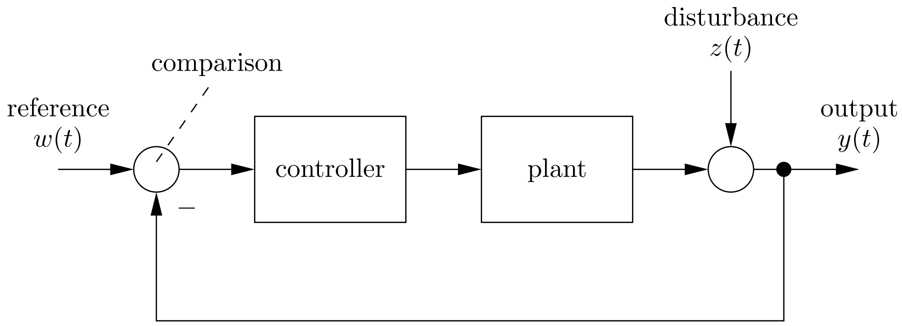
Proof (without).

The converse is not true: as an example, a stable system with some uncontrollable modes is stabilizable (by choosing e.g. $\mathbf{K} = \mathbf{0}$) but not controllable.

Chapter 3

State Feedback Control

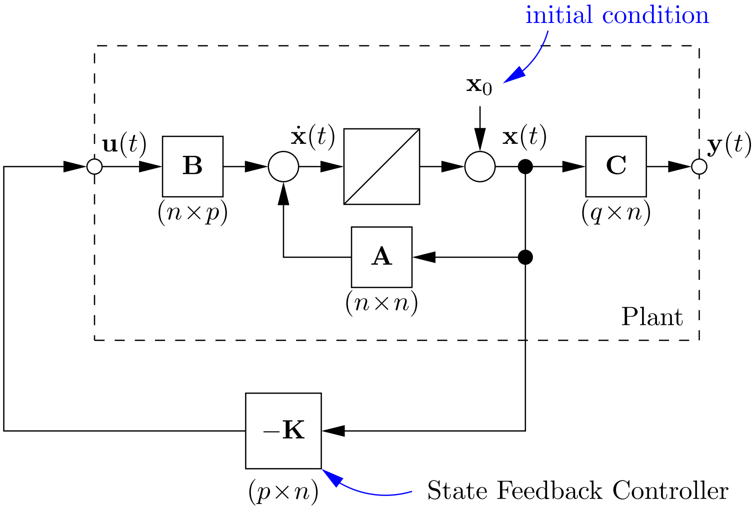
Before diving into the details of state feedback control, we remind ourselves of the 'classical' control loop



The distinguishing feature is the feedback of the output, which is compared to the reference value and thereby enables the control loop to compensate for disturbances $z(t) \neq 0$. In this chapter, the implementation of feedback controllers for state space systems will be discussed. Note that in the subsequent sections, we will focus on state feedback, which is different to the output feedback of the 'classical' control loop above. Firstly, this is done as equations become easier as for output feedback. Secondly and more importantly, state feedback can be implemented as there are methods to reconstruct the state from output measurements by observers, which will be discussed in chapter 5 in detail.

3.1 State Feedback

For further considerations, we assume $\mathbf{D} = \mathbf{0}$ to simplify notation. Now, a feedback is added to our state space system as follows



The state feedback controller is defined by

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

Inserting this equation into the state space ODE $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$ yields the following ODE for the feedback system

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t) \tag{3.2}$$

We now demand two requirements:

- **(REQ1)** Choose $\mathbf{K}$ such that the state space control loop is stable.

  $\Updownarrow$

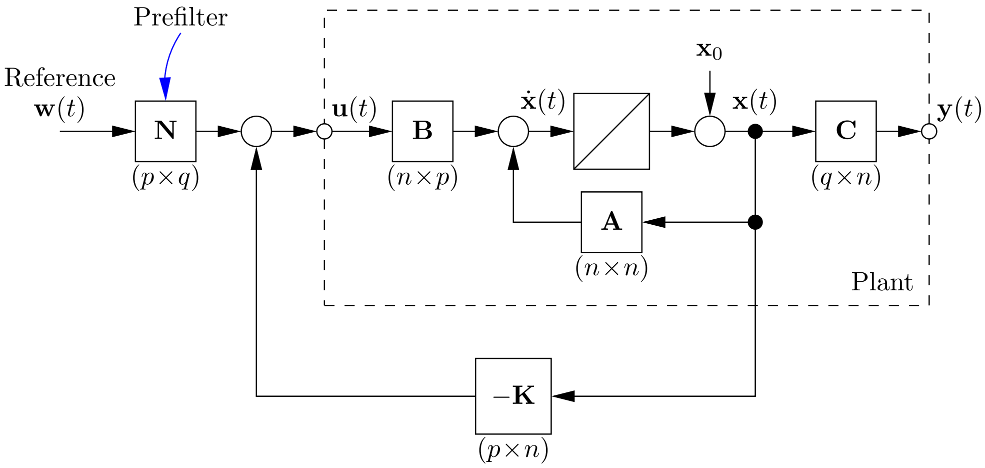
  For any initial value $\mathbf{x}_0 \neq \mathbf{0}$, $\mathbf{x}(t) \overset{t\to\infty}{\longrightarrow} \mathbf{0}$.

  $\Updownarrow$

  $(\mathbf{A} - \mathbf{B}\mathbf{K})$ is a stable matrix, i.e., all its eigenvalues have a negative real part.

- **(REQ2)** Introduce a reference input $\mathbf{w}$ and demand that the output vector $\mathbf{y}(t) \to \mathbf{w}$ for $t \to \infty$.

The second requirement can be achieved by adding a prefilter to the control feedback loop as follows



Before discussing implementation details, we would like to summarize some properties

1. A feedback feature was added to the plant.

2. The control law reads

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) + \mathbf{N}\mathbf{w} \tag{3.3}$$

It should be remarked that no classical 'comparison' of reference $\mathbf{w}$ and output values $\mathbf{y}$ is carried out.

3. The complete state vector $\mathbf{x}(t)$ (or at least an estimate, see chapter 5) may be needed for the controller implementation.

4. A disturbance is considered as initial condition $\mathbf{x}(t_0) = \mathbf{x}_0 \neq 0$, which corresponds to an 'initial kick' rather than to a persistent disturbance.

Control design task

1. Choose $\mathbf{K}$, $\mathbf{N}$ such that **REQ1** and **REQ2** are fulfilled.

2. Consider performance measures for the closed loop. The following two possibilities will be discussed further in detail

   - select eigenvalues and thereby determine speed and overshooting of the control loop (pole placement), see section 3.4.
   - minimize a quadratic performance index (LQR), see chapter 4.

3.2 (*) Prefilter

In the following, the prefilter will be discussed in order to achieve a certain set-point $\mathbf{w}_0$. It should be noted that most of the discussed control issues in the subsequent sections and chapters will be simplified to a zero set-point $\mathbf{x} \to \mathbf{0}$ controller for clarity of concepts. The reader should keep in mind that adding a prefilter as presented in this section will extend those to arbitrary set-points.

For determination of the prefilter we insert (3.3) into the ODE

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{3.4}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \tag{3.5}$$

and obtain for a constant (or at least step-wise constant) $\mathbf{w}(t) = \mathbf{w}_0$

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t) - \mathbf{B}\mathbf{N}\mathbf{w}_0 \tag{3.6}$$

The system is assumed to be stable ($\dot{\mathbf{x}}(t) \to \mathbf{0}$ for $t \to \infty$) and hence the state converges $\mathbf{x}(t) \to \mathbf{x}_\infty$. Inserting these two relations into (3.6) yields

$$\mathbf{0} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}_\infty + \mathbf{B}\mathbf{N}\mathbf{w}_0 \tag{3.7}$$

and with (3.5)

$$\mathbf{y}_\infty = \mathbf{C}(\mathbf{B}\mathbf{K} - \mathbf{A})^{-1}\mathbf{B}\mathbf{N}\mathbf{w}_0 \tag{3.8}$$

As we demand for $\mathbf{y}_\infty = \mathbf{w}_0$ **(REQ2)**, we get

$$\mathbf{C}(\mathbf{B}\mathbf{K} - \mathbf{A})^{-1}\mathbf{B}\mathbf{N} = \mathbf{I} \tag{3.9}$$

and for the prefilter

$$\mathbf{N} = \left(\mathbf{C}(\mathbf{B}\mathbf{K} - \mathbf{A})^{-1}\mathbf{B}\right)^{-1} \tag{3.10}$$

Without going further into detail, a final remark on the number of control variables shall be given. Regarding the dimensions of the matrices in (3.10)

$$\mathbf{N} = (\underbrace{\mathbf{C}}_{(q\times n)}\underbrace{(\mathbf{B}\mathbf{K} - \mathbf{A})}_{(n\times n)}{}^{-1}\underbrace{\mathbf{B}}_{(n\times p)})^{-1} \tag{3.11}$$

we get $(q \times p)$ for $\mathbf{N}$. Hence, it is invertible for $p = q$, i.e. for control of q output variables, q control variables (or more) are necessary. Note that this is different to controllability (section 2.1), where the value is given for a *certain* point in time. Here we demand that $\mathbf{y}(t)$ approaches $\mathbf{w}_0$ for $t \to \infty$.

3.3 Prefilter as a Reference Generator

The prefilter matrix $\mathbf{N}$ can also be obtained in a different way that can be interpreted as a reference generator. Though equivalent, the reference generator perspective is a bit more intuitive and also more robust against implementation errors and therefore slightly preferable. It replaces the control law in Eq. (3.3) by the control law

$$\mathbf{u}(t) = \mathbf{u}_{\mathrm{ss}} - \mathbf{K}(\mathbf{x}(t) - \mathbf{x}_{\mathrm{ss}}) \tag{3.12}$$

where the reference steady state values $\mathbf{u}_{\mathrm{ss}}$ and $\mathbf{x}_{\mathrm{ss}}$ are obtained from the desired reference value $\mathbf{w}$ via the linear maps

$$\mathbf{u}_{\mathrm{ss}} = \mathbf{N_u w} \quad \text{and} \quad \mathbf{x}_{\mathrm{ss}} = \mathbf{N_x w} \tag{3.13}$$

such that they satisfy the conditions that the reference is indeed in a steady state, i.e. that $0 = \mathbf{A}\mathbf{x}_{\mathrm{ss}} + B\mathbf{u}_{\mathrm{ss}}$, and that the output is at the desired reference value, i.e. that $\mathbf{w} = \mathbf{C}\mathbf{x}_{\mathrm{ss}}$. Together, this yields a linear system that the matrices $\mathbf{N_x}$ and $\mathbf{N_u}$ need to satisfy for all $\mathbf{w}$, namely

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{N_x} \\ \mathbf{N_u} \end{bmatrix} \mathbf{w} = \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix} \mathbf{w} \tag{3.14}$$

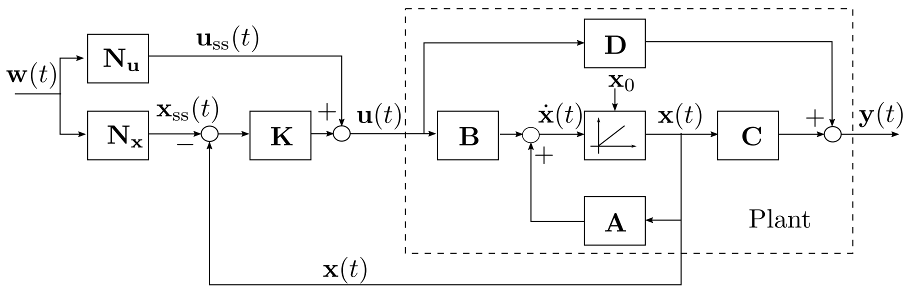
Assuming invertibility of the matrix on the left hand side, this yields the explicit expression

$$\begin{bmatrix} \mathbf{N_x} \\ \mathbf{N_u} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix} \tag{3.15}$$

Note that the matrices $\mathbf{N_u}$ and $\mathbf{N_x}$ do not depend on the feedback matrix K and can thus be computed independently from it. A comparison between Eq. (3.3) on the one hand and Eqs. (3.12) and (3.13) on the other hand shows that the prefilter matrix $\mathbf{N}$ in Eq. (3.3) could in principle also be obtained from the relation

$$\mathbf{N} = \mathbf{N_u} + \mathbf{K}\mathbf{N_x} \tag{3.16}$$

but in the reference generator implementation one would directly generate the control using Eq. (3.12), leading to the control diagram shown below, which is preferable in an actual implementation of the feedback controller, and which decouples the design of the prefilter in reference generator form from the design of the feedback controller.



3.4 Pole Placement for SISO Systems

Pole placement in our case means putting the *eigenvalues of the closed loop* to given values. Before explaining the method in detail, some brief hints how to choose the eigenvalues for the closed loop shall be summarized (some more information can be found on the slides).

- In order to achieve stability, all eigenvalues must be shifted to the left half plane, i.e. $\mathrm{Re}(\lambda_i) < 0$ for $i = 1, \ldots, n$.
- The location of the eigenvalues determines speed and overshooting/oscillations of the closed loop.

- For many systems, system dynamics is mainly determined by a dominant eigenvalue (or eigenvalue pair). In this situation, the focus should be put on this eigenvalue (pair).

