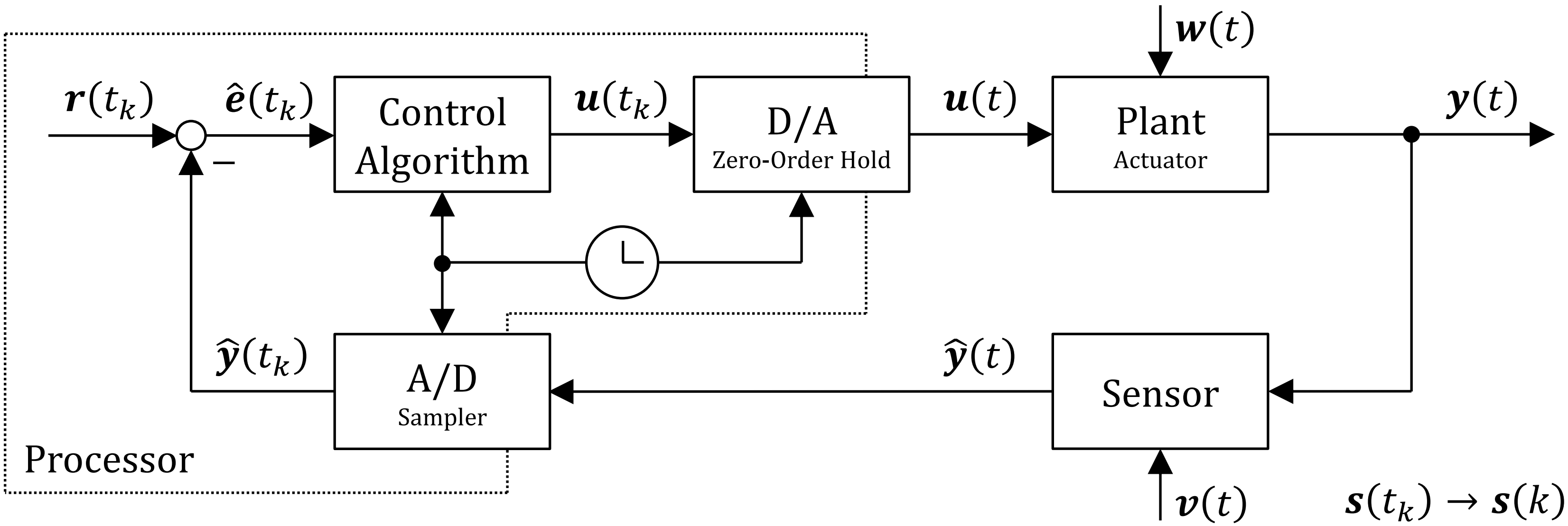
TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Model Predictive Control

2. Fundamentals of Discrete-Time Systems

Jun.-Prof. Dr.-Ing. Daniel Görges
Juniorprofessur für Elektromobilität
Technische Universität Kaiserslautern

Structure of a Sampled-Data Control System



$\boldsymbol{r}$ – reference or command input
$\boldsymbol{u}$ – control or actuator signal
$\boldsymbol{y}$ – controlled or output signal
$\boldsymbol{e} = \boldsymbol{r} - \boldsymbol{y}$ – control error

$\boldsymbol{w}$ – disturbance to the plant
$\boldsymbol{v}$ – noise in the sensor
$\hat{\boldsymbol{y}}$ – instrument or sensor output
$\hat{\boldsymbol{e}} = \boldsymbol{r} - \hat{\boldsymbol{y}}$ – indicated error

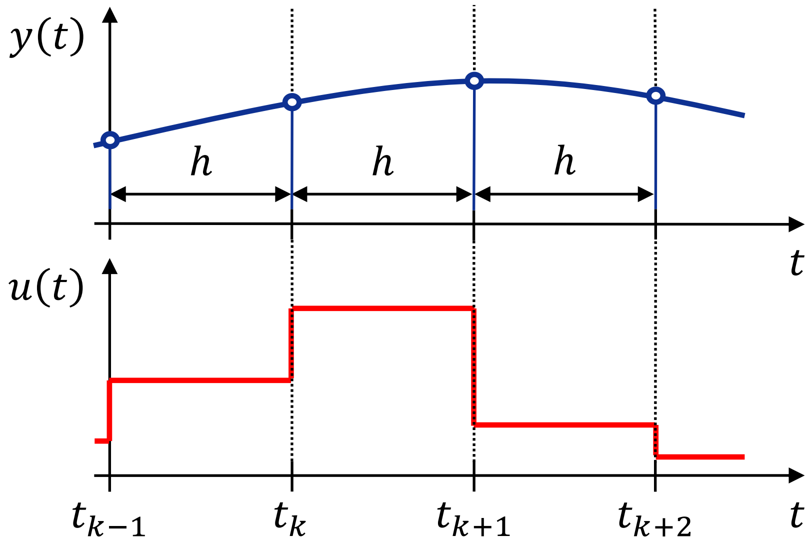
t – continuous time
k – discrete time
t_k – sampling instant
h_k – sampling period

Implementation of a Sampled-Data Control System

- **Components and Functionalities**
  - The analog-to-digital (A/D) converter samples at sampling instant t_k a voltage at the input, converts this voltage into a binary number, and writes this binary number to the output
  - The control algorithm reads the binary number from the A/D converter, evaluates the control law, and writes the result as a binary number to the D/A converter
  - The digital-to-analog (D/A) converter reads the binary number from the control algorithm, converts this binary number into a voltage, and writes this voltage to the output
  - The D/A converter usually holds the voltage over the sampling period h_k (zero-order hold (ZOH))
  - The control algorithm, A/D and D/A converter are triggered by a clock with the sampling period h_k
  - The clock is usually realized with a timer interrupt service routine on the processor
- **Remarks**
  - The control algorithm introduces a computation time (time delay from a control perspective)
  - The A/D and D/A converter introduce a quantization error (not considered in this lecture)

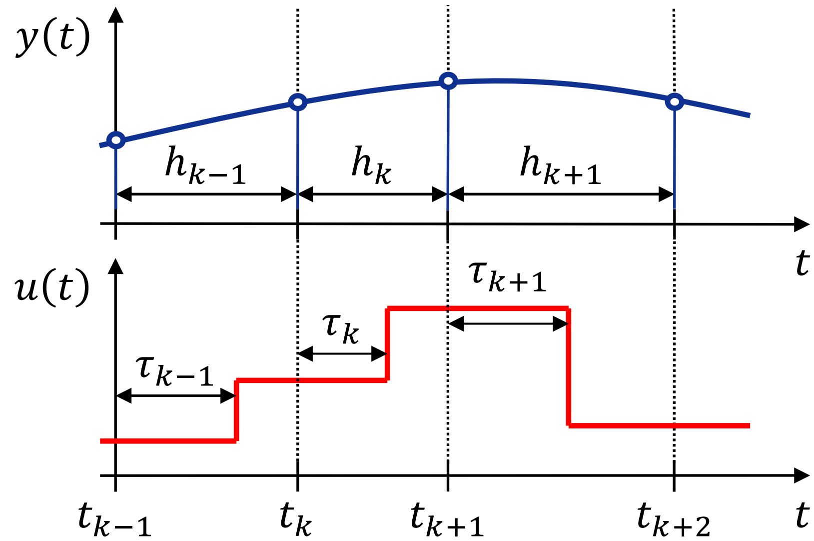
Implementation of a Sampled-Data Control System

Ideal Implementation



- Sampling period h constant
- Computation time τ negligible

Real Implementation



- Sampling period h_k time-varying
- Computation time τ_k non-negligible and varying

Implementation of a Sampled-Data Control System

- **What are Networked Embedded Control Systems? [HNX07]**
  - Controllers, sensors, and actuators are connected via a communication network
  - Controllers are implemented on processors which are embedded into the application
- **Why Networked Embedded Control Systems?**
  - Reduced wiring costs, increased reconfigurability, fewer and better utilized processors (cost aspects)
  - Control of spatially distributed systems, control of mobile systems (functional aspects)
- **What Challenges arise in Networked Embedded Control Systems?**
  - Computation and communication times can be non-negligible (e.g. due to cheap but slow processors)
  - Computation and communication times can be time-varying (e.g. due to access conflicts)
  - Sampling periods can be time-varying (e.g. due to access conflicts and packet loss)
- **What is Event-Based Control? [GHJ+14]**
  - Control only when required from a stability and performance perspective e.g. to save battery energy

Paradigms for Sampled-Data Control Systems

Control by Emulation

- Continuous-time system model
- Design a continuous-time controller based on the continuous-time system model
- Discretize the continuous-time controller using an approximation method (e.g. Tustin's method)
- Implement the discrete-time controller

Remarks

- Usually a small sampling period is required due to the approximation
- Addressed in Linear Control Systems

Direct Digital Control

- Continuous-time system model
- Discretize the continuous-time system model considering zero-order hold
- Design a discrete-time controller based on the discrete-time system model
- Implement the discrete-time controller

Remarks

- Usually a large sampling period can be utilized due to zero-order hold
- Addressed in this lecture

Active Suspension System

- **Equations of Motion**

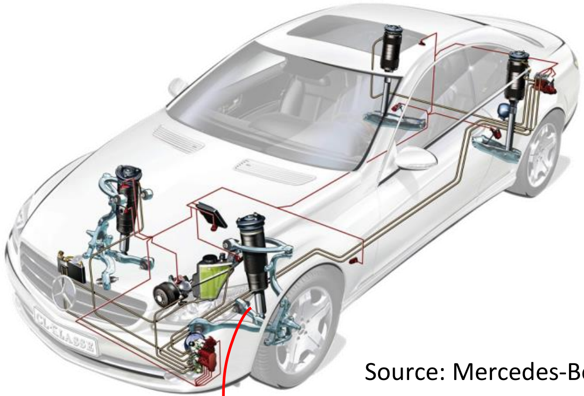
$$m_{\mathrm{s}}\ddot{z}_{\mathrm{s}} = k_{\mathrm{s}}(z_{\mathrm{u}} - z_{\mathrm{s}}) + b_{\mathrm{s}}(\dot{z}_{\mathrm{u}} - \dot{z}_{\mathrm{s}}) + F$$

$$m_{\mathrm{u}}\ddot{z}_{\mathrm{u}} = k_{\mathrm{s}}(z_{\mathrm{s}} - z_{\mathrm{u}}) + b_{\mathrm{s}}(\dot{z}_{\mathrm{s}} - \dot{z}_{\mathrm{u}}) - k_{\mathrm{u}}(z_{\mathrm{r}} - z_{\mathrm{u}}) - F$$

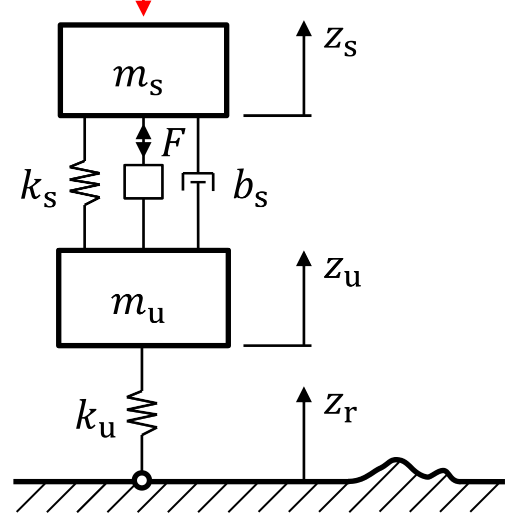
- **State-Space Model**

$$\underbrace{\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{pmatrix}}_{\dot{\boldsymbol{x}}} = \underbrace{\begin{pmatrix}0 & 1 & 0 & -1\\ -\frac{k_{\mathrm{s}}}{m_{\mathrm{s}}} & -\frac{b_{\mathrm{s}}}{m_{\mathrm{s}}} & 0 & \frac{b_{\mathrm{s}}}{m_{\mathrm{s}}}\\ 0 & 0 & 0 & 1\\ \frac{k_{\mathrm{s}}}{m_{\mathrm{u}}} & \frac{b_{\mathrm{s}}}{m_{\mathrm{u}}} & -\frac{k_{\mathrm{u}}}{m_{\mathrm{u}}} & -\frac{b_{\mathrm{s}}}{m_{\mathrm{u}}}\end{pmatrix}}_{\boldsymbol{A}_{\mathrm{c}}} \underbrace{\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\end{pmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{pmatrix}0\\ \frac{1}{m_{\mathrm{s}}}\\ 0\\ -\frac{1}{m_{\mathrm{u}}}\end{pmatrix}}_{\boldsymbol{B}_{\mathrm{c}}} u + \underbrace{\begin{pmatrix}0\\ 0\\ -1\\ 0\end{pmatrix}}_{\boldsymbol{B}_{w\mathrm{c}}} w$$