For the pole placement, we assume that the system is given in control canonical form

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & \cdots & \cdots & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) && (3.17) \\
y(t) &= [b_0, \dots, b_{n-1}]\, \mathbf{x}(t) && (3.18)
\end{aligned}$$

The feedback controller is given by

$$u(t) = -\mathbf{k}^\top \mathbf{x}(t) \qquad \text{with} \qquad \mathbf{k}^\top = [k_0, \dots, k_{n-1}] \tag{3.19}$$

Then, the ODE for the feedback system read

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) = (\mathbf{A} - \mathbf{b}\mathbf{k}^\top)\mathbf{x}(t) \\
&= \underbrace{\begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ (-a_0 - k_0) & \cdots & \cdots & (-a_{n-1} - k_{n-1}) \end{bmatrix}}_{\mathbf{A}_{\mathrm{cl}} \doteq} \mathbf{x}(t) && (3.20)
\end{aligned}$$

with the thereby defined matrix for the closed loop $\mathbf{A}_{\mathrm{cl}}$. The idea is now to give the eigenvalues λ_i for $i = 1, \dots, n$ in order to obtain a certain dynamical behavior of the system. Hence, the characteristic polynomial reads

$$p_{\mathrm{cl}}(\lambda) = \prod_{i=1}^{n} (\lambda - \lambda_i) = \lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_1\lambda + p_0 \tag{3.21}$$

and defines the coefficients $p_0, \dots, p_n$. As the system is given in control canonical form, the coefficients of $p_{\mathrm{cl}}(\lambda)$ determine the last row of $\mathbf{A}_{\mathrm{cl}}$, hence

$$\mathbf{A}_{\mathrm{cl}} = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -p_0 & \cdots & \cdots & -p_{n-1} \end{bmatrix} \tag{3.22}$$

Comparison with (3.20) yields $-a_i - k_i = -p_i$ and hence for the coefficients of the controller $k_i = p_i - a_i$. In summary, we get the first rule for pole placement

Pole Placement

Assume a system in <u>control canonical</u> form with characteristic polynomial (CP)

$$\lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda + a_0 \tag{3.23}$$

A given CP (calculated from given eigenvalues) for the closed loop

$$\lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_1\lambda + p_0 \tag{3.24}$$

is implemented by the state feedback controller with vector

$$\mathbf{k}^\top = [(p_0 - a_0), \dots, (p_{n-1} - a_{n-1})] \tag{3.25}$$

For systems not given in control canonical form, the control feedback might be determined by calculating the characteristic polynomial of the closed loop

$$p_{\mathrm{cl}}(\lambda) \doteq \det(\lambda \mathbf{I} - \mathbf{A} + \mathbf{b}\mathbf{k}^\top) \tag{3.26}$$

and comparing the coefficients. This will be demonstrated by the following simple example. Assume a system given by

$$\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 0 & -1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{3.27}$$

The eigenvalues (poles) shall be placed at $\lambda_1 = -1$ and $\lambda_2 = -2$. Hence the given characteristic polynomial for the closed loop is

$$p_{\mathrm{cl}}(\lambda) = (\lambda - \lambda_1)(\lambda - \lambda_2) = \lambda^2 + (-\lambda_1 - \lambda_2)\lambda + (\lambda_1 \lambda_2) \tag{3.28}$$

This must be same as computed by using (3.26)

$$\begin{aligned} p_{\mathrm{cl}}(\lambda) &= \det(\lambda \mathbf{I} - \mathbf{A} + \mathbf{b}\mathbf{k}^\top) = \det(\begin{bmatrix} \lambda - 1 + k_1 & -3 + k_2 \\ k_1 & \lambda + 1 + k_2 \end{bmatrix}) \\ &= \lambda^2 + \underbrace{(k_1 + k_2)}_{=-\lambda_1 - \lambda_2 = 3} + \underbrace{4k_1 - k_2 - 1}_{=\lambda_1 \lambda_2 = 2} \end{aligned} \tag{3.29}$$

Comparing the coefficients as indicated results in

$$\mathbf{k}^\top = [k_1, k_2] = [1.2, 1.8] \tag{3.30}$$

3.5 (*) Transformation to Control Canonical Form and Ackermann's Formula for LTI-SISO Systems

In the following, we will apply the pole placement method to systems given in another than control canonical form. The procedure is to consider the transformation to control canonical form first and then derive a general formula for $\mathbf{k}^\top$.

Transformation to Control Canonical Form

The transformation $\mathbf{T}$, defining the new state vector $\mathbf{z}(t) = \mathbf{T}\mathbf{x}(t)$ and applied to

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t) \tag{3.31}$$

results in the control canonical form

$$\dot{\mathbf{z}}(t) = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & \cdots & \cdots & -a_{n-1} \end{bmatrix} \mathbf{z}(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \tag{3.32}$$

for

$$\mathbf{T} = \begin{bmatrix} \mathbf{t}_1^\top \\ \mathbf{t}_1^\top \mathbf{A} \\ \vdots \\ \mathbf{t}_1^\top \mathbf{A}^{n-1} \end{bmatrix} \tag{3.33}$$

where $\mathbf{t}_1^\top$ is the last row of the inverse controllability matrix

$$\mathcal{C}^{-1} = \left[\mathbf{B}, \mathbf{A}\mathbf{B}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right]^{-1} \tag{3.34}$$

Note that the system must be controllable for calculation of the inverse $\mathcal{C}^{-1}$.

Proof (sketch): Applying the transformation $\mathbf{z}(t) = \mathbf{T}\mathbf{x}(t)$ to (3.31) yields

$$\begin{aligned} \mathbf{T}\dot{\mathbf{x}}(t) &= \mathbf{T}\mathbf{A}\underbrace{\mathbf{T}^{-1}\mathbf{T}}_{=\mathbf{I}}\mathbf{x}(t) + \mathbf{T}\mathbf{b}u(t) \\ \hookrightarrow \quad \dot{\mathbf{z}}(t) &= \mathbf{T}\mathbf{A}\mathbf{T}^{-1}\mathbf{z}(t) + \mathbf{T}\mathbf{b}u(t) \end{aligned} \tag{3.35}$$

The following two steps show that $\mathbf{Tb}$ and $\mathbf{TAT}^{-1}$ are the respective matrices of the control canonical form (3.32).

1. Using the definitions of $\mathcal{C}$, $\mathbf{t}_1^\top$ and the relation $\mathcal{C}^{-1}\mathcal{C} = \mathbf{I}$ we get

$$\begin{bmatrix} \star \\ \star \\ \vdots \\ \mathbf{t}_1^\top \end{bmatrix} \left[\mathbf{B}, \mathbf{AB}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right] = \mathbf{I} \tag{3.36}$$

   Consideration of the last row yields

$$\mathbf{t}_1^\top \mathbf{A}^\nu \mathbf{b} = 0 \qquad \nu = 1, \ldots, (n-2) \tag{3.37}$$

$$\mathbf{t}_1^\top \mathbf{A}^{n-1} \mathbf{b} = 1 \tag{3.38}$$

   and hence

$$\mathbf{Tb} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{3.39}$$

2. It remains to show that

$$\mathbf{TAT}^{-1} = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & \cdots & \cdots & -a_{n-1} \end{bmatrix} \tag{3.40}$$

   which can be written as

$$\mathbf{TA} = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & \cdots & \cdots & -a_{n-1} \end{bmatrix} \mathbf{T} \tag{3.41}$$

   Inserting the definition of $\mathbf{t}_1^\top$ (3.33) yields

$$\begin{bmatrix} \mathbf{t}_1^\top \\ \mathbf{t}_1^\top \mathbf{A} \\ \vdots \\ \mathbf{t}_1^\top \mathbf{A}^{n-2} \\ \mathbf{t}_1^\top \mathbf{A}^{n-1} \end{bmatrix} \mathbf{A} = \begin{bmatrix} \mathbf{t}_1^\top \mathbf{A} \\ \mathbf{t}_1^\top \mathbf{A}^2 \\ \vdots \\ \mathbf{t}_1^\top \mathbf{A}^{n-1} \\ (-a_0\mathbf{t}_1^\top - a_1\mathbf{t}_1^\top\mathbf{A} - \cdots - a_{n-1}\mathbf{t}_1^\top\mathbf{A}^{n-1}) \end{bmatrix} \tag{3.42}$$

   The equality of the rows can be easily recognized except for the last row which reads

$$\mathbf{t}_1^\top \mathbf{A}^n = -a_0\mathbf{t}_1^\top - a_1\mathbf{t}_1^\top\mathbf{A} - \cdots - a_{n-1}\mathbf{t}_1^\top\mathbf{A}^{n-1} \tag{3.43}$$

   This relation can be shown using the theorem of Cayley-Hamilton

$$\mathbf{t}_1^\top(a_0 + a_1\mathbf{A} + \cdots + a_{n-1}\mathbf{A}^{n-1} + \mathbf{A}^n) = \mathbf{t}_1^\top p(\mathbf{A}) = 0 \tag{3.44}$$

$\square$

We now consider the pole placement

$$u(t) = -\tilde{\mathbf{k}}^\top \mathbf{z}(t) = -\underbrace{\tilde{\mathbf{k}}^\top \mathbf{T}}_{\mathbf{k}^\top}\mathbf{x}(t) \tag{3.45}$$

The state feedback can be calculated as

$$\begin{aligned}
\mathbf{k}^\top &= \tilde{\mathbf{k}}^\top \mathbf{T} && (3.46)\\
&= [(p_0 - a_0), \ldots, (p_{n-1} - a_{n-1})] \begin{bmatrix} \mathbf{t}_1^\top \\ \mathbf{t}_1^\top \mathbf{A} \\ \vdots \\ \mathbf{t}_1^\top \mathbf{A}^{n-1} \end{bmatrix} && (3.47)\\
&= (p_0 - a_0)\mathbf{t}_1^\top + (p_1 - a_1)\mathbf{t}_1^\top \mathbf{A} + \cdots + (p_{n-1} - a_{n-1})\mathbf{t}_1^\top \mathbf{A}^{n-1} && (3.48)\\
&= \mathbf{t}_1^\top (p_0 + p_1 \mathbf{A} + \cdots + p_{n-1}\mathbf{A}^{n-1} \underbrace{-a_0 - a_1 \mathbf{A} - \cdots - a_{n-1}\mathbf{A}^{n-1}}_{=\mathbf{A}^n \text{ as } p_{\mathbf{A}}(\mathbf{A})=0} && (3.49)\\
&= \mathbf{t}_1^\top (p_0 + p_1 \mathbf{A} + \cdots + p_{n-1}\mathbf{A}^{n-1} + \mathbf{A}^n) && (3.50)\\
&= \mathbf{t}_1^\top p(\mathbf{A}) && (3.51)
\end{aligned}$$

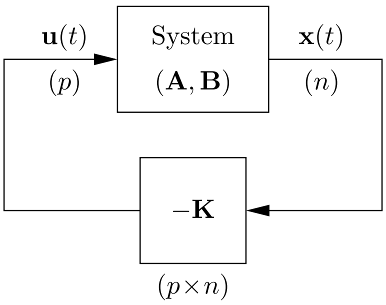
This controller realization is called Ackermann's formula, which shall be summarized

Pole Placement (Ackermann's Formula)

Given the characteristic polynomial $p(\lambda)$ for the closed loop, the control feedback has to be chosen as $\mathbf{k}^\top = \mathbf{t}_1^\top p(\mathbf{A})$ where $\mathbf{t}_1^\top$ is the last row of the inverse controllability matrix $\mathcal{C}^{-1} = \left[\mathbf{B}, \mathbf{AB}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right]^{-1}$.

3.6 (*) Modal Control for MIMO Systems

Consider the state feedback for a MIMO system given as follows



For SISO systems the feedback matrix $\mathbf{K}$ has the dimensions $(1 \times n)$ and is determined uniquely by given n eigenvalues. For MIMO system there is an ambiguity and an infinite number of possible feedback realizations for n given eigenvalues. Hence, different controller design principles have to be applied.

The idea of modal control is the following: for p control variables the eigenvalues for p observable modes are given in order to define the feedback controller. In other words, the eigenvalues of p modes are "shifted" towards desired design values. This approach is denoted *modal control.*

Considering the problem in modal canonical form (assume $\lambda_i \neq \lambda_j$ for $i \neq j$), we would like to get

$$\dot{\mathbf{z}}(t) = \begin{bmatrix} \lambda_1 & & & & & \\ & \ddots & & & & \\ & & \lambda_p & & & \\ & & & \lambda_{p+1} & & \\ & & & & \ddots & \\ & & & & & \lambda_n \end{bmatrix} \mathbf{z}(t) - \begin{bmatrix} (\lambda_1 - \lambda_{\mathrm{cl},1}) & & & 0 & \cdots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & (\lambda_p - \lambda_{\mathrm{cl},p}) & 0 & \cdots & 0 \\ \star & \cdots & \star & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ \star & \cdots & \star & 0 & \cdots & 0 \end{bmatrix} \mathbf{z}(t) \tag{3.52}$$

where $\lambda_{\mathrm{cl},i}$, $i = 1, \dots, p$ are the new eigenvalues of the closed loop. Note that we ordered the state variables in $\mathbf{z}$ such that the first p eigenvalues are to be shifted. From (3.52) we get

$$\begin{array}{rclll} \dot{z}_i(t) & = & \lambda_{\mathrm{cl},i} z(t) & i = 1, \dots, p \\ \dot{z}_i(t) & = & \lambda_i z_i(t) + \text{coupling } (\star) & i = (p+1), \dots, n \end{array} \tag{3.53}$$

Thus, the first p eigenvalues are shifted while the remaining eigenvalues are unchanged.

Modal Control Feedback

For given eigenvalues $\lambda_{\mathrm{cl},1}, \dots, \lambda_{\mathrm{cl},p}$ for the closed loop, the control feedback is given by

$$\mathbf{u}(t) = -\overbrace{\begin{bmatrix} \mathbf{w}_1^\top \mathbf{B} \\ \vdots \\ \mathbf{w}_p^\top \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} (\lambda_1 - \lambda_{\mathrm{cl},1}) & & \\ & \ddots & \\ & & (\lambda_p - \lambda_{\mathrm{cl},p}) \end{bmatrix} \begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_p^\top \end{bmatrix}}^{\mathbf{K} \doteq} \mathbf{x}(t) \tag{3.54}$$

where $\lambda_1, \dots, \lambda_p$ are the eigenvalues and $\mathbf{w}_1^\top, \dots, \mathbf{w}_p^\top$ are the left eigenvectors of $\mathbf{A}$.

Proof: consider transformation $\mathbf{z}(t) = \mathbf{V}^{-1}\mathbf{x}(t)$ for

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + B\mathbf{u}(t) \tag{3.55}$$

$$\mathbf{V}^{-1}\mathbf{x}(t) = \mathbf{V}^{-1}\mathbf{A}\underbrace{V\mathbf{V}^{-1}}_{\mathbf{I}}\mathbf{x}(t) + \mathbf{V}^{-1}\mathbf{B}\mathbf{u}(t) \tag{3.56}$$

$$\begin{aligned} \hookrightarrow \mathbf{z}(t) &= \mathbf{\Lambda}\mathbf{z}(t) + \mathbf{V}^{-1}\mathbf{B}\mathbf{u}(t) \\ &= \mathbf{\Lambda}\mathbf{z}(t) - \underbrace{\mathbf{V}^{-1}\mathbf{B}\mathbf{K}\mathbf{V}}_{\tilde{\mathbf{K}}}\mathbf{z}(t) \end{aligned} \tag{3.57}$$

We now derive $\tilde{\mathbf{K}}$ using (3.54)

$$\tilde{\mathbf{K}} = \mathbf{V}^{-1}\mathbf{B} \begin{bmatrix} \mathbf{w}_1^\top \mathbf{B} \\ \vdots \\ \mathbf{w}_p^\top \mathbf{B} \end{bmatrix}^{-1} \begin{bmatrix} (\lambda_1 - \lambda_{\mathrm{cl},1}) & & \\ & \ddots & \\ & & (\lambda_p - \lambda_{\mathrm{cl},p}) \end{bmatrix} \begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_p^\top \end{bmatrix} \tag{3.58}$$

$$= \underbrace{\begin{bmatrix} \mathbf{w}_1^\top \mathbf{B} \\ \vdots \\ \mathbf{w}_p^\top \mathbf{B} \\ \mathbf{w}_{p+1}^\top \mathbf{B} \\ \vdots \\ \mathbf{w}_n^\top \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1^\top \mathbf{B} \\ \vdots \\ \mathbf{w}_p^\top \mathbf{B} \end{bmatrix}^{-1}}_{\mathbf{M}_1} \begin{bmatrix} (\lambda_1 - \lambda_{\mathrm{cl},1}) & & \\ & \ddots & \\ & & (\lambda_p - \lambda_{\mathrm{cl},p}) \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{w}_1^\top \\ \vdots \\ \mathbf{w}_p^\top \end{bmatrix} [\mathbf{v}_1, \dots, \mathbf{v}_n]}_{\mathbf{M}_2}$$

where $\mathbf{M}_1$ and $\mathbf{M}_2$ can be computed using the definitions of $\mathbf{w}_i^\top$ and $\mathbf{v}_i$

$$\mathbf{M}_1 = \underbrace{\begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \\ \star & \cdots & \star \\ \vdots & & \vdots \\ \star & \cdots & \star \end{bmatrix}}_{(n\times p)} \qquad \mathbf{M}_2 = \underbrace{\begin{bmatrix} 1 & & & 0 & \cdots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & 1 & 0 & \cdots & 0 \end{bmatrix}}_{(p\times n)} \tag{3.59}$$

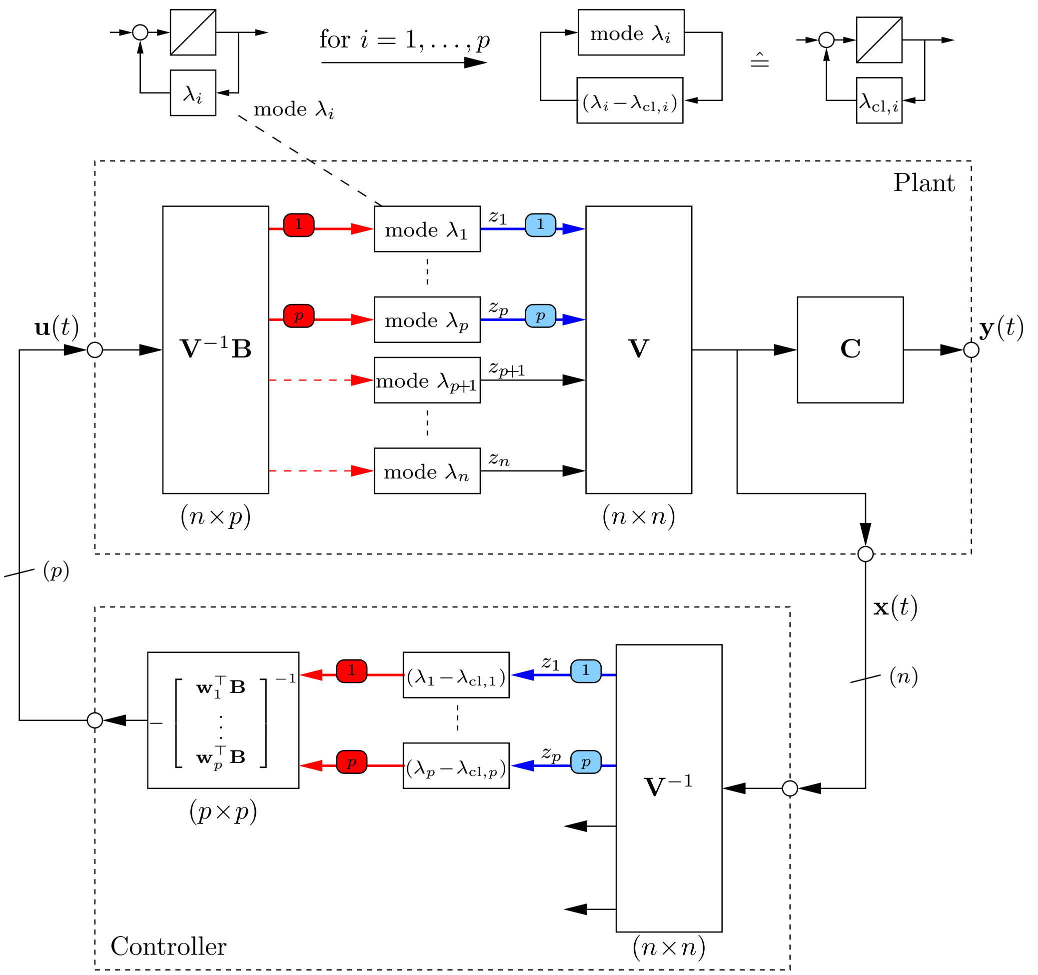
We finally get

$$\tilde{\mathbf{K}} = \begin{bmatrix} (\lambda_1 - \lambda_{\mathrm{cl},1}) & & & 0 & \cdots & 0 \\ & \ddots & & \vdots & & \vdots \\ & & (\lambda_p - \lambda_{\mathrm{cl},p}) & 0 & \cdots & 0 \\ \star & \cdots & \star & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ \star & \cdots & \star & 0 & \cdots & 0 \end{bmatrix} \tag{3.60}$$

which is equivalent to the feedback matrix in (3.52).

$\square$

The modal control feedback can be depicted as follows



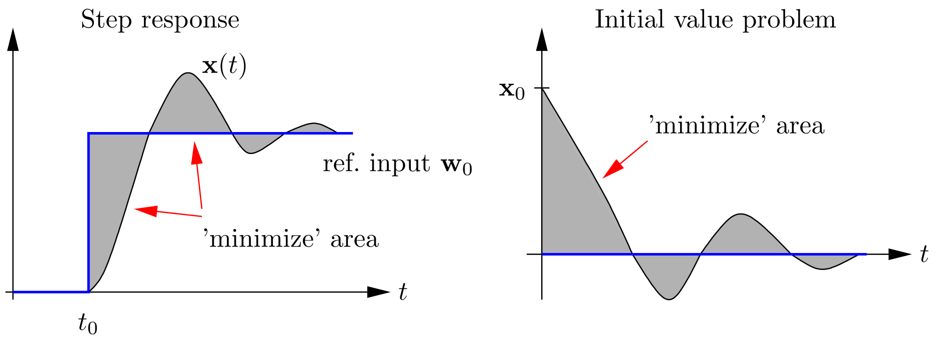
The controller 'picks out' the first p eigenmodes and 'shifts them towards' the desired values $\lambda_{\mathrm{cl},1}, \ldots, \lambda_{\mathrm{cl},p}$. Note that a coupling to the remaining modes is introduced as indicated by the red dashed arrows. However, the coupling does not modify the other eigenvalues $\lambda_{p+1}, \ldots, \lambda_n$.

It should be finally noted that there are even more general ways to set up the feedback. In the case of $p = n$, all n eigenvalues and n eigenvectors of the closed loop could be 'designed' independently.

Chapter 4

Linear Quadratic Regulator (LQR)

The idea is to introduce and optimize a performance index as depicted in the following



For a good controller performance, one would demand for a fast response and little overshooting, hence for minimizing the shaded areas. The performance index for the LQR controller is introduced as

$$J\{\mathbf{x},\mathbf{u}\} = \frac{1}{2}\int_0^\infty \left(\mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^\top(t)\mathbf{R}\mathbf{u}(t)\right)\,\mathrm{d}t \tag{4.1}$$

where $\mathbf{Q}$ is a positive definite $(n \times n)$ matrix and $\mathbf{R}$ a positive definite $(p \times p)$ matrix. The vector $\mathbf{x}(t)$ is the solution of the ODE of the system with initial condition $\mathbf{x}_0$ and $\mathbf{u}(t)$ the according steering input. The matrices $\mathbf{Q}$ and $\mathbf{R}$ can be regarded as tuning parameters in order to meet design requirements. While $\mathbf{Q}$ penalizes slow responses and overshoots, $\mathbf{R}$ adds a penalization to steering actuation. Although this interpretation is relevant for some applications e.g. saving steering gas in satellites, it could be generally regarded as a general knob in order to achieve the desired controller behavior. Note, that both terms in the integral are quadratic measures (instead of a norm measure indicated by the colored areas in the figure above).

Now, the control task is to find a feedback matrix $\mathbf{K}$

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \tag{4.2}$$

such that J is minimized. This could be formally written as

$$\mathbf{K} = \arg\min_{\mathbf{K}'}\Big|_{\mathbf{u}=-\mathbf{K}'\mathbf{x}} J\{\mathbf{x},\mathbf{u}\} \tag{4.3}$$

4.1 Lyapunov Equation

Before tackling the problem with the whole performance index (4.1), the problem shall be solved for quadratic functionals, thus considering

$$J = \frac{1}{2}\int_0^\infty \mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t)\,\mathrm{d}t \tag{4.4}$$

With the homogeneous solution for the system ODE $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t)$

$$\mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{x}_0 \tag{4.5}$$

we get

$$J = \frac{1}{2}\int_0^\infty \mathbf{x}_0^\top e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\mathbf{x}_0\,\mathrm{d}t = \frac{1}{2}\mathbf{x}_0^\top\left(\int_0^\infty e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\,\mathrm{d}t\right)\mathbf{x}_0 \tag{4.6}$$

By defining

$$P \doteq \int_0^\infty e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\,\mathrm{d}t \tag{4.7}$$

we get the performance index as function of the initial condition $\mathbf{x}_0$

$$J = \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 \tag{4.8}$$

In the following, we derive an equation for $\mathbf{P}$ by partial integration of the definition (4.7)

$$P = \int_0^\infty e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\,\mathrm{d}t = \underbrace{\left[e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\mathbf{A}^{-1}\right]_0^\infty}_{-\mathbf{Q}\mathbf{A}^{-1}} - \underbrace{\int_0^\infty \mathbf{A}^\top e^{\mathbf{A}^\top t}\mathbf{Q}e^{\mathbf{A}t}\mathbf{A}^{-1}\,\mathrm{d}t}_{\mathbf{A}^\top\mathbf{P}\mathbf{A}^{-1}} \tag{4.9}$$

The two terms on the RHS result in $-\mathbf{Q}\mathbf{A}^{-1}$ as $e^{\mathbf{A}t} \overset{t\to\infty}{\to} 0$ for a stable system and in $\mathbf{A}^\top\mathbf{P}\mathbf{A}^{-1}$ by using the definition (4.7). The resulting equation

$$\mathbf{P} = -\mathbf{Q}\mathbf{A}^{-1} - \mathbf{A}^\top\mathbf{P}\mathbf{A}^{-1} \tag{4.10}$$

is multiplied with $\mathbf{A}$ from the right hand side to obtain the

$$\boxed{\text{Lyapunov Equation:}\qquad \mathbf{P}\mathbf{A} + \mathbf{A}^\top\mathbf{P} = -\mathbf{Q}} \tag{4.11}$$

This equation allows for calculation of $\mathbf{P}$ from the system matrix $\mathbf{A}$ and the weighting matrix $\mathbf{Q}$.

4.2 Optimal Controller

We now come back to controller design and consider

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \qquad \hookrightarrow \qquad \mathbf{u}^\top(t) = -\mathbf{x}^\top(t)\mathbf{K}^\top \tag{4.12}$$

For any symmetric matrix $\mathbf{P}$, let us consider the function

$$\mathbf{V}(t) = \frac{1}{2}\mathbf{x}^\top(t)\mathbf{P}\mathbf{x}(t) \tag{4.13}$$

In the first part of the derivation, we will derive an expression true for any value of $\mathbf{P}$. The total derivative of $\mathbf{V}$ w.r.t. t is

$$
\begin{aligned}
\dot{\mathbf{V}}(\mathbf{x}(t)) &= \frac{1}{2}\dot{\mathbf{x}}(t)\mathbf{P}\mathbf{x}(t) + \frac{1}{2}\mathbf{x}(t)\mathbf{P}\dot{\mathbf{x}}(t) \\
&= \frac{1}{2}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t))\mathbf{P}\mathbf{x}(t) + \frac{1}{2}\mathbf{x}(t)\mathbf{P}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) \\
&= \frac{1}{2}\begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} & \mathbf{P}\mathbf{B} \\ \mathbf{B}^\top\mathbf{P} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}
\end{aligned}
$$

By integrating this expression from 0 to ∞ (assuming that $\mathbf{x} \to 0$ as $t \to \infty$) and using the fundamental theorem of calculus we have

$$
\begin{aligned}
\int_0^\infty \dot{\mathbf{V}}(t)\mathrm{d}t = \mathbf{V}(\infty) - \mathbf{V}(0) = 0 - \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 = \\
= \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} & \mathbf{P}\mathbf{B} \\ \mathbf{B}^\top\mathbf{P} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t
\end{aligned}
$$

that gives

$$
0 = \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} & \mathbf{P}\mathbf{B} \\ \mathbf{B}^\top\mathbf{P} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t + \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0
$$

and therefore, since this expression is always equal to 0 for any value of the matrix $\mathbf{P}$, it can be added to any other expression without changing its value.

In particular, we can add it to the expression of the performance index J rewritten in matrix form,

$$
\begin{aligned}
J &= \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{Q} & 0 \\ 0 & \mathbf{R}\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t = \\
&= \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} + \mathbf{Q} & \mathbf{P}\mathbf{B} \\ \mathbf{B}^\top\mathbf{P} & \mathbf{R}\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t + \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0
\end{aligned}
$$

Again, the above expression holds for any value of $\mathbf{P}$. Now we make a choice for $\mathbf{P}$, that we take as the solution of the matrix Riccati equation:

$$
\boxed{\text{Matrix-Riccati-Equation:} \qquad \mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} + \mathbf{Q} = \mathbf{0}} \tag{4.14}
$$

By using the expression $\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} + \mathbf{Q} = \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}$ it is possible to 'complete the square', and the performance index J becomes

$$
\begin{aligned}
J &= \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} & \\ \mathbf{B}^\top\mathbf{P} & \mathbf{R}\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t + \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 = \\
&= \frac{1}{2}\int_0^\infty \begin{bmatrix}\mathbf{x}^\top(t) & \mathbf{u}^\top(t)\end{bmatrix}\begin{bmatrix}\mathbf{P}\mathbf{B}\mathbf{R}^{-1} \\ \mathbf{I}\end{bmatrix}\mathbf{R}\begin{bmatrix}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{x}(t) \\ \mathbf{u}(t)\end{bmatrix}\mathrm{d}t + \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 \\
&= \frac{1}{2}\int_0^\infty \mathbf{w}^\top(t)\mathbf{R}\mathbf{w}(t)\mathrm{d}t + \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0
\end{aligned}
$$

Since the matrix R is positive definite, the integrand is always positive or zero, and the minimum (optimal) value zero is attained only for

$$
0 = \mathbf{w}(t) = \mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}\mathbf{x}(t) + \mathbf{u}(t)
$$

that gives the optimal input as the state feedback

$$
\begin{aligned}
\mathbf{u}(t) &= -\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}\mathbf{x}(t) \\
&= -\mathbf{K}\mathbf{x}(t)
\end{aligned}
$$

with feedback matrix $\mathbf{K}$

$$
\boxed{\text{Optimal LQR Controller:} \qquad \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}} \tag{4.15}
$$

4.3 (*) Optimal Controller - alternative derivation

We now come back to controller design and consider

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) \qquad \hookrightarrow \qquad \mathbf{u}^\top(t) = -\mathbf{x}^\top(t)\mathbf{K}^\top \tag{4.16}$$