$$\underbrace{\begin{pmatrix}y_1\\ y_2\end{pmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{pmatrix}-\frac{k_{\mathrm{s}}}{m_{\mathrm{s}}} & -\frac{b_{\mathrm{s}}}{m_{\mathrm{s}}} & 0 & \frac{b_{\mathrm{s}}}{m_{\mathrm{s}}}\\ 0 & 0 & 1 & 0\end{pmatrix}}_{\boldsymbol{C}_{\mathrm{c}}} \underbrace{\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\end{pmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{pmatrix}\frac{1}{m_{\mathrm{s}}}\\ 0\end{pmatrix}}_{\boldsymbol{D}_{\mathrm{c}}} u$$



Source: Mercedes-Benz



Active Suspension System

- **States**

$x_1 = z_s - z_u$ suspension deflection

$x_2 = \dot{z}_s$ sprung mass velocity

$x_3 = z_u - z_r$ tire deflection

$x_4 = \dot{z}_u$ unsprung mass velocity

regulation problem

- **Input**

$u = F$ actuator force

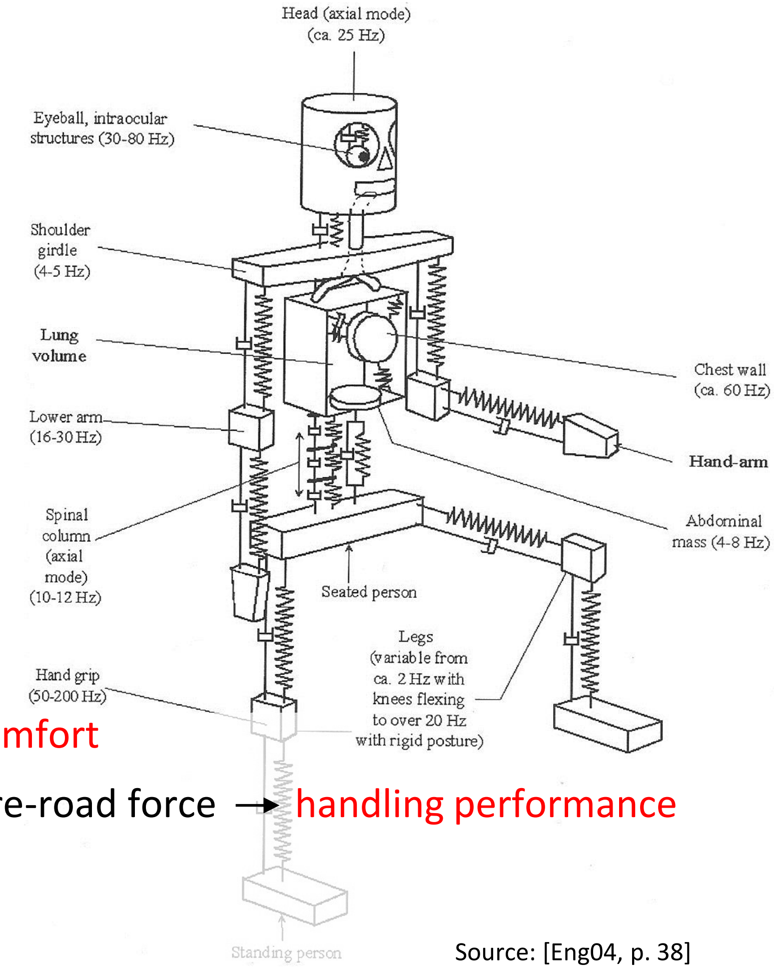
- **Outputs**

$y_1 = \ddot{z}_s$ sprung mass acceleration → ride comfort

$y_2 = z_u - z_r$ tire deflection → proportional to tire-road force → handling performance

- **Disturbance**

$w = \dot{z}_r$ derivative of the road displacement



Source: [Eng04, p. 38]

Active Suspension System

- **Parameters [PSD+08]**

$m_{\mathrm{s}} = 315\ \mathrm{kg}$	sprung mass
$k_{\mathrm{s}} = 29500\ \mathrm{N/m}$	suspension stiffness
$b_{\mathrm{s}} = 1500\ \mathrm{Ns/m}$	suspension damping
$m_{\mathrm{u}} = 37.5\ \mathrm{kg}$	tire mass
$k_{\mathrm{u}} = 210000\ \mathrm{N/m}$	tire stiffness

Source: Renault

- **Eigenvalues**

$\lambda_{1/2} = -20.52 \pm 76.31\mathrm{j}$ natural frequency $\omega_{0,1/2} = 79.02\ \mathrm{rad/s}$ damping $\zeta_{1/2} = 0.26$

$\lambda_{3/4} = -1.86 \pm 8.97\mathrm{j}$ natural frequency $\omega_{0,3/4} = 9.16\ \mathrm{rad/s}$ damping $\zeta_{3/4} = 0.20$

stiff system

- **Remark**
  - With the natural frequencies $f_{0,1/2} = \frac{\omega_{0,1/2}}{2\pi} = 12.58\ \mathrm{Hz}$ and $f_{0,3/4} = \frac{\omega_{0,3/4}}{2\pi} = 1.45\ \mathrm{Hz}$ the passive suspension system is already well designed. The damping may, however, be increased.