For the ODE of the closed loop we get

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) - \mathbf{B}\mathbf{K}\mathbf{x}(t) = (\underbrace{\mathbf{A} - \mathbf{B}\mathbf{K}}_{\mathbf{A}_{\text{cl}} \doteq})\mathbf{x}(t) \tag{4.17}$$

Hence with the definition

$$\mathbf{A}_{\text{cl}} \doteq \mathbf{A} - \mathbf{B}\mathbf{K} \tag{4.18}$$

we have the ODE

$$\dot{\mathbf{x}}(t) = \mathbf{A}_{\text{cl}}\mathbf{x}(t) \tag{4.19}$$

The performance index (4.1) now reads

$$\begin{aligned} J &= \frac{1}{2}\int_0^\infty \big(\mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^\top(t)\mathbf{R}\mathbf{u}(t)\big)\,\mathrm{d}t = \frac{1}{2}\int_0^\infty \big(\mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{x}^\top(t)\mathbf{K}^\top\mathbf{R}\mathbf{K}\mathbf{x}(t)\big)\,\mathrm{d}t \\ &= \frac{1}{2}\int_0^\infty \mathbf{x}^\top(t)\mathbf{Q}_{\text{cl}}\mathbf{x}(t)\,\mathrm{d}t \end{aligned} \tag{4.20}$$

with

$$\mathbf{Q}_{\text{cl}} \doteq \mathbf{Q} + \mathbf{K}^\top\mathbf{R}\mathbf{K} \tag{4.21}$$

For $\mathbf{P}$, defined by

$$J = \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 \tag{4.22}$$

the Lyapunov equation for the *closed loop* reads

$$\mathbf{P}\mathbf{A}_{\text{cl}} + \mathbf{A}_{\text{cl}}^\top\mathbf{P} = -\mathbf{Q}_{\text{cl}} \tag{4.23}$$

In order to compute the matrix $\mathbf{K}$ leading to an optimum (minimum) of J, we demand

$$\frac{\partial J}{\partial k_{ij}} = \frac{\partial}{\partial k_{ij}}\frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0 = 0 \tag{4.24}$$

for $i = 1, \ldots, p$ and $j = 1, \ldots, n$. The k_{ij} denote the elements of $\mathbf{K}$. In the following, we consider the optimality condition for all $\mathbf{x}_0 \in \mathbb{R}^n$ (which is reasonable for state feedback systems) and thus are allowed to reduce the condition to

$$\frac{\partial}{\partial k_{ij}}\mathbf{P} = 0 \tag{4.25}$$

The element-wise partial derivative $\frac{\partial}{\partial k_{ij}}$ of (4.23) yields

$$\underbrace{\frac{\partial \mathbf{P}}{\partial k_{ij}}}_{=0}\mathbf{A}_{\text{cl}} + \mathbf{P}\frac{\partial \mathbf{A}_{\text{cl}}}{\partial k_{ij}} + \frac{\partial \mathbf{A}_{\text{cl}}^\top}{\partial k_{ij}}\mathbf{P} + \mathbf{A}_{\text{cl}}^\top\underbrace{\frac{\partial \mathbf{P}}{\partial k_{ij}}}_{=0} = -\frac{\partial \mathbf{Q}_{\text{cl}}}{\partial k_{ij}} \tag{4.26}$$

With

$$\mathbf{A}_{\text{cl}} = \mathbf{A} - \mathbf{B}\mathbf{K} \qquad \text{and} \qquad \mathbf{Q}_{\text{cl}} = \mathbf{Q} + \mathbf{K}^\top\mathbf{R}\mathbf{K} \tag{4.27}$$

we get

$$-\mathbf{P}\mathbf{B}\frac{\partial \mathbf{K}}{\partial k_{ij}} - \frac{\partial \mathbf{K}^\top}{\partial k_{ij}}\mathbf{B}^\top\mathbf{P} = -\frac{\partial \mathbf{K}^\top}{\partial k_{ij}}\mathbf{R}\mathbf{K} - \mathbf{K}^\top\mathbf{R}\frac{\partial \mathbf{K}}{\partial k_{ij}} \tag{4.28}$$

and

$$\frac{\partial \mathbf{K}^\top}{\partial k_{ij}}(\mathbf{RK} - \mathbf{B}^\top\mathbf{P}) = (\mathbf{PB} - \mathbf{K}^\top\mathbf{R})\frac{\partial \mathbf{K}}{\partial k_{ij}} \tag{4.29}$$

The above equation has to be satisfied for all indices i, j. This yields

$$\mathbf{RK} - \mathbf{B}^\top\mathbf{P} = \mathbf{0} \tag{4.30}$$

As result, we get the

$$\boxed{\text{Optimal LQR Controller:} \qquad \mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}} \tag{4.31}$$

Note, that it depends on matrix $\mathbf{P}$, which will be computed in the following. Inserting (4.31) into (4.18) and (4.21) results in

$$\mathbf{A}_{\text{cl}} = \mathbf{A} - \mathbf{BK} = \mathbf{A} - \mathbf{BR}^{-1}\mathbf{B}^\top\mathbf{P} \tag{4.32}$$

and

$$\mathbf{Q}_{\text{cl}} = \mathbf{Q} + \mathbf{K}^\top\mathbf{RK} = \mathbf{Q} + \mathbf{P}^\top\mathbf{B}(\mathbf{R}^{-1})^\top\mathbf{RR}^{-1}\mathbf{B}^\top\mathbf{P} = \mathbf{Q} + \mathbf{P}^\top\mathbf{B}(\mathbf{R}^{-1})^\top\mathbf{B}^\top\mathbf{P} \tag{4.33}$$

Insertion of these relations into (4.23) yields

$$\mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^\top\mathbf{P} + \mathbf{A}^\top\mathbf{P} - \underbrace{\mathbf{P}^\top\mathbf{B}(\mathbf{R}^{-1})^\top\mathbf{B}^\top\mathbf{P}}_{(\star 1)} = -\mathbf{Q} - \underbrace{\mathbf{P}^\top\mathbf{B}(\mathbf{R}^{-1})^\top\mathbf{B}^\top\mathbf{P}}_{(\star 1)} \tag{4.34}$$

As result we get the

$$\boxed{\text{Matrix-Riccati-Equation:} \qquad \mathbf{A}^\top\mathbf{P} + \mathbf{PA} - \mathbf{PBR}^{-1}\mathbf{B}^\top\mathbf{P} + \mathbf{Q} = \mathbf{0}} \tag{4.35}$$

We recall, that $(\mathbf{A}, \mathbf{B})$ is the state space description of the plant and $\mathbf{Q}$, $\mathbf{R}$ are the given parameter matrices. The basic steps are to use (4.35) for computation of $\mathbf{P}$ and then determining $\mathbf{K}$ via (4.31). Two final notes for the given LQR design shall be noted for sake of completeness without further discussion of details: the system $(\mathbf{A}, \mathbf{B})$ has to be controllable and the system $(\mathbf{A}, \bar{\mathbf{Q}})$ has to be observable, where $\bar{\mathbf{Q}}$ is given by $\mathbf{Q} = \bar{\mathbf{Q}}^\top\bar{\mathbf{Q}}$.

4.4 Choice of Q and R Matrices

In this section, some rough ideas on the choices of the $\mathbf{Q}$ and $\mathbf{R}$ matrices shall be given.

- As the choice of the $\mathbf{Q}$ and $\mathbf{R}$ matrices is crucial for the result, the LQR concept should be regarded more as a mathematical recipe for carrying out the controller design rather than as a self-contained procedure, which comes up with the 'optimal' controller. In practice one would choose certain matrices $\mathbf{Q}$ and $\mathbf{R}$, then compute the controller based on these matrices and compare simulations to given specifications. Eventually, the whole design process has to be repeated with different $\mathbf{Q}$ and $\mathbf{R}$ matrices to end up at the desired controller behavior after some iterations.

- As a thumb rule, one could start with diagonal matrices and choose

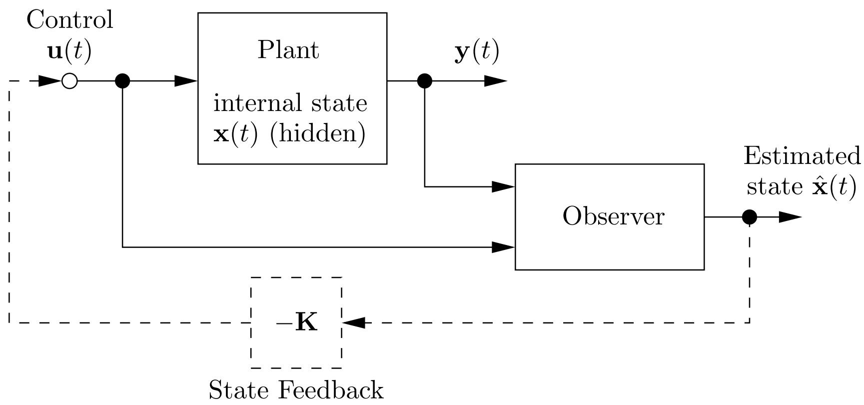
$$q_{i,i} = \frac{1}{\text{Maximum acceptable value for } x_i^2} \qquad i = 1, \ldots, n \tag{4.36}$$

$$r_{j,j} = \frac{1}{\text{Maximum acceptable value for } u_j^2} \qquad j = 1, \ldots, p \tag{4.37}$$

Chapter 5

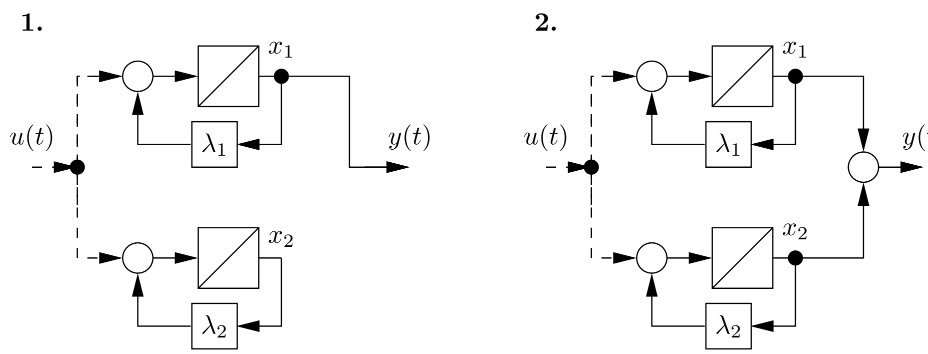
Observability, State Estimation and Kalman Filter

The task of an observer (also known as state estimator) is to reconstruct the (hidden) state vector of an system, especially in order to implement a state feedback controller, which is based on the knowledge of the state vector. This can be depicted as follows



5.1 Observability for SISO Systems

Introductory examples: can the state $\mathbf{x}(t_0)$ be determined from $y(t)$?

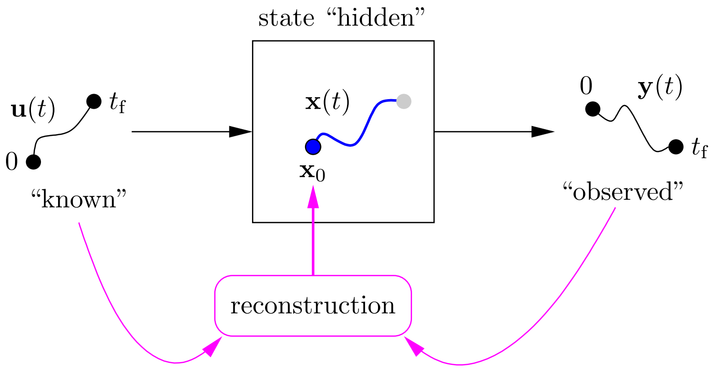


1. is not observable, as state variable x_2 is not connected to the output.
2. is observable if $\lambda_1 \neq \lambda_2$. Note that the output has to be observed for an interval of finite duration in order to discriminate the values x_1 and x_2.

Observability

A system is observable, if the initial state $\mathbf{x}_0 = \mathbf{x}(0)$ can be determined from the knowledge of the control input $u(t)$ and the output $y(t)$ over a finite time interval $[0, t_\mathrm{f}]$.

Illustration:



The time evolution of $y(t)$ can be computed as

$$y(t) = \underbrace{\mathbf{C}e^{\mathbf{A}t}\mathbf{x}_0}_{y_\mathrm{free}(t) \doteq} + \int_0^t \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{b}u(\tau)\,\mathrm{d}\tau \tag{5.1}$$

The homogeneous solution is defined as $y_\mathrm{free}(t)$. The second summand is the inhomogeneous part and represents the driven time evolution, which can be computed as function of the known input $u(t)$ as

$$y_\mathrm{free}(t) = y(t) - \int_0^t \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{b}u(\tau)\,\mathrm{d}\tau \tag{5.2}$$

Therefore, if the undriven system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) \tag{5.3}$$
$$y(t) = \mathbf{C}\mathbf{x}(t) \tag{5.4}$$

is observable, the same holds for the driven system. In other words: the system is observable, if state $\mathbf{x}_0$ can be reconstructed from $y_\mathrm{free}(t)$ by "inversion" of

$$y_\mathrm{free}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{x}_0 \tag{5.5}$$

Kalman Observability Criterion

Define

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{5.6}$$

The system $(\mathbf{A}, \mathbf{C})$ is observable, if $\mathcal{O}$ has full rank n.

Proof (sketch): Reconstruction of $\mathbf{x}_0$ from $y_\mathrm{free}(t_1), \ldots, y_\mathrm{free}(t_n)$ for $t_1, \ldots, t_n \in [0; t_\mathrm{f}]$.

$$\begin{bmatrix} y_\mathrm{free}(t_1) \\ \vdots \\ y_\mathrm{free}(t_n) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{C}e^{\mathbf{A}t_1} \\ \vdots \\ \mathbf{C}e^{\mathbf{A}t_n} \end{bmatrix}}_{\mathbf{M} \doteq} \mathbf{x}_0 \tag{5.7}$$

The state $\mathbf{x}_0$ can be computed as

$$\mathbf{x}_0 = \mathbf{M}^{-1} \begin{bmatrix} y_{\text{free}}(t_1) \\ \vdots \\ y_{\text{free}}(t_n) \end{bmatrix} \tag{5.8}$$

if $t_1, \ldots, t_n$ can be chosen such that $\mathbf{M}$ is invertible. A row of $\mathbf{M}$ reads

$$\begin{aligned} \mathbf{C}e^{\mathbf{A}t_i} &= \mathbf{C} + \mathbf{CA}t_i + \mathbf{C}\frac{\mathbf{A}^2}{2}t_i^2 + \mathbf{C}\frac{\mathbf{A}^3}{3!}t_i^3 + \cdots \\ &= \mathbf{C} + \alpha_{i,1}\mathbf{CA} + \alpha_{i,2}\mathbf{CA}^2 + \cdots + \alpha_{i,n}\mathbf{CA}^{n-1} \end{aligned} \tag{5.9}$$

The last line represents a linear combination of the vectors $\mathbf{C}, \mathbf{CA}, \ldots, \mathbf{CA}^{n-1}$ with the coefficients $\alpha_{i,j}$. The sum can be stopped at $(n-1)$ due to the theorem of Cayley-Hamilton. Now, $\mathbf{M}$ is invertible, if its rows are linearly independent. In order to get n linearly independent rows, the vectors $\mathbf{C}, \mathbf{CA}, \ldots, \mathbf{CA}^{n-1}$ have to be linearly independent (or $\mathcal{O}$ non-singular).

□

Example We consider observability of the introductory examples on page 37

$$\mathbf{A} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{5.10}$$

1.
$$\mathbf{C} = [1, 0] \qquad \hookrightarrow \qquad \mathcal{O} = \begin{bmatrix} 1 & 0 \\ -\lambda_1 & 0 \end{bmatrix} \tag{5.11}$$

   For the determinant follows $\det(\mathcal{O}) = 0$, hence system is not observable.

2.
$$\mathbf{C} = [1, 1] \qquad \hookrightarrow \qquad \mathcal{O} = \begin{bmatrix} 1 & 1 \\ \lambda_1 & \lambda_2 \end{bmatrix} \tag{5.12}$$

   Here, $\det(\mathcal{O}) = \lambda_2 - \lambda_1$, hence $\det(\mathcal{O}) \neq 0$ and system is observable for $\lambda_1 \neq \lambda_2$.

Reconstruction of States Based on the proof, we can summarize a recipe for reconstruction of the state for given $y(t_i)$ and $u(t)$, $0 \leq t \leq \max(t_i)$ for $i = 1, \ldots, n$.

1. Compute $y_{\text{free}}(t_i)$, compare (5.2)

$$y_{\text{free}}(t_i) = y(t_i) - \int_0^{t_i} \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{b}u(\tau)\,\mathrm{d}\tau \tag{5.13}$$

   for $i = 1, \ldots, n$

2. Reconstruct state $\mathbf{x}_0$, compare (5.7) and (5.8)

$$\mathbf{x}_0 = \begin{bmatrix} \mathbf{C}e^{\mathbf{A}t_1} \\ \vdots \\ \mathbf{C}e^{\mathbf{A}t_n} \end{bmatrix}^{-1} \begin{bmatrix} y_{\text{free}}(t_1) \\ \vdots \\ y_{\text{free}}(t_n) \end{bmatrix} \tag{5.14}$$

5.2 Extension to MIMO Systems

Having introduced observability for SISO system, we now sketch the criteria for MIMO systems.

Observability for MIMO systems

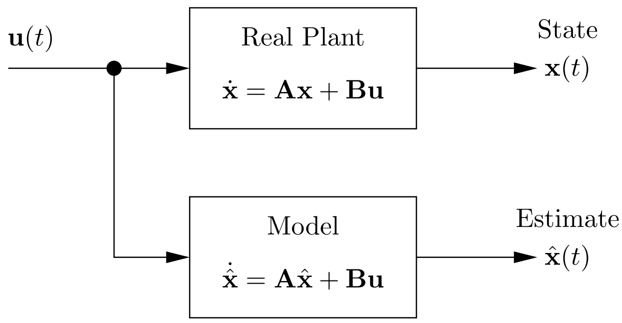
The observability matrix $\mathcal{O}$ is defined as

$$\mathcal{O} \doteq \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{5.15}$$

The system $(\mathbf{A}, \mathbf{C})$ is observable if $\mathrm{rank}(\mathcal{O}) = n$. (Note that $\mathcal{O}$ is a matrix of size $(nr) \times n$.)

5.3 Luenberger Observer

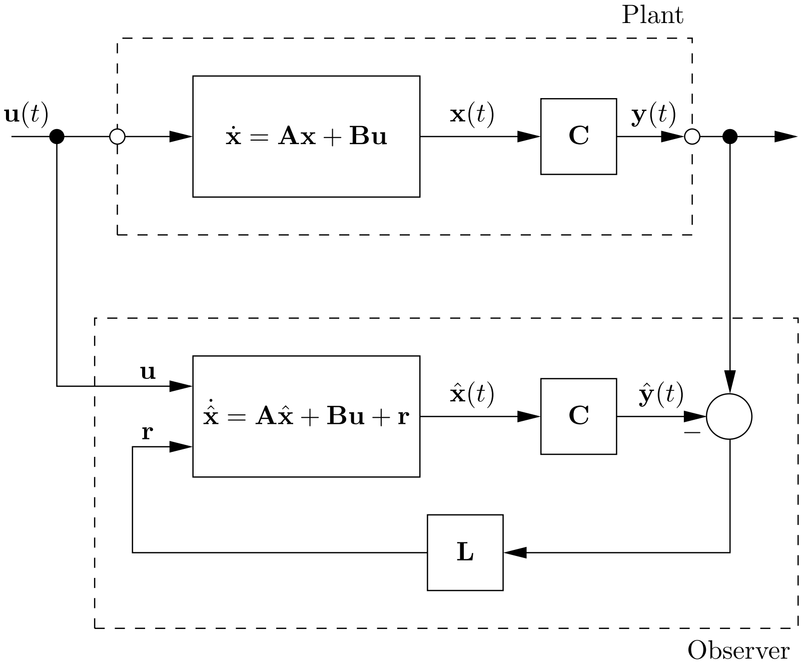
In principle state estimation could be accomplished by the following scheme



There are some prerequisites that $\hat{\mathbf{x}}(t)$ becomes a 'good' estimate for the state vector $\mathbf{x}(t)$.

- The system has to be *stable*.
- Absence of significant disturbances.
- Model should be accurate.

In order to obtain a better estimate or make the estimation feasible for unstable plants, a *feedback* is introduced. This leads to the **Luenberger Observer** depicted in the following



where $\mathbf{L}$ is the feedback matrix.

The ODE for the observer reads

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{r}(t) \tag{5.16}$$

Insertion of

$$\mathbf{r}(t) = \mathbf{L}(\mathbf{y}(t) - \hat{\mathbf{y}}(t)) = \mathbf{L}\mathbf{y}(t) - \mathbf{L}\mathbf{C}\hat{\mathbf{x}}(t) \tag{5.17}$$

yields

$$\dot{\hat{x}}(t) = (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\mathbf{y}(t) \tag{5.18}$$

Considering the ODE for the estimation error, defined by

$$\mathbf{e}(t) \doteq \mathbf{x}(t) - \hat{\mathbf{x}}(t) \tag{5.19}$$

gives with $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

$$\begin{aligned} \dot{\mathbf{e}}(t) = \dot{\mathbf{x}}(t) - \dot{\hat{\mathbf{x}}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) - (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}(t) - \mathbf{B}\mathbf{u}(t) - \mathbf{L}\mathbf{y}(t) \\ &= (\mathbf{A} - \mathbf{L}\mathbf{C})\,(\mathbf{x}(t) - \hat{\mathbf{x}}(t)) \end{aligned} \tag{5.20}$$

Hence the dynamics is described by the state equation

$$\dot{\mathbf{e}}(t) = (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{e}(t) \tag{5.21}$$

In order to obtain a reasonable estimate, we demand for the following

- The observer must be *stable*, i.e. $\mathbf{e}(t) \to 0$ for $t \to \infty$.
- As a consequence, the real parts of the eigenvalues of $(\mathbf{A} - \mathbf{L}\mathbf{C})$ must be negative $\mathrm{Re}(\lambda_i) < 0$ for $i = 1, \ldots, n$.
- The speed of the observer is determined by the position of the eigenvalues.

5.4 Detectability

Detectability stands to observability similarly to how stabilizability stands to controllability. Namely, detectability is a weaker notion than observability.

Detectability

The system $(\mathbf{A}, \mathbf{C})$ is detectable if there exist a matrix $\mathbf{L} \in \mathbb{R}^{n \times q}$ such that the matrix $\mathbf{A} - \mathbf{LC}$ is stable.

The idea of detectability is that all unstable modes of the system must be observable, such that all modes of the system $(\mathbf{A} - \mathbf{LC}, \mathbf{C})$ can be made stable. That is formalized in the following theorem

Observability and Detectability

If the system $(\mathbf{A}, \mathbf{C})$ is observable, then it is detectable.

Proof (without).

The converse is not true: as an example, a stable system with some unobservable modes is detectable (by choosing e.g. $\mathbf{K} = \mathbf{0}$) but not observable.

5.5 Observer Design

The problem could be recognized as similar to controller design

Controller	Observer
$\mathbf{A} - \mathbf{BK}$	$\mathbf{A} - \mathbf{LC}$

In order to apply the design principles of state feedback control of chapter 3, we apply a trick. As eigenvalues are the same for a matrix and its transpose, the *transposed system* could be considered, i.e. the following

$$\mathbf{A}^\top - \mathbf{C}^\top \mathbf{L}^\top \tag{5.22}$$

Hence, we obtain the following mapping

Controller	Observer
$\mathbf{A}$	$\mathbf{A}^\top$
$\mathbf{B}$	$\mathbf{C}^\top$
$\mathbf{K}$	$\mathbf{L}^\top$

(*) Pole placement for Observer Applying the pole placement of chapter 3.4 to the introduced substitutions results in

Pole Placement for Observer Canonical Form (SISO)

For a system given in observer canonical form

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 & -a_0 \\ 1 & \ddots & & & & -a_1 \\ & 1 & \ddots & & & \vdots \\ & & \ddots & \ddots & & \vdots \\ & & & \ddots & 0 & \vdots \\ & & & & 1 & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ \vdots \\ \vdots \\ b_{n-1} \end{bmatrix} u(t) \tag{5.23}$$

$$y(t) = [0, \ldots, 0, 1]\, \mathbf{x}(t) \tag{5.24}$$

The characteristic polynomial

$$p(\lambda) = \lambda^n + l_{n-1}\lambda^{n-1} + \cdots + l_1\lambda + l_0 \tag{5.25}$$

is implemented by the feedback

$$\mathbf{l} = \begin{bmatrix} (l_0 - a_0) \\ \vdots \\ (l_{n-1} - a_{n-1}) \end{bmatrix} \tag{5.26}$$

(*) Ackermann's Formula for Observer Recalling section 3.5 the controller reads

$$\mathbf{k}^\top = \mathbf{t}_1^\top p(\mathbf{A}) \tag{5.27}$$

where $\mathbf{t}_1^\top$ is the last row of the inverse controllability matrix $\mathcal{C}^{-1} = \left[\mathbf{B}, \mathbf{AB}, \ldots, \mathbf{A}^{n-1}\mathbf{B}\right]^{-1}$ As introduced in the previous section, we use $\mathbf{A}^\top$ for $\mathbf{A}$ and $(\mathbf{C})^\top = \mathbf{C}^\top$ for $\mathbf{b}$. The feedback gain is given by

$$\mathbf{l} = (\mathbf{k}^\top)^\top = (\mathbf{t}_1^\top p(\mathbf{A}))^\top = p(\mathbf{A}^\top)\mathbf{t}_1 \tag{5.28}$$

where $\mathbf{t}_1^\top$ is the last row of $\left[\mathbf{C}^\top, \mathbf{A}^\top\mathbf{C}^\top, \ldots, (\mathbf{A}^\top)^{n-1}\mathbf{C}^\top\right]^{-1}$. Therefore $\mathbf{t}_1$ is the last column of

$$\left(\left[\mathbf{C}^\top, \mathbf{A}^\top\mathbf{C}^\top, \ldots, (\mathbf{A}^\top)^{n-1}\mathbf{C}^\top\right]^\top\right)^{-1} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}^{-1} = \mathcal{O}^{-1} \tag{5.29}$$

In summary we get

Ackermann's Formula for Observer

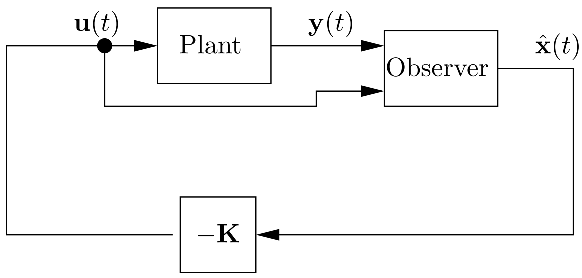
Given the characteristic polynomial $p(\lambda)$ for the observer, the feedback has to be chosen as $\mathbf{l} = p(\mathbf{A}^\top)\mathbf{t}_1$ where $\mathbf{t}_1$ is the last column of the inverse observability matrix

$$\mathcal{O}^{-1} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix}^{-1}$$

Note that the system has to be *observable* in order to compute the inverse of $\mathcal{O}$.

5.6 Control Loop with Observer

In this section, a state feedback of the *estimated* state vector will be considered as follows



State Feedback
Controller

Hence the feedback is given by

$$\mathbf{u}(t) = -\mathbf{K}\hat{\mathbf{x}}(t) \tag{5.30}$$

Combining the plant system

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{5.31}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \tag{5.32}$$

and the observer state equation (5.18)

$$\dot{\hat{\mathbf{x}}}(t) = (\mathbf{A} - \mathbf{LC})\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\mathbf{y}(t) \tag{5.33}$$

into a set of ODE for the combined system yields

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\hat{\mathbf{x}}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{BK} \\ \mathbf{LC} & (\mathbf{A} - \mathbf{LC} - \mathbf{BK}) \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \hat{\mathbf{x}}(t) \end{bmatrix} \tag{5.34}$$

The eigenvalues of the combined system are calculated as follows

$$\begin{aligned} 0 = p(\lambda) &= \det\left(\begin{bmatrix} (\lambda\mathbf{I} - \mathbf{A}) & \mathbf{BK} \\ -\mathbf{LC} & (\lambda\mathbf{I} - \mathbf{A} + \mathbf{LC} + \mathbf{BK}) \end{bmatrix}\right) \\ &= \det\left(\begin{bmatrix} (\lambda\mathbf{I} - \mathbf{A} + \mathbf{BK}) & \mathbf{BK} \\ (\lambda\mathbf{I} - \mathbf{A} + \mathbf{BK}) & (\lambda\mathbf{I} - \mathbf{A} + \mathbf{LC} + \mathbf{BK}) \end{bmatrix}\right) \\ &= \det\left(\begin{bmatrix} (\lambda\mathbf{I} - \mathbf{A} + \mathbf{BK}) & \mathbf{BK} \\ \mathbf{0} & (\lambda\mathbf{I} - \mathbf{A} + \mathbf{LC}) \end{bmatrix}\right) \\ &= \underbrace{\det(\lambda\mathbf{I} - (\mathbf{A} - \mathbf{BK}))}_{\text{closed loop}} \; \underbrace{\det(\lambda\mathbf{I} - (\mathbf{A} - \mathbf{LC}))}_{\text{observer}} \end{aligned} \tag{5.35}$$

For the manipulations above we made use of the linear algebra lemma that the determinant does not change when adding columns $n+1, \dots, 2n$ to columns $1, \dots, n$ from first to second line and when subtracting rows $1, \dots, n$ from rows $n+1, \dots, 2n$ from second to third line. The last equality utilized the lemma for computing the determinant of a block matrix. As result we could state that the eigenvalues of the state feedback control loop are not changed by the observer design, this is called **separation theorem**. Based on this, the state feedback design can be carried out independently from the observer.