Discretization using ZOH

- **Continuous-Time Linear Time-Invariant (LTI) System**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_{\mathrm{c}}\boldsymbol{x}(t) + \boldsymbol{B}_{\mathrm{c}}\boldsymbol{u}(t) \qquad \text{state equation} \tag{2.1}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}_{\mathrm{c}}\boldsymbol{x}(t) + \boldsymbol{D}_{\mathrm{c}}\boldsymbol{u}(t) \qquad \text{output equation} \tag{2.2}$$

- **Symbols**

$\boldsymbol{x}(t) \in \mathbb{R}^n$ state vector $\quad$ $\boldsymbol{u}(t) \in \mathbb{R}^m$ input vector $\quad$ $\boldsymbol{y}(t) \in \mathbb{R}^p$ output vector

$\boldsymbol{A}_{\mathrm{c}} \in \mathbb{R}^{n\times n}$ system matrix $\quad$ $\boldsymbol{B}_{\mathrm{c}} \in \mathbb{R}^{n\times m}$ input matrix

$\boldsymbol{C}_{\mathrm{c}} \in \mathbb{R}^{p\times n}$ output matrix $\quad$ $\boldsymbol{D}_{\mathrm{c}} \in \mathbb{R}^{p\times m}$ feedthrough matrix

- **Solution of the Continuous-Time LTI System**

$$\boldsymbol{x}(t) = \mathrm{e}^{\boldsymbol{A}_{\mathrm{c}}(t-t_k)}\boldsymbol{x}(t_k) + \int_{t_k}^{t} \mathrm{e}^{\boldsymbol{A}_{\mathrm{c}}(t-s)}\boldsymbol{B}_{\mathrm{c}}\boldsymbol{u}(s)ds$$

- **Modeling of ZOH**

$$\boldsymbol{u}(t) = \boldsymbol{u}(t_k) \text{ for } t_k \leq t < t_{k+1}$$

Discretization using ZOH

- **Solution of the Continuous-Time LTI System over One Sampling Interval using ZOH**

$$\boldsymbol{x}(t_{k+1}) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-t_k)}\boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-s)}\boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k)ds = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}\boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-s)}ds\,\boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k)$$

$$= \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}\boldsymbol{x}(t_k) + \int_{0}^{t_{k+1}-t_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\,\boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}\boldsymbol{x}(t_k) + \int_{0}^{h_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\,\boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k)$$

- **Discrete-Time Linear Time-Varying (LTV) System**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}(k)\boldsymbol{x}(k) + \boldsymbol{B}(k)\boldsymbol{u}(k) \qquad \text{with } \boldsymbol{A}(k) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k},\ \boldsymbol{B}(k) = \int_0^{h_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\boldsymbol{B}_\mathrm{c}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) \qquad \text{with } \boldsymbol{C} = \boldsymbol{C}_c,\ \boldsymbol{D} = \boldsymbol{D}_c$$

- **Remarks**
  - The discrete-time system is time-invariant for a constant sampling period $h_k = h = \text{const.}$

Discretization using ZOH

- **Remarks**
  - The discretization using ZOH is exact, i.e. the trajectories of the continuous-time system and the discrete-time system coincide at the sampling instants t_k if the continuous-time system is linear or linear with input saturation
  - The computation of $\boldsymbol{A}(k)$ and $\boldsymbol{B}(k)$ requires the computation of a matrix exponential. This computation can be based on the Taylor series expansion

$$\mathrm{e}^{\boldsymbol{A}_\mathrm{c} h_k} = \sum_{i=0}^{\infty} \frac{\boldsymbol{A}_\mathrm{c}^i}{i!} h_k^i = \boldsymbol{I} + \boldsymbol{A}_c h_k + \frac{\boldsymbol{A}_\mathrm{c}^2}{2!} h_k^2 + \frac{\boldsymbol{A}_\mathrm{c}^3}{3!} h_k^3 + \cdots$$

$$\int_0^{h_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c} s} ds = \int_0^{h_k} \sum_{i=0}^{\infty} \frac{\boldsymbol{A}_\mathrm{c}^i}{i!} h_k^i \, ds = \sum_{i=0}^{\infty} \frac{\boldsymbol{A}_\mathrm{c}^i}{(i+1)!} h_k^{i+1} = \boldsymbol{I} h_k + \frac{\boldsymbol{A}_\mathrm{c}}{2!} h_k^2 + \frac{\boldsymbol{A}_\mathrm{c}^2}{3!} h_k^3 + \frac{\boldsymbol{A}_\mathrm{c}^3}{4!} h_k^4 + \cdots$$

    which is, however, numerically fragile. Alternative methods are given in [MV78] and [Van78].
  - The discretization using ZOH can be performed in MATLAB with `c2d`

Discretization of Systems with Time Delay

- **Continuous-Time LTI System with Time Delay**

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_c\boldsymbol{x}(t) + \boldsymbol{B}_c\boldsymbol{u}(t - \tau_k)$$

- **Assumption**

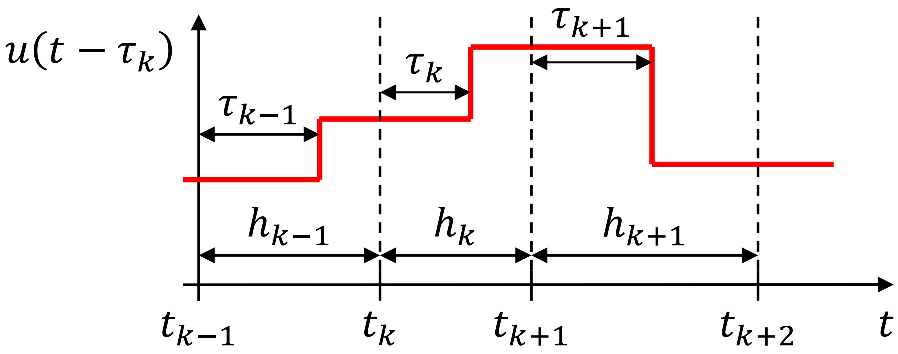
The time delay is smaller than or equal to the sampling period, i.e. $\tau_k \leq h_k$

- **Solution of the Continuous-Time LTI System with Time Delay**

$$\boldsymbol{x}(t) = \mathrm{e}^{\boldsymbol{A}_c(t-t_k)}\boldsymbol{x}(t_k) + \int_{t_k}^{t} \mathrm{e}^{\boldsymbol{A}_c(t-s)}\boldsymbol{B}_c\boldsymbol{u}(s - \tau_k)ds$$

- **Modeling of ZOH regarding the Time Delay**

$$\boldsymbol{u}(t) = \begin{cases} \boldsymbol{u}(t_{k-1}) & \text{for } t_k \leq t < t_k + \tau_k \\ \boldsymbol{u}(t_k) & \text{for } t_k + \tau_k \leq t < t_{k+1} \end{cases}$$