On the choice of eigenvalues for the observer, the following could be stated

- The eigenvalues should be placed to the left of the closed loop eigenvalues, otherwise the reaction of the system to disturbances, which cause differences between the state of the plant and the estimate, would be too slow.
- Theoretically, the observer could be made arbitrarily fast. As the algorithm involves differentiation, this is critical w.r.t. noise in measurements. Hence, the observer should be made faster than the state feedback, but not significantly faster.

5.7 Relation to Kalman Filter

The stationary Kalman filter gives the state observer that is optimal with respect to the performance index

$$J = \frac{1}{2}\int_{-\infty}^{0} \mathbf{v}^\top(t)\mathbf{R}^{-1}\mathbf{v}(t) + \mathbf{w}^\top(t)\mathbf{Q}^{-1}\mathbf{w}(t)\,\mathrm{d}t$$

where $\mathbf{w}(t)$ is the process noise and $\mathbf{v}(t)$ is the measurement noise, that affects the system as

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

In a statistical setting, $\mathbf{w}$ and $\mathbf{v}$ are assumed to be uncorrelated white Gaussian noises with zero mean. The matrices $\mathbf{Q}$ and $\mathbf{R}$ can be interpreted as a form of pointwise covariance matrices of the process and measurement noises, respectively.

The expression for the Kalman filter derived using statistical approaches can be *formally* derived from the expression for the LQR controller by means of the substitutions $\mathbf{A} \to \mathbf{A}^\top$ and $\mathbf{B} \to \mathbf{C}^\top$. Using these substitutions, the gain for the Kalman estimator is computed as $\mathbf{L} = \mathbf{K}^\top$ with (4.31)

$$\mathbf{L} = \mathbf{P}\mathbf{C}^\top\mathbf{R}^{-1}$$

where the matrix $\mathbf{P}$ is the solution of the algebraic Riccati equation (ARE)

$$\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^\top - \mathbf{P}\mathbf{C}^\top\mathbf{R}^{-1}\mathbf{C}\mathbf{P} + \mathbf{Q} = 0$$

that is obtained from (4.35) with the above substitutions.

Chapter 6

Discrete Time Systems

This chapter deals with linear time invariant systems in discrete time.

The continuous time system is discretized at equidistant sampling instants, with sampling time T_s (that is, the time between two consecutive sampling instants). The value of the system matrices and vectors at the sampling instant k are denoted using the index $_k$, where the sampling starts at time 0. As an example, the value of the vector $\mathbf{x}(t)$ at the k-th sampling instant is the value of the vector at time $t = k\mathrm{T_s}$,

$$\mathbf{x}_k = \mathbf{x}(k\mathrm{T}_s) \tag{6.1}$$

The input vector is constant in between sampling instants, i.e. a piecewise constant parametrization $\mathbf{u}$ is employed. Note that other parametrizations are possible (e.g. piecewise linear or more generally polynomial).

6.1 Discrete Time LTI Systems

In this chapter, the general LTI system in state space and in continuous time is represented as

$$\dot{\mathbf{x}}(t) = \mathbf{A}_c\mathbf{x}(t) + \mathbf{B}_c\mathbf{u}(t) \tag{6.2}$$

$$\mathbf{y}(t) = \mathbf{C}_c\mathbf{x}(t) + \mathbf{D}_c\mathbf{u}(t) \tag{6.3}$$

where the index $_c$ denotes continuous time.

Generally speaking, the state space representation in discrete time can be derived from the state space representation in continuous time by means of simulation (that is, integration over time). In case of LTI systems, it is possible to derive an analytic expression for the state space system in discrete time, without using any numerical integration.

In discrete time, the general LTI system in state space $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ can be written as

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \tag{6.4}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k \tag{6.5}$$

Thanks to the time-invariance property, the system representation is the same at all times. In particular, we can consider the time $t_0 = 0$ and $t = \mathrm{T}_s$ in equation (1.32), obtaining

$$\begin{aligned}\mathbf{x}_1 = \mathbf{x}(\mathrm{T}_s) &= e^{\mathbf{A}_c(\mathrm{T}_s-0)}\mathbf{x}(0) + \int_0^{\mathrm{T}_s} e^{\mathbf{A}_c(\mathrm{T}_s-\tau)}\mathbf{B}_c\mathbf{u}(\tau)\mathrm{d}\tau \\ &= e^{\mathbf{A}_c\mathrm{T}_s}\mathbf{x}(0) + \int_0^{\mathrm{T}_s} e^{\mathbf{A}_c t}\mathrm{d}t\mathbf{B}_c\mathbf{u}(0) = \mathbf{A}\mathbf{x}_0 + \mathbf{B}\mathbf{u}_0\end{aligned} \tag{6.6}$$

where the fact that $\mathbf{u}(t)$ is piecewise constant in between sampling instants has been exploited to move $\mathbf{u}$ outside the integral, and the change of variable $t = \mathrm{T}_s - \tau$ is performed in the integration. The equation (6.5) is simply obtained evaluating equation (6.3) at the time $t = k\mathrm{T}_s$.

In summary, the expression of the matrices in the state space representation in discrete time $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is

$$\mathbf{A} = e^{\mathbf{A}_c \mathrm{T_s}} \tag{6.7}$$

$$\mathbf{B} = \int_0^{\mathrm{T}_s} e^{\mathbf{A}_c t} \mathrm{d}t \mathbf{B}_c \tag{6.8}$$

$$\mathbf{C} = \mathbf{C}_c \tag{6.9}$$

$$\mathbf{D} = \mathbf{D}_c \tag{6.10}$$

6.1.1 Homogeneous Response

The homogeneous response with zero input and initial state $\mathbf{x}_0$ can be found by successive substitutions

$$\begin{aligned}
\mathbf{x}_1 &= \mathbf{A}\mathbf{x}_0 \\
\mathbf{x}_2 &= \mathbf{A}\mathbf{x}_1 = \mathbf{A}^2\mathbf{x}_0 \\
\cdots &= \cdots \\
\mathbf{x}_k &= \mathbf{A}\mathbf{x}_{k-1} = \mathbf{A}^k\mathbf{x}_0
\end{aligned}$$

Note that it is computed using only the matrix $\mathbf{A}$.

6.1.2 Forced Response

The forced response with generic non-zero input is computed by induction. The expression for two consecutive substitutions

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{u}_{k+1} \\
&= \mathbf{A}(\mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k) + \mathbf{B}\mathbf{u}_{k+1} \\
&= \mathbf{A}^2\mathbf{x}_k + \mathbf{A}\mathbf{B}\mathbf{u}_k + \mathbf{B}\mathbf{u}_{k+1}
\end{aligned}$$

can be generalized as

$$\mathbf{x}_k = \mathbf{A}^k\mathbf{x}_0 + \sum_{m=0}^{k-1} \mathbf{A}^{k-m-1}\mathbf{B}\mathbf{u}_m \tag{6.11}$$

for $k \geq 0$.

6.1.3 System output response

The output response is computed by substitution of (6.11) into the equation $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k$, obtaining

$$\mathbf{y}_k = \mathbf{C}\mathbf{A}^k\mathbf{x}_0 + \sum_{m=0}^{k-1} \mathbf{C}\mathbf{A}^{k-m-1}\mathbf{B}\mathbf{u}_m + \mathbf{D}\mathbf{u}_k$$

6.2 Stability in Discrete Time

The eigenvalues of the matrix $\mathbf{A}$ are defined as

$$\lambda_i \mathbf{v}_i = \mathbf{A}\mathbf{v}_i \quad \text{for } \mathbf{v}_i \neq 0 \tag{6.12}$$

The corresponding vectors $\mathbf{v}_i$ are the eigenvectors. Equation (6.12) can be rewritten as

$$(\lambda_i \mathbf{I} - \mathbf{A})\mathbf{v}_i = 0$$

and together with the condition $\mathbf{v}_i \neq 0$ implies

$$\det(\lambda_i \mathbf{I} - \mathbf{A}) = 0$$

which defines the characteristic polynomial of the matrix $\mathbf{A}$.

If $\mathbf{A}$ has size $n \times n$, the characteristic polynomial

$$\lambda^n + a_{n-1}\lambda^{n-1} + \cdots + a_1\lambda + a_0 = 0$$

can be factorized as

$$(\lambda - \lambda_1)(\lambda - \lambda_2)\cdots(\lambda - \lambda_{n-1}) = 0$$

with $\lambda_i \in \mathbb{C}$. Defined

$$\mathbf{V} = \begin{bmatrix}\mathbf{v}_1 \mathbf{v}_1 \dots \mathbf{v}_n\end{bmatrix}$$

and

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & \\ 0 & \dots & & \lambda_{n-1} \end{bmatrix}$$

then

$$\Lambda^k = \begin{bmatrix} \lambda_1^k & 0 & \dots & 0 \\ 0 & \lambda_2^k & & \vdots \\ \vdots & & \ddots & \\ 0 & \dots & & \lambda_n^k \end{bmatrix}$$

and

$$\mathbf{A}^k = (\mathbf{V}\Lambda\mathbf{V}^{-1})^k = \mathbf{V}\Lambda^k\mathbf{V}^{-1} \tag{6.13}$$

The asymptotic stability of the system is defined in terms of homogeneous response. Given any initial state $\mathbf{x}_0$, the system is said to be asymptotically stable if the homogeneous response $\mathbf{x}_k$ converges to 0 as time $k \to \infty$,

$$\lim_{k\to\infty} \mathbf{x}_k = \lim_{k\to\infty} \mathbf{A}^k \mathbf{x}_0 = \lim_{k\to\infty} \mathbf{V}\Lambda^k\mathbf{V}^{-1}\mathbf{x}_0 = 0$$

for any $\mathbf{x}_0$. Equation (6.13) shows that all elements of $\mathbf{A}^k$ are a linear combination of the system modes λ_i^k, and therefore stability depends on all components decaying to zero with time.

Asymptotic stability

A linear discrete time system is asymptotically stable if and only if all eigenvalues have magnitude smaller than one, i.e. if they are strictly inside the unit circle in the complex plane.

The system is not asymptotically stable in the following cases:

- If $|\lambda_i| > 1$ for one real eigenvalue or a couple of complex-conjugate eigenvalues, the mode grows exponentially. The system is said to be unstable.
- If $|\lambda_i| = 1$ for one single real eigenvalue or a couple of complex-conjugate eigenvalues, while all other eigenvalues have module smaller than one, the system response neither decays or grows. The system is said to be marginally stable.

BIBO stability

If a linear discrete time system is asymptotically stable, then it is BIBO stable, i.e., a bounded input gives a bounded output for every initial value.

6.3 Discrete Time Linear Quadratic Regulator

In this section, we derive the LQR regulator for LTI systems in discrete time, considering both infinite and finite control horizons. In the infinite horizon case, the optimal input is a state feedback with constant gain matrix $\mathbf{K}$, while in the finite horizon case it is a state feedback with time-varying gain matrix $\mathbf{K}_k$

6.3.1 Infinite horizon

In discrete time, the performance index for the infinite horizon LQR controller reads

$$J_0^\infty\{\mathbf{x},\mathbf{u}\} = \sum_{k=0}^{\infty} J_k = \sum_{k=0}^{\infty} \frac{1}{2}\mathbf{x}_k^\top \mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^\top \mathbf{S}\mathbf{x}_k + \frac{1}{2}\mathbf{u}_k^\top \mathbf{R}\mathbf{u}_k$$

where $\mathbf{u} = [\mathbf{u}_0, \mathbf{u}_1, \dots]$, $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, \dots]$, $\mathbf{Q}$ is a symmetric positive semi-definite $n \times n$ matrix, $\mathbf{S}$ is a $p \times n$ matrix and $\mathbf{R}$ is a symmetric positive definite $p \times p$ matrix. If the system $(\mathbf{A}, \mathbf{B})$ is controllable, then there exist an input sequence $\{\mathbf{u}\}$ such that the index has finite value. In fact, if the system is controllable, it can be steered to zero in a finite number of steps T, and by choosing $\mathbf{u}_k = 0$ for $k \geq T$ all cost terms are zero for all stages $k \geq T$.

The aim is to compute the optimal input sequence $\mathbf{u}$ that minimizes the performance index J_0^∞. At this stage, no assumptions are made on the structure of $\mathbf{u}$.

Showing the components at the first stage $k = 0$ and at the generic stage $k = n$ of the performance index, we get

$$J_0^\infty\{\mathbf{x},\mathbf{u}\} = \frac{1}{2}\mathbf{x}_0^\top \mathbf{Q}\mathbf{x}_0 + \mathbf{u}_0^\top \mathbf{S}\mathbf{x}_0 + \frac{1}{2}\mathbf{u}_0^\top \mathbf{R}\mathbf{u}_0 + \dots + \frac{1}{2}\mathbf{x}_n^\top \mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top \mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top \mathbf{R}\mathbf{u}_n + \dots$$

The value of the index does not change by adding

$$\begin{aligned}0 = \frac{1}{2}\mathbf{x}_0^\top \mathbf{P}\mathbf{x}_0 - \frac{1}{2}\mathbf{x}_0^\top \mathbf{P}\mathbf{x}_0 + \frac{1}{2}\mathbf{x}_1^\top \mathbf{P}\mathbf{x}_1 - \frac{1}{2}\mathbf{x}_1^\top \mathbf{P}\mathbf{x}_1 + \dots +& \\ \dots + \frac{1}{2}\mathbf{x}_n^\top \mathbf{P}\mathbf{x}_n - \frac{1}{2}\mathbf{x}_n^\top \mathbf{P}\mathbf{x}_n + \mathbf{x}_{n+1}^\top \mathbf{P}\mathbf{x}_{n+1} - \mathbf{x}_{n+1}^\top \mathbf{P}\mathbf{x}_{n+1} + \dots&\end{aligned}$$

where $\mathbf{P}$ is any positive semidefinite matrix, obtaining

$$\begin{aligned}J_0^\infty\{\mathbf{x},\mathbf{u}\} = \frac{1}{2}\mathbf{x}_0^\top \mathbf{P}\mathbf{x}_0 + \left(-\frac{1}{2}\mathbf{x}_0^\top \mathbf{P}\mathbf{x}_0 + \frac{1}{2}\mathbf{x}_0^\top \mathbf{Q}\mathbf{x}_0 + \mathbf{u}_0^\top \mathbf{S}\mathbf{x}_0 + \frac{1}{2}\mathbf{u}_0^\top \mathbf{R}\mathbf{u}_0 + \frac{1}{2}\mathbf{x}_1^\top \mathbf{P}\mathbf{x}_1\right) + \dots \\ \dots + \left(-\frac{1}{2}\mathbf{x}_n^\top \mathbf{P}\mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top \mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top \mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top \mathbf{R}\mathbf{u}_n + \frac{1}{2}\mathbf{x}_{n+1}^\top \mathbf{P}\mathbf{x}_{n+1}\right) + \dots \quad (6.14)\end{aligned}$$

By using the dynamic equation $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$, the expression at the generic stage $k = n$ is

$$\begin{aligned}J_n &= \frac{1}{2}\mathbf{x}_n^\top \mathbf{P}\mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top \mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top \mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top \mathbf{R}\mathbf{u}_n + \frac{1}{2}(\mathbf{A}\mathbf{x}_n + \mathbf{B}\mathbf{u}_n)^\top \mathbf{P}(\mathbf{A}\mathbf{x}_n + \mathbf{B}\mathbf{u}_n) \\ &= \frac{1}{2}\mathbf{u}_n^\top (\mathbf{R} + \mathbf{B}^\top \mathbf{P}\mathbf{B})\mathbf{u}_n + \mathbf{u}_n^\top (\mathbf{S} + \mathbf{B}^\top \mathbf{P}\mathbf{B})\mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top (-\mathbf{P} + \mathbf{Q} + \mathbf{A}^\top \mathbf{P}\mathbf{A})\mathbf{x}_n\end{aligned}$$

that is a quadratic function of $\mathbf{u}_n$ with positive definite Hessian matrix $(\mathbf{R}+\mathbf{B}^\top \mathbf{P}\mathbf{B})$. Furthermore, note that the quadratic function has the same expression at all stages.

The unique minimizer $\mathbf{u}_n^*$ of the convex quadratic function J_n can be obtained by setting the gradient w.r.t. $\mathbf{u}_n$ to zero

$$\nabla J_n = (\mathbf{R} + \mathbf{B}^\top \mathbf{P}\mathbf{B})\mathbf{u}_n + (\mathbf{S} + \mathbf{B}^\top \mathbf{P}\mathbf{A})\mathbf{x}_n = 0$$

obtaining

$$\mathbf{u}_n = -(\mathbf{R} + \mathbf{B}^\top \mathbf{P}\mathbf{B})^{-1}(\mathbf{S} + \mathbf{B}^\top \mathbf{P}\mathbf{A})\mathbf{x}_n = -\mathbf{K}\mathbf{x}_n$$

that computes the optimal input as a state feedback with constant gain matrix $\mathbf{K}$.

Using this expression for $\mathbf{u}_k$, the performance index at stage $k = n$ is

$$J_n = \frac{1}{2}\mathbf{x}_n^\top \left(-\mathbf{P} + \mathbf{Q} + \mathbf{A}^\top\mathbf{P}\mathbf{A} - (\mathbf{S}^\top + \mathbf{A}^\top\mathbf{P}\mathbf{B})(\mathbf{R} + \mathbf{B}^\top\mathbf{P}\mathbf{B})^{-1}(\mathbf{S} + \mathbf{B}^\top\mathbf{P}\mathbf{A})\right)\mathbf{x}_n$$

that by choosing the positive definite matrix $\mathbf{P}$ as

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}^\top\mathbf{P}\mathbf{A} - (\mathbf{S}^\top + \mathbf{A}^\top\mathbf{P}\mathbf{B})(\mathbf{R} + \mathbf{B}^\top\mathbf{P}\mathbf{B})^{-1}(\mathbf{S} + \mathbf{B}^\top\mathbf{P}\mathbf{A}) \tag{6.15}$$

sets to zero the performance index at stage $k = n$

$$J_n = \frac{1}{2}\mathbf{x}_n^\top\,(0)\,\mathbf{x}_n = 0$$

Equation (6.15) is the discrete time algebraic Riccati equation (DARE).

Therefore, the performance index expression in (6.14) reduces to

$$V_0^\infty\{\mathbf{x}_0\} = \min_{\mathbf{x},\mathbf{u}} J_0^\infty\{\mathbf{x},\mathbf{u}\} = \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}\mathbf{x}_0$$

that gives the optimal value of the performance index as a function of the initial state $\mathbf{x}_0$.

An important property of the infinite horizon LQR is that if $(\mathbf{A}, \mathbf{Q}^{\frac{1}{2}})$ is observable then $\mathbf{A}-\mathbf{B}\mathbf{K}$ is stable, i.e. the optimal state feedback is a stabilizing control.

6.3.2 Finite horizon

In discrete time, the performance index for the finite horizon LQR controller reads

$$J_0^N\{\mathbf{x},\mathbf{u}\} = \sum_{k=0}^{N} \frac{1}{2}\mathbf{x}_k^\top\mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^\top\mathbf{S}\mathbf{x}_k + \frac{1}{2}\mathbf{u}_k^\top\mathbf{R}\mathbf{u}_k + \frac{1}{2}\mathbf{x}_N^\top\mathbf{Q}_N\mathbf{x}_N$$

where $\mathbf{Q}$ and $\mathbf{Q}_N$ are symmetric positive semi-definite $n \times n$ matrices, $\mathbf{S}$ is a $p \times n$ matrix and $\mathbf{R}$ is a symmetric positive definite $p \times p$ matrix.

The aim is to compute the optimal input sequence $\mathbf{u}$ that minimizes the performance index J_0^N. At this stage, no assumptions are made on the structure of $\mathbf{u}$.

Showing the components at the first stage $k = 0$ and at the generic stage $k = n$ of the performance index, we get

$$J_0^N\{\mathbf{x},\mathbf{u}\} = \frac{1}{2}\mathbf{x}_0^\top\mathbf{Q}\mathbf{x}_0 + \mathbf{u}_0^\top\mathbf{S}\mathbf{x}_0 + \frac{1}{2}\mathbf{u}_0^\top\mathbf{R}\mathbf{u}_0 + \cdots + \frac{1}{2}\mathbf{x}_n^\top\mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top\mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top\mathbf{R}\mathbf{u}_n + \frac{1}{2}\mathbf{x}_N^\top\mathbf{Q}_N\mathbf{x}_N$$

The value of the index does not change by adding

$$\begin{aligned}
0 = {}& \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}_1\mathbf{x}_0 - \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}_1\mathbf{x}_0 + \frac{1}{2}\mathbf{x}_1^\top\mathbf{P}_2\mathbf{x}_1 - \frac{1}{2}\mathbf{x}_1^\top\mathbf{P}_2\mathbf{x}_1 + \cdots + \\
& \cdots + \frac{1}{2}\mathbf{x}_n^\top\mathbf{P}_{n+1}\mathbf{x}_n - \frac{1}{2}\mathbf{x}_n^\top\mathbf{P}_{n+1}\mathbf{x}_n + \mathbf{x}_{n+1}^\top\mathbf{P}\mathbf{x}_{n+1} - \mathbf{x}_{n+1}^\top\mathbf{P}\mathbf{x}_{n+1} + \cdots + \mathbf{x}_N^\top\mathbf{P}_N\mathbf{x}_N - \mathbf{x}_N^\top\mathbf{P}_N\mathbf{x}_N
\end{aligned}$$

where $\mathbf{P}_k$ is any sequence of positive semidefinite matrices, obtaining

$$\begin{aligned}
J_0^N\{\mathbf{x},\mathbf{u}\} = {}& \frac{1}{2}\mathbf{x}_0^\top\mathbf{P}_0\mathbf{x}_0 + \left(-\frac{1}{2}\mathbf{x}_0^\top\mathbf{P}_0\mathbf{x}_0 + \frac{1}{2}\mathbf{x}_0^\top\mathbf{Q}\mathbf{x}_0 + \mathbf{u}_0^\top\mathbf{S}\mathbf{x}_0 + \frac{1}{2}\mathbf{u}_0^\top\mathbf{R}\mathbf{u}_0 + \frac{1}{2}\mathbf{x}_1^\top\mathbf{P}_1\mathbf{x}_1\right) + \ldots \\
& \cdots + \left(-\frac{1}{2}\mathbf{x}_n^\top\mathbf{P}_n\mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top\mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top\mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top\mathbf{R}\mathbf{u}_n + \frac{1}{2}\mathbf{x}_{n+1}^\top\mathbf{P}_{n+1}\mathbf{x}_{n+1}\right) + \ldots \\
& \cdots + \left(-\frac{1}{2}\mathbf{x}_N^\top\mathbf{P}_N\mathbf{x}_N + \frac{1}{2}\mathbf{x}_N^\top\mathbf{Q}_N\mathbf{x}_N\right)
\end{aligned} \tag{6.16}$$

The performance index at the last stage j_N is equal to zero by choosing $\mathbf{P}_N = \mathbf{Q}_N$. By using the dynamic equation $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u} + k$, the expression at the generic stage $k = n$ is

$$\begin{aligned} J_n &= \frac{1}{2}\mathbf{x}_n^\top \mathbf{P}_n \mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top \mathbf{Q}\mathbf{x}_n + \mathbf{u}_n^\top \mathbf{S}\mathbf{x}_n + \frac{1}{2}\mathbf{u}_n^\top \mathbf{R}\mathbf{u}_n + \frac{1}{2}(\mathbf{A}\mathbf{x}_n + \mathbf{B}\mathbf{u}_n)^\top \mathbf{P}_{n+1}(\mathbf{A}\mathbf{x}_n + \mathbf{B}\mathbf{u}_n) \\ &= \frac{1}{2}\mathbf{u}_n^\top (\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B})\mathbf{u}_n + \mathbf{u}_n^\top (\mathbf{S} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B})\mathbf{x}_n + \frac{1}{2}\mathbf{x}_n^\top (-\mathbf{P}_n + \mathbf{Q} + \mathbf{A}^\top \mathbf{P}_{n+1}\mathbf{A})\mathbf{x}_n \end{aligned}$$

that is a quadratic function of $\mathbf{u}_n$ with positive definite Hessian matrix $(\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B})$.

By choosing the matrix $\mathbf{P}_n$ as

$$\mathbf{P}_n = \mathbf{Q}_n + \mathbf{A}^\top \mathbf{P}_{n+1}\mathbf{A} - (\mathbf{S}^\top + \mathbf{A}^\top \mathbf{P}_{n+1}\mathbf{B})(\mathbf{R} + \mathbf{B}^\top \mathbf{P}\mathbf{B})^{-1}(\mathbf{S} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B}) \tag{6.17}$$

that is called Riccati recursion, the performance index at stage $k = n$ can be written as

$$J_n = \frac{1}{2}\begin{bmatrix}\mathbf{u}_n^\top & \mathbf{x}_n^\top\end{bmatrix}\begin{bmatrix}\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B} \\ \mathbf{S}^\top + \mathbf{A}^\top \mathbf{P}_{n+1}\mathbf{B}\end{bmatrix}\left(\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B}\right)^{-1}\begin{bmatrix}\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B} & \mathbf{S} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{A}\end{bmatrix}\begin{bmatrix}\mathbf{u}_n \\ \mathbf{x}_n\end{bmatrix}$$

showing that j_n is a convex quadratic function (since the matrix $(\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B})^{-1}$ is positive definite, being the inverse of a positive definite matrix), and its minimum $j_n = 0$ is obtained for

$$\begin{bmatrix}\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B} & \mathbf{S} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{A}\end{bmatrix}\begin{bmatrix}\mathbf{u}_n \\ \mathbf{x}_n\end{bmatrix} = 0$$

giving

$$\mathbf{u}_n = -(\mathbf{R} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{B})^{-1}(\mathbf{S} + \mathbf{B}^\top \mathbf{P}_{n+1}\mathbf{A})\mathbf{x}_n = -\mathbf{K}_n\mathbf{x}_n$$

that computes the optimal input as state feedback with time-varying gain matrix $\mathbf{K}_n$.

Therefore, the performance index expression in (6.16) reduces to

$$V_0^N\{\mathbf{x}_0\} = \min_{\mathbf{x},\mathbf{u}} J_0^N\{\mathbf{x}, \mathbf{u}\} = \frac{1}{2}\mathbf{x}_0^\top \mathbf{P}_0\mathbf{x}_0$$

that gives the optimal value of the performance index as a function of the initial state $\mathbf{x}_0$.

Note that by choosing $\mathbf{Q}_N = \mathbf{P}$, where $\mathbf{P}$ is the solution of the DARE, then the Riccati recursion (6.17) becomes the DARE (6.15).

6.4 (*) Discrete Time Observer

In discrete time, we define the state estimate at time k computed using output measurements up to time k as $\hat{\mathbf{x}}_{k|k}$, and the one-step-ahead state predictor at time $k+1$ computed using output measurements up to time k as $\hat{\mathbf{x}}_{k+1|k}$.

The one-step-ahead state predictor is simply computed by forward simulation of the estimator, as

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}\hat{\mathbf{x}}_{k|k} + \mathbf{B}\mathbf{u}_k \tag{6.18}$$

If we define the output error $\mathbf{e}_k$ at time k as

$$\mathbf{e}_k = \mathbf{y}_k - (\mathbf{C}\hat{\mathbf{x}}_{k|k-1} + \mathbf{D}\mathbf{u}_k)$$

the state estimator can be computed by correcting the one-step-ahead state predictor using the information in the new output error

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{L}_e\mathbf{e}_{k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{L}_e(\mathbf{y}_{k+1} - (\mathbf{C}\hat{\mathbf{x}}_{k+1|k} + \mathbf{D}\mathbf{u}_{k+1})) \tag{6.19}$$

where $\mathbf{L}_e$ is the gain for the state estimator.

Insertion of equation (6.19) into equation (6.18) gives an expression to compute the one-step-ahead state predictor at time $k+1$ as a function of the one-step-ahead state predictor at time k and the new output measurement at time $k+1$

$$\begin{aligned}
\hat{\mathbf{x}}_{k+1|k} &= \mathbf{A}(\hat{\mathbf{x}}_{k|k-1} + \mathbf{L}_e\mathbf{e}_k) + \mathbf{B}\mathbf{u}_k \\
&= \mathbf{A}(\hat{\mathbf{x}}_{k|k-1} + \mathbf{L}_e(\mathbf{y}_k - (\mathbf{C}\hat{\mathbf{x}}_{k|k-1} + \mathbf{D}\mathbf{u}_k)) + \mathbf{B}\mathbf{u}_k \\
&= \mathbf{A}\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{A}\mathbf{L}_e(\mathbf{y}_k - (\mathbf{C}\hat{\mathbf{x}}_{k|k-1} + \mathbf{D}\mathbf{u}_k) \\
&= \mathbf{A}\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{L}(\mathbf{y}_k - (\mathbf{C}\hat{\mathbf{x}}_{k|k-1} + \mathbf{D}\mathbf{u}_k)
\end{aligned}$$

where we defined the gain for the one-step-ahead state predictor L as

$$\mathbf{L} = \mathbf{A}\mathbf{L}_e$$

The error in the one-step-ahead state prediction has the dynamic

$$\begin{aligned}
\Delta\mathbf{x}_{k+1|k} &= \hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1} \\
&= \mathbf{A}\hat{\mathbf{x}}_{k|k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{L}(\mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k - (\mathbf{C}\hat{\mathbf{x}}_{k|k-1}\mathbf{D}\mathbf{u}_k)) - (\mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k) \\
&= (\mathbf{A} - \mathbf{L}\mathbf{C})\Delta\mathbf{x}_{k|k-1}
\end{aligned}$$

and therefore it converges to zero if the matrix $\mathbf{A} - \mathbf{L}\mathbf{C}$ is stable.