Discretization of Systems with Time Delay

- **Solution of the Continuous-Time LTI System with Time Delay over One Sampling Interval using ZOH**

$$\boldsymbol{x}(t_{k+1}) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-t_k)}\boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-s)}\boldsymbol{B}_\mathrm{c}\boldsymbol{u}(s-\tau_k)ds$$

$$= \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}\boldsymbol{x}(t_k) + \int_{t_k}^{t_k+\tau_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-s)}ds\, \boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_{k-1}) + \int_{t_k+\tau_k}^{t_{k+1}} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(t_{k+1}-s)}ds\, \boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k)$$

$$= \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}\boldsymbol{x}(t_k) + \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(h_k-\tau_k)}\int_0^{\tau_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\, \boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_{k-1}) + \int_0^{h_k-\tau_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\, \boldsymbol{B}_\mathrm{c}\boldsymbol{u}(t_k)$$

- **Discrete-Time LTV Equation**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}(k)\boldsymbol{x}(k) + \boldsymbol{B}_1(k)\boldsymbol{u}(k-1) + \boldsymbol{B}_0(k)\boldsymbol{u}(k)$$

with $\boldsymbol{A}(k) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}h_k}$, $\boldsymbol{B}_0(k) = \int_0^{h_k-\tau_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\,\boldsymbol{B}_\mathrm{c}$, $\boldsymbol{B}_1(k) = \mathrm{e}^{\boldsymbol{A}_\mathrm{c}(h_k-\tau_k)}\int_0^{\tau_k} \mathrm{e}^{\boldsymbol{A}_\mathrm{c}s}ds\,\boldsymbol{B}_\mathrm{c}$

Discretization of Systems with Time Delay

- **Augmented Discrete-Time LTV System**

$$\begin{pmatrix} \boldsymbol{x}(k+1) \\ \boldsymbol{u}(k) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}(k) & \boldsymbol{B}_1(k) \\ \boldsymbol{0}_{m\times n} & \boldsymbol{0}_{m\times m} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B}_0(k) \\ \boldsymbol{I}_{m\times m} \end{pmatrix} \boldsymbol{u}(k)$$

- **Remarks**
  - The component $\boldsymbol{u}(k-1)$ of the augmented state vector can be considered as a “memory”
  - The discrete-time system is time-invariant for $h_k = h = \text{const.}$ and $\tau_k = \tau = \text{const.}$
  - The discretization can be performed analogously when the time delay is larger than the sampling period, i.e. $\tau_k > h_k$. Details can be found e.g. in [ÅW90, p. 50f].
  - The continuous-time LTI system with time delay is infinite-dimensional. The stability analysis and control design for such systems is involved. The augmented discrete-time LTV system resulting by discretitzation of the continuous-time LTI system with time delay is finite-dimensional. The stability analysis and control design for such systems can be performed with standard methods.

Selection of the Sampling Period

- **Motivation**
  - A small sampling period leads to large hardware costs (processor, network, A/D and D/A converter)
  - A small sampling period leads to good control performance (usually)
  - “In general, overall system performance and budgets press to push control engineers to set as low a sampling rate as possible. Within this environment, the following three rules guide sample rate selection: 1) Sample as fast as project managers, technology, and money permit. 2) Follow the guidelines given in standard textbooks, such as Chapter 11 of [FPW97]. 3) Select a ‘reasonable’ rate and explore other choices by simulation.” [Fra07]
- **Guidelines**
  - Select the sampling period h such that there are four to ten samples N_r over the rise time T_r, i.e.

$$N_\mathrm{r} = \frac{T_\mathrm{r}}{h} = 4 \dots 10$$

[ÅW90, Section 3.7]

  - The rise time T_r of a first-order system is equal to the time constant

Selection of the Sampling Period

- **Guidelines**
  - The rise time T_{r} of a second-order system is given by

    $$T_{\mathrm{r}} = \omega_0^{-1} \mathrm{e}^{\varphi / \tan \varphi}$$

    where ω_0 is the natural frequency and $\zeta = \cos \varphi$ is the damping
  - The rise time T_{r} for higher-order systems depends on the control objective, specifically

    reference tracking (often low bandwidth signals, then only slow poles relevant)
    disturbance rejection (often high bandwidth signals, then also fast poles relevant)
    robustness w.r.t. uncertainties (often smaller sampling period recommendable)

- **Remarks**
  - The sampling period for closed-loop systems should be selected based on the closed-loop poles.
  - The sampling period should not be selected too small since due to $z = \mathrm{e}^{sh}$ all poles will be mapped close to $z = 1$. This can cause numerical problems on processors with small word length (e.g. 8 bit).

Solution of Discrete-Time Linear Time-Invariant Systems

- **Discrete-Time Linear Time-Invariant (LTI) System**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \qquad \text{state equation} \qquad (2.3)$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) \qquad \text{output equation} \qquad (2.4)$$

- **Symbols**

$\boldsymbol{x}(k) \in \mathbb{R}^n$ state vector $\boldsymbol{u}(k) \in \mathbb{R}^m$ input vector $\boldsymbol{y}(k) \in \mathbb{R}^p$ output vector

$\boldsymbol{A} \in \mathbb{R}^{n\times n}$ system matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ input matrix

$\boldsymbol{C} \in \mathbb{R}^{p\times n}$ output matrix $\boldsymbol{D} \in \mathbb{R}^{p\times m}$ feedthrough matrix

- **Solution of the Discrete-Time LTI System**

$$\boldsymbol{x}(k) = \boldsymbol{A}^k\boldsymbol{x}(0) + \sum_{i=0}^{k-1} \boldsymbol{A}^{k-i-1}\boldsymbol{B}\boldsymbol{u}(i) \qquad (2.5)$$

- **Remark**
  - A nice proof will be given on Slide 4-7

Definitions

- **Discrete-Time Nonlinear Time-Varying System**

$$\boldsymbol{x}(k+1) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), k), \quad \boldsymbol{x}(k_0) = \boldsymbol{x}_{k_0} \tag{2.6}$$

- **Symbols**
  - $\boldsymbol{x}(k) \in \mathbb{R}^n$ state vector
  - $\boldsymbol{u}(k) \in \mathbb{R}^m$ input vector
  - $k \in \mathbb{N}_0$ discrete time
  - $k_0 \in \mathbb{N}_0$ initial time
  - $\boldsymbol{f}: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{N}_0 \to \mathbb{R}^n$ nonlinear function

- **Remarks**
  - Autonomous system $\triangleq$ unforced (i.e. $\boldsymbol{u}(k) = \boldsymbol{0}$) and time-invariant (i.e. $\boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), k) = \boldsymbol{f}\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big)$)
  - Non-autonomous system $\triangleq$ forced and/or time-varying
  - In the following an unforced time-varying system (i.e. $\boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), k) = \boldsymbol{f}(\boldsymbol{x}(k), k)$) is considered

Definitions

Definition 2.1 A state vector $\boldsymbol{x}_\mathrm{e}$ is an equilibrium point of the discrete-time nonlinear time-varying system (2.6) iff $\boldsymbol{f}(\boldsymbol{x}_\mathrm{e}, k) = \boldsymbol{x}_\mathrm{e}\ \forall k \geq k_0$.

- **Remarks**
  - Note that this definition differs from the definition for continuous-time systems
  - Stability definitions and criteria are usually formulated assuming that $\boldsymbol{x}_\mathrm{e} = \boldsymbol{0}$ is an equilibrium point. This assumption is not restrictive since definitions and criteria for other equilibrium points can always be reformulated to definitions and criteria for the equilibrium point $\boldsymbol{x}_\mathrm{e} = \boldsymbol{0}$ by a change of variables.

Definition 2.2 The equilibrium point $\boldsymbol{x}_\mathrm{e} = \boldsymbol{0}$ of the discrete-time nonlinear time-varying system (2.6) is

- stable at k_0 if for each $\varepsilon > 0$ there exists a $\delta = \delta(\varepsilon, k_0) > 0$ such that

$$\|\boldsymbol{x}(k_0)\| < \delta \Rightarrow \|\boldsymbol{x}(k)\| < \varepsilon \quad \forall k \geq k_0, \tag{2.7}$$

Stability

Definitions

- uniformly stable if for each $\varepsilon > 0$ there exists a $\delta = \delta(\varepsilon) > 0$ independ. of k_0 such that (2.7) is fulfilled,
- asymptotically stable at k_0 if it is stable and there exists a $\delta' = \delta'(k_0) > 0$ such that

$$\|\boldsymbol{x}(k_0)\| < \delta' \Rightarrow \lim_{k\to\infty} \|\boldsymbol{x}(k)\| = 0, \tag{2.8}$$

- uniformly asymptotically stable if it is uniformly stable and there exists a $\delta'(\varepsilon') > 0$ indep. of k_0 such that (2.8) is fulfilled uniformly in k_0, i.e. for each $\varepsilon' > 0$ there exists a $K = K(\varepsilon')$ indep. of k_0 such that

$$\|\boldsymbol{x}(k_0)\| < \delta' \Rightarrow \|\boldsymbol{x}(k)\| < \varepsilon' \quad \forall k \geq k_0 + K,$$

- globally uniformly asymptotically stable if it is uniformly asymptotically stable for all $\boldsymbol{x}(k_0) \in \mathbb{R}^n$,
- unstable if it is not stable.

- **Remark**
  - By considering uniform stability the dependence on the initial time k_0 can be removed
  - For discrete-time time-invariant systems uniform stability and stability are equivalent

Definitions

Definition 2.3 A function $V: \mathbb{D} \to \mathbb{R}$ is

- positive semidefinite in $\mathbb{D} \subset \mathbb{R}^n$ if

  (1) $V(\mathbf{0}) = 0$

  (2) $V\big(\boldsymbol{x}(k)\big) \geq 0 \quad \forall \boldsymbol{x}(k) \in \mathbb{D}\backslash\{\mathbf{0}\}$,

- positive definite in $\mathbb{D} \subset \mathbb{R}^n$ if (2) is replaced by

  (2') $V\big(\boldsymbol{x}(k)\big) > 0 \quad \forall \boldsymbol{x}(k) \in \mathbb{D}\backslash\{\mathbf{0}\}$,

- negative definite (semidefinite) in $\mathbb{D} \subset \mathbb{R}^n$ if $-V$ is positive definite (semidefinite).

Definition 2.4 A function $V: \mathbb{D} \times \mathbb{N}_0 \to \mathbb{R}$ is

- positive semidefinite in $\mathbb{D} \subset \mathbb{R}^n$ if

  (1) $V(\mathbf{0}, k) = 0 \qquad \forall k \in \mathbb{N}_0$

  (2) $V(\boldsymbol{x}(k), k) \geq 0 \quad \forall \boldsymbol{x}(k) \in \mathbb{D}\backslash\{\mathbf{0}\} \quad \forall k \in \mathbb{N}_0$,

Definitions

- positive definite in $\mathbb{D} \subset \mathbb{R}^n$ if (2) is replaced by

  (2') there exists a positive definite function $V_1 : \mathbb{D} \to \mathbb{R}$ independent of k such that

$$V_1\big(\boldsymbol{x}(k)\big) \leq V\big(\boldsymbol{x}(k), k\big) \quad \forall \boldsymbol{x}(k) \in \mathbb{D} \quad \forall k \in \mathbb{N}_0,$$

- negative definite (semidefinite) in $\mathbb{D} \subset \mathbb{R}^n$ if $-V$ is positive definite (semidefinite),
- decrescent if there exists a positive definite function $V_2 : \mathbb{D} \to \mathbb{R}$ independent of k such that

$$V\big(\boldsymbol{x}(k), k\big) \leq V_2\big(\boldsymbol{x}(k)\big) \quad \forall \boldsymbol{x}(k) \in \mathbb{D} \quad \forall k \in \mathbb{N}_0,$$

- radially unbounded if there exists a positive definite function $V_1 : \mathbb{D} \to \mathbb{R}$ independent of k with $V_1\big(\boldsymbol{x}(k)\big) \to \infty$ as $\|\boldsymbol{x}(k)\| \to \infty$ such that

$$V_1\big(\boldsymbol{x}(k)\big) \leq V\big(\boldsymbol{x}(k), k\big) \quad \forall \boldsymbol{x}(k) \in \mathbb{D} \quad \forall k \in \mathbb{N}_0.$$

Lyapunov‘s Direct Method

Theorem 2.1 If in a neighborhood $\mathbb{D} \subset \mathbb{R}^n$ of the equilibrium point $\boldsymbol{x}_\mathrm{e} = \mathbf{0}$ of the discrete-time nonlinear time-varying system (2.6) there exists a function $V: \mathbb{D} \times \mathbb{N}_0 \rightarrow \mathbb{R}$ such that

(1) $V(\boldsymbol{x}(k), k)$ is positive definite,

(2) $\Delta V(\boldsymbol{x}(k), k) = V(\boldsymbol{x}(k+1), k+1) - V(\boldsymbol{x}(k), k)$ is negative semidefinite,

then the equilibrium point is stable. If furthermore

(3) $V(\boldsymbol{x}(k), k)$ is decrescent,

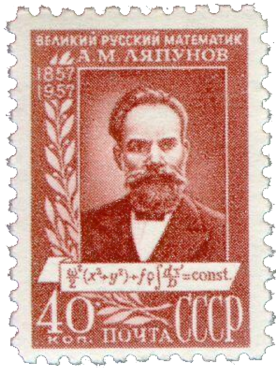
then the equilibrium point is uniformly stable. If furthermore

(2‘) $\Delta V(\boldsymbol{x}(k), k) = V(\boldsymbol{x}(k+1), k+1) - V(\boldsymbol{x}(k), k)$ is negative definite,

then the equilibrium point is uniformly asymptotically stable. If furthermore $\mathbb{D} = \mathbb{R}^n$ and

(4) $V(\boldsymbol{x}(k), k)$ is radially unbounded,

then the equilibrium point is globally uniformly asymptotically stable.



Lyapunov‘s Direct Method

- **Remarks**
  - A function $V(\boldsymbol{x}(k), k)$ fulfilling at least conditions (1) and (2) is called Lyapunov function.
  - The function $V(\boldsymbol{x}(k), k)$ describes the energy stored in the system (2.6) in an abstract way. Requiring $\Delta V(\boldsymbol{x}(k), k)$ to be negative definite thus corresponds to requiring the energy to decrease.
  - Lyapunov stability relates to an unforced system (i.e. $\boldsymbol{u}(k) = \mathbf{0}$) or to a closed-loop system (i.e. $\boldsymbol{u}(k) = \boldsymbol{f}_{\mathrm{C}}(\boldsymbol{x}(k), k)$ with the function $\boldsymbol{f}_{\mathrm{C}}: \mathbb{R}^n \times \mathbb{N}_0 \rightarrow \mathbb{R}^m$ describing some control law). The latter leads to a control Lyapunov function (CLF).
  - Global uniform asymptotic stability of the equilibrium point $\boldsymbol{x}_{\mathrm{e}} = \mathbf{0}$ implies uniqueness of this equilibrium point. The system (2.6) is therefore commonly denoted itself as globally uniformly asymptotically stable.
  - Often quadratic Lyapunov functions (QLFs) described by quadratic forms are considered.

Quadratic Forms

Tutorial

Definition 2.5 A function $f: \mathbb{R}^n \to \mathbb{R}$, $f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} = \|\boldsymbol{x}\|_{\boldsymbol{P}}^2$ with $\boldsymbol{P} \in \mathbb{R}^{n \times n}$ symmetric is a quadratic form.

Definition 2.6 A quadratic form $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x}$ with $\boldsymbol{P} \in \mathbb{R}^{n \times n}$ symmetric is

- positive definite if $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} > 0 \ \forall \boldsymbol{x} \in \mathbb{R}^n \backslash \{\boldsymbol{0}\}$,
- positive semidefinite if $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} \geq 0 \ \forall \boldsymbol{x} \in \mathbb{R}^n$,
- negative definite if $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} < 0 \ \forall \boldsymbol{x} \in \mathbb{R}^n \backslash \{\boldsymbol{0}\}$,
- negative semidefinite if $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} \leq 0 \ \forall \boldsymbol{x} \in \mathbb{R}^n$,
- indefinite otherwise.

- **Remarks**
  - $\boldsymbol{P}$ symmetric is not restrictive since $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} = \boldsymbol{x}^T 1/2(\boldsymbol{P} + \boldsymbol{P}^T)\boldsymbol{x}$ where $1/2(\boldsymbol{P} + \boldsymbol{P}^T)$ is symmetric
  - $\boldsymbol{P}$ is called positive definite ($\boldsymbol{P} \succ \boldsymbol{0}$), positive semidefinite ($\boldsymbol{P} \succcurlyeq \boldsymbol{0}$), negative definite ($\boldsymbol{P} \prec \boldsymbol{0}$), negative semidefinite ($\boldsymbol{P} \preccurlyeq \boldsymbol{0}$) or indefinite if the quadratic form $\boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x}$ has the related properties

Quadratic Forms

Tutorial

Theorem 2.2 The quadratic form $\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$ with $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ symmetric is

- positive definite iff $\lambda_i(\boldsymbol{P}) > 0$ or $\mathrm{D}_i(\boldsymbol{P}) > 0 \quad \forall i \in \{1, \dots, n\}$,
- positive semidefinite iff $\lambda_i(\boldsymbol{P}) \geq 0$ or $\mathrm{D}_i(\boldsymbol{P}) \geq 0 \quad \forall i \in \{1, \dots, n\}$,
- negative definite iff $\lambda_i(\boldsymbol{P}) < 0$ or $(-1)^i\mathrm{D}_i(\boldsymbol{P}) < 0 \quad \forall i \in \{1, \dots, n\}$,
- negative semidefinite iff $\lambda_i(\boldsymbol{P}) \leq 0$ or $(-1)^i\mathrm{D}_i(\boldsymbol{P}) \leq 0 \quad \forall i \in \{1, \dots, n\}$,
- indefinite otherwise,

where $\lambda_i(\boldsymbol{P})$ denotes the ith eigenvalue of $\boldsymbol{P}$ and $\mathrm{D}_i(\boldsymbol{P})$ denotes the ith leading principal minor of $\boldsymbol{P}$

- **Remark**
  - The leading principal minors of $\boldsymbol{P}$ are defined as

$$\mathrm{D}_1(\boldsymbol{P}) = p_{11},\ \mathrm{D}_2(\boldsymbol{P}) = \det\begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix},\ \dots,\ \mathrm{D}_n(\boldsymbol{P}) = \det\boldsymbol{P}$$

Quadratic Forms

Tutorial

Lemma 2.1 The minimum and maximum value of the quadratic form $\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$ with $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ symmetric on the unit hypersphere $\boldsymbol{\mathcal{S}} = \{\boldsymbol{x} \in \mathbb{R}^n | \|\boldsymbol{x}\|_2 = 1\}$ is given by

$$\min_{\boldsymbol{x}\in\boldsymbol{\mathcal{S}}} \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x} = \lambda_{\min}(\boldsymbol{P})$$

$$\max_{\boldsymbol{x}\in\boldsymbol{\mathcal{S}}} \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x} = \lambda_{\max}(\boldsymbol{P}),$$

where $\lambda_{\min}(\boldsymbol{P}) = \min\{\lambda_1(\boldsymbol{P}), \dots, \lambda_n(\boldsymbol{P})\}$ and $\lambda_{\max}(\boldsymbol{P}) = \max\{\lambda_1(\boldsymbol{P}), \dots, \lambda_n(\boldsymbol{P})\}$.
This leads to the Rayleigh-Ritz inequality

$$\lambda_{\min}(\boldsymbol{P})\|\boldsymbol{x}\|_2^2 \leq \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x} \leq \lambda_{\max}(\boldsymbol{P})\|\boldsymbol{x}\|_2^2.$$

- **Remark**
  - A proof of Lemma 2.1 is given in [Mey00, Example 7.5.1] and [HJ85, Theorem 4.2.2]

Quadratic Forms

Tutorial

Lemma 2.2 Let $\boldsymbol{x} \in \mathbb{R}^n$ be a Gaussian random variable with expected value $\mathrm{E}(\boldsymbol{x}) = \mathbf{0}$ and covariance matrix $\mathrm{E}(\boldsymbol{x}\boldsymbol{x}^T) = \boldsymbol{I}$. The expected value of the quadratic form $\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$ with $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ symmetric is then given by

$$\mathrm{E}(\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}) = \mathrm{tr}\,\boldsymbol{P},$$

where $\mathrm{tr}\,\boldsymbol{P}$ denotes the trace of $\boldsymbol{P}$.

- **Remarks**
  - Note that $\mathrm{tr}\,\boldsymbol{P} = p_{11} + p_{22} + \cdots + p_{nn} = \lambda_1(\boldsymbol{P}) + \lambda_2(\boldsymbol{P}) + \cdots + \lambda_n(\boldsymbol{P})$
  - A proof of Lemma 2.2 and a more general formulation is given in [ÅW90, p. 338]
- **Useful Facts**
  - $\boldsymbol{P} \succ \mathbf{0} \Leftrightarrow \boldsymbol{P}^{-1} \succ \mathbf{0}, \quad \boldsymbol{P} \prec \mathbf{0} \Leftrightarrow \boldsymbol{P}^{-1} \prec \mathbf{0}$
  - $\boldsymbol{P} \succ \mathbf{0} \Rightarrow \mathrm{tr}\,\boldsymbol{P} > 0, \quad \boldsymbol{P} \succcurlyeq \mathbf{0} \Rightarrow \mathrm{tr}\,\boldsymbol{P} \geq 0$
  - $\boldsymbol{P} \succ \mathbf{0} \Rightarrow \det \boldsymbol{P} > 0, \quad \boldsymbol{P} \succcurlyeq \mathbf{0} \Rightarrow \det \boldsymbol{P} \geq 0$
  - $\boldsymbol{P} \succ \mathbf{0} \Rightarrow p_{ii} > 0 \;\; \forall i \in \{1, \dots, n\}$

Stability of Discrete-Time Linear Time-Invariant Systems

- **Discrete-Time Linear Time-Invariant (LTI) System**

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) \tag{2.9}$$

- **Equilibrium Points**
  - The origin $\boldsymbol{x}_\mathrm{e} = \boldsymbol{0}$ is an equilibrium point
  - Any eigenvector $\boldsymbol{x}_\mathrm{e}$ of the system matrix $\boldsymbol{A}$ related to the eigenvalue $\lambda_\mathrm{e} = 1$ is an equilibrium point (follows immediately from the eigenvector equation $\boldsymbol{A}\boldsymbol{x}_\mathrm{e} = \lambda_\mathrm{e}\boldsymbol{x}_\mathrm{e}$)

Theorem 2.3 The discrete-time linear time-invariant system (2.9) is globally asymptotically stable iff all eigenvalues of the system matrix $\boldsymbol{A}$ are inside the unit circle (i.e. $\rho(\boldsymbol{A}) < 1$).

- **Remarks**
  - The discrete-time LTI system (2.9) corresponds to the unforced discrete-time LTI system (2.3)
  - $\rho(\boldsymbol{A}) = \max\{|\lambda_1(\boldsymbol{A})|, \dots, |\lambda_n(\boldsymbol{A})|\}$ is the spectral radius of the system matrix $\boldsymbol{A}$

Stability of Discrete-Time Linear Time-Invariant Systems

- **Remarks**
  - Asymptotic stability is always „global“ for linear systems
  - If the discrete-time LTI system (2.3) is obtained by discretizing the continuous-time LTI system (2.1) using ZOH, then their eigenvalues are related by $\lambda_i(\boldsymbol{A}) = \mathrm{e}^{\lambda_i(\boldsymbol{A}_\mathrm{c})h}$, see [Lun13, p. 449] for the proof
  - Continuous-time LTI system (2.1) g. a. stable $\Rightarrow$ discrete-time LTI system (2.3) g. a. stable under ZOH

Theorem 2.4 The discrete-time linear time-invariant system (2.9) is globally asymptotically stable iff there exists a matrix $\boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$ such that $\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P} \prec \boldsymbol{0}$.

- **Proof (only Sufficiency)**
  - Let‘s consider the quadratic Lyapunov function candidate $V\big(\boldsymbol{x}(k)\big) = \boldsymbol{x}^T(k)\boldsymbol{P}\boldsymbol{x}(k)$ with