6.5 (*) Discrete Time Kalman Filter

The stationary Kalman filter gives the state observer that is optimal with respect to the performance index

$$J = \sum_{k=0}^{\infty} \frac{1}{2}\mathbf{v}_k^\top \mathbf{R}^{-1}\mathbf{v}_k + \frac{1}{2}\mathbf{w}_k^\top \mathbf{Q}^{-1}\mathbf{w}_k$$

where $\mathbf{w}_k$ is the process noise and $\mathbf{v}_k$ is the measurement noise, that affect the system as

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{w}_k \\
\mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{v}_k
\end{aligned}$$

In a statistical setting, $\mathbf{w}$ and $\mathbf{v}$ are assumed to be uncorrelated Gaussian noises. The matrices $\mathbf{Q}$ and $\mathbf{R}$ are interpreted as the covariance of the process and measurement noises, respectively.

The expression for the Kalman filter derived using statistical approaches can be *formally* derived from the expression for the LQR controller by means of the substitutions $\mathbf{A} \to \mathbf{A}^\top$ and $\mathbf{B} \to \mathbf{C}^\top$. Using these substitutions, the gain for the Kalman estimator is

$$\mathbf{L}_e = \mathbf{P}\mathbf{C}^\top(\mathbf{R} + \mathbf{C}\mathbf{P}\mathbf{C}^\top)^{-1}$$

where the matrix $\mathbf{P}$ is the solution of the DARE

$$\mathbf{P} = \mathbf{Q} + \mathbf{A}\mathbf{P}\mathbf{A}^\top - \mathbf{A}\mathbf{P}\mathbf{C}^\top(\mathbf{R} + \mathbf{C}\mathbf{P}\mathbf{C}^\top)^{-1}\mathbf{C}\mathbf{P}\mathbf{A}^\top$$

Note that simply plugging the above substitutions in the expression of the LQR gain does not give the expression for the gain of the Kalman estimator, but instead the expression for the gain of the Kalman one-step-ahead predictor

$$\mathbf{L} = \mathbf{A}\mathbf{L}_e = \mathbf{A}\mathbf{P}\mathbf{C}^\top(\mathbf{R} + \mathbf{C}\mathbf{P}\mathbf{C}^\top)^{-1}$$

that is used to compute the dynamic $\mathbf{A} - \mathbf{L}\mathbf{C}$ of the one-step-ahead prediction error.

Chapter 7

Introduction to Model Predictive Control

This chapter contains a brief introduction to Model Predictive Control (MPC) for linear discrete-time systems. MPC is an advanced control technique with wide industrial use. It formulates the control problem as an optimization problem, which typically is repeatedly solved on-line, at each sampling time, as soon as a new state estimate is available. The state estimation is generally obtained either using Kalman filter, or using Moving Horizon Estimation (that stands to MPC in the same way as Kalman filter stands to LQR).

MPC employs a model of the system to predict its future evolution (over a finite window of future steps) and to compute an input sequence optimal with respect to some performance index. As a difference with respect to LRQ, MPC can naturally and optimally handle constraints and changes in set point. Furthermore, it allows all matrices and vectors in the state space system, cost function and constraints to vary stage-wise. The main drawback of MPC is that it requires significantly longer time to compute the optimal control trajectory, and that this has to be repeated at each sampling time, since the optimal input sequence is a function of the current state estimate $\hat{\mathbf{x}}_0$.

7.1 Quadratic Program

In optimization, a Quadratic Program (QP) is an optimization problem with quadratic cost function and linear constraints

$$\begin{aligned}
\min_{\mathbf{v}} \quad & \frac{1}{2}\mathbf{v}^\top \widetilde{\mathbf{H}}\mathbf{v} + \widetilde{\mathbf{g}}^\top \mathbf{v} && (7.1a)\\
\text{s.t.} \quad & \widetilde{\mathbf{A}}\mathbf{v} = \widetilde{\mathbf{b}} && (7.1b)\\
& \underline{\widetilde{\mathbf{d}}} \le \widetilde{\mathbf{C}}\mathbf{v} \le \overline{\widetilde{\mathbf{d}}} && (7.1c)\\
& \underline{\widetilde{\mathbf{v}}} \le \mathbf{v} \le \overline{\widetilde{\mathbf{v}}} && (7.1d)
\end{aligned}$$

where (7.1a) is the cost function, (7.1b) are the equality constraints, (7.1c) are the inequality constraints, and (7.1d) are bounds on variable (that are a special case of general constraints, but that are much cheaper to handle from a computational point of view, and therefore often treated explicitly in numerical solvers).

7.2 Linear-Quadratic Optimal Control Problem

The MPC formulation that we will consider in this chapter is the discrete time Linear-Quadratic Optimal Control Problem (LQOCP) with box constraints

$$
\begin{aligned}
\min_{\mathbf{u},\mathbf{x}} \quad & \sum_{k=0}^{N-1} \left(\frac{1}{2}\mathbf{x}_k^\top \mathbf{Q}_k \mathbf{x}_k + \mathbf{u}_k^\top \mathbf{S}_k \mathbf{x}_k + \frac{1}{2}\mathbf{x}_k^\top \mathbf{R}_k \mathbf{x}_k + \mathbf{q}_k^\top \mathbf{x}_k + \mathbf{r}_k^\top \mathbf{u}_k \right) + \frac{1}{2}\mathbf{x}_N^\top \mathbf{Q}_N \mathbf{x}_N + \mathbf{q}_N^\top \mathbf{x}_N \\
\text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{b}_k, \quad k = 0, \ldots, N-1 \\
& \underline{\mathbf{x}}_k \le \mathbf{x}_k \le \overline{\mathbf{x}}_k, \quad k = 0, \ldots, N \\
& \underline{\mathbf{u}}_k \le \mathbf{u}_k \le \overline{\mathbf{u}}_k, \quad k = 0, \ldots, N-1
\end{aligned}
$$

The value N is called control horizon.

All matrices and vectors can generally vary at each stage of the control problem. Note that general affine constraints can be defined as well, but that we will not consider them to keep the exposition easier.

7.2.1 LQOCP as QP

When considered from an optimization point of view, the LQOCP is a QP. Therefore, it can be solved with any software for QPs (as e.g. `quadprog` in Matlab). However, the LQOCP has a special structure that can be exploited to solve it efficiently.

When the LQOCP is represented as a QP, its matrices look like (for the case $N = 2$)

$$
\widetilde{\mathbf{H}} = \begin{bmatrix} \mathbf{Q}_0 & \mathbf{S}_0^\top & 0 & 0 & 0 \\ \mathbf{R}_0 & \mathbf{S}_0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{Q}_1 & \mathbf{S}_1^\top & 0 \\ 0 & 0 & \mathbf{S}_1 & \mathbf{R}_1 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{Q}_2 \end{bmatrix} \quad , \quad \widetilde{\mathbf{g}} = \begin{bmatrix} \mathbf{q}_0 \\ \mathbf{r}_0 \\ \mathbf{q}_1 \\ \mathbf{r}_1 \\ \mathbf{q}_2 \end{bmatrix} \quad ,
$$

$$
\widetilde{\mathbf{A}} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 \\ -\mathbf{A}_0 & -\mathbf{B}_0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & -\mathbf{A}_1 & -\mathbf{B}_1 & \mathbf{I} \end{bmatrix} \quad , \quad \widetilde{\mathbf{b}} = \begin{bmatrix} \hat{\mathbf{x}}_0 \\ \mathbf{b}_0 \\ \mathbf{b}_1 \end{bmatrix} \quad ,
$$

$$
\underline{\widetilde{\mathbf{v}}_k} = \begin{bmatrix} \underline{\mathbf{x}}_0 \\ \underline{\mathbf{u}}_0 \\ \underline{\mathbf{x}}_1 \\ \underline{\mathbf{u}}_1 \\ \underline{\mathbf{x}}_2 \end{bmatrix} \quad , \quad \overline{\widetilde{\mathbf{v}}_k} = \begin{bmatrix} \overline{\mathbf{x}}_0 \\ \overline{\mathbf{u}}_0 \\ \overline{\mathbf{x}}_1 \\ \overline{\mathbf{u}}_1 \\ \overline{\mathbf{x}}_2 \end{bmatrix} \quad ,
$$

where it is clear that the matrices have a special structure, and as the horizon length N increases, they get increasingly sparse (that is, most of their elements are zero). This special structure can be efficiently exploited by specialized solvers, that work only with the dense sub-matrices.

Bibliography

[1] Otto Föllinger. *Regelungstechnik*. Hüthig, 2008.

[2] Gene F. Franklin, J. David Powell, and Abbas Emami-Naeini. *Feedback Control of Dynamic Systems*. Pearson, 2010.

[3] Jan Lunze. *Regelungstechnik 2*. Springer Vieweg, 2014.

Appendix A

Summary of Useful MATLAB Commands

In this chapter, useful MATLAB commands for this course will be given without claim to be complete. Alternatively, you can use the free software package octave.

A.1 Basic Commands

For demonstration, commands and the program output are printed below. In order to suppress the output, append ';' to the end of the line. The explanations are given as comments, which have to be preceded by '%'. Note, that for some of the following commands, the control package has to be installed. In *Octave* it has to be loaded in the beginning

```
octave:1> pkg load control
```

For demonstration of the first commands, the following matrices and vectors are used

$$\mathbf{A} = \begin{bmatrix} -0.25 & 0.25 & 0 \\ 0 & -0.2 & 0.4 \\ -1 & 0 & 0 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \qquad \mathbf{c}^\top = [1, 0, 0] \tag{A.1}$$

Setting up vectors and matrix

```
octave:1> B = [0;0;2]                              % assign column vector
B =

   0
   0
   2

octave:2> C = [1 0 0]                              % assign row vector
C =

   1   0   0

octave:3> C = [1,0,0];                             % assign row vector (alternatively)
octave:4> A = [[-0.25 0.25 0];[0 -0.2 0.4];[1 0 0]]  % matrix
A =

  -0.25000   0.25000   0.00000
   0.00000  -0.20000   0.40000
```

```
   1.00000   0.00000   0.00000

octave:5> A = [-0.25 0.25 0 ; 0 -0.2 0.4 ; 1 0 0];   % matrix (alternatively)
octave:6> A_1 = [B, B, [0;0;1]];                     % combine vectors into matrix
octave:7> A_2 = [A;C]                                % combine matrix and vector
A_2 =

  -0.25000   0.25000   0.00000
   0.00000  -0.20000   0.40000
   1.00000   0.00000   0.00000
   1.00000   0.00000   0.00000

octave:8> eye(3);                                    % 3x3 identity matrix
```

Matrix manipulations:

```
octave:9> A*B;    % multiplication of matrix and vector
octave:10> C*B;    % multiplication of vector and vector
octave:11> B*C;    % multiplication of vector and vector
octave:12> A*2;    % multiplication of matrix and scalar
```

Matrix computations

```
octave:13> det(A)        ; % determinant
octave:14> inv(A)        ; % inverse matrix
octave:15> A'            ; % transpose
octave:16> transpose(A)  ; % transpose (alternatively)
octave:17> eig(A)          % eigenvalues
ans =

   0.32785 + 0.00000i
  -0.38892 + 0.39212i
  -0.38892 - 0.39212i

octave:18> [V,D] = eig(A)  % eigenvectors and diagonal form with eigenvalues
V =

   0.25281 + 0.00000i   0.26527 - 0.26745i   0.26527 + 0.26745i
   0.58435 + 0.00000i   0.27207 + 0.56469i   0.27207 - 0.56469i
   0.77112 + 0.00000i  -0.68206 + 0.00000i  -0.68206 - 0.00000i

D =

Diagonal Matrix

   0.32785 + 0.00000i                    0                    0
                    0  -0.38892 + 0.39212i                    0
                    0                    0  -0.38892 - 0.39212i

octave:19> poly(A)         % characteristic polynomial
ans =

   1.000000   0.450000   0.050000  -0.100000

octave:20> rank(A)       ; % rank of a matrix
```

Useful stuff

```
octave:21> f = @(x) x*x  % define a function
f =

@(x) x * x

octave:22> f(2)          % call the function
ans =  4
```

A.2 ODE Simulation Example

Numerical simulation of the following system

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x} = f(\mathbf{x}) \quad \Longrightarrow \quad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -x_1 \end{bmatrix} \tag{A.2}$$

First, a an Octave function is defined

```
>> function dx = f_ode(x,t)
     dx = [x(2),-x(1)];
   endfunction
```

Then, the time vector is set up

```
>> t = (0:0.1:10)
```

corresponding to a simulation for $t = 0..10\,\mathrm{s}$ with a timestep of $0.1\,\mathrm{s}$. Further, the initial condition is defined

```
>> x0 = [1 0];
```

Finally, the simulation is carried out by

```
>> x_sol = lsode("f_ode", x0, t);
```

For plotting, use e.g.

```
>> plot(t,x_sol(:,1));
```

note, that the index `(:,1)` picks the first column from the vector.

A.3 State Space Example

In this section, some commands for treating state space systems are demonstrated.

```
octave:1> A_2 = [[-0.25 0.25 0];[0 -0.2 0.4];[0 0 -0.1]];
octave:2> B = [0;0;2];
octave:3> C = [1 0 0];
octave:4> D = 0;
octave:5> sys = ss(A_2,B,C,D);   % setup system
octave:6> step(sys)              % plot step response
octave:7> impulse(sys)           % plot impulse response
octave:8> rlocus(sys)            % plot root locus
```

The introduction of a proportional feedback control loop to the system (compare exercises) with

$$u(t) = -ky(t) \tag{A.3}$$

yields a closed loop $\mathbf{A}_{\mathrm{cl}}$-matrix

$$\mathbf{A}_{\mathrm{cl}} = \mathbf{A} - k\mathbf{b}\mathbf{c}^{\top} \tag{A.4}$$

We thus could define the control loop as system

```
octave:18> A = [[-0.25 0.25 0];[0 -0.2 0.4];[0 0 0]];
octave:19> B = [0;0;2];
octave:20> C = [1 0 0];
octave:21> roots_cl = @(k) roots(poly(A-k*B*C));  % function for EVals
octave:22> roots_cl(0.008)                    % eigenvalues of CL for k=0.008
ans =

  -0.301965
  -0.087432
  -0.060603

octave:23> roots_cl(0.2)                      % eigenvalues of CL for k=0.2
ans =

  -0.50700 + 0.00000i
   0.02850 + 0.27944i
   0.02850 - 0.27944i

octave:24> sys_cl = @(k) ss(A-k*B*C,B,C,0); % setup function for CL system
octave:25> step(sys_cl(0.008))              % plot step response (k=0.008)
octave:26> step(sys_cl(0.1))                % plot step response (k=0.1)
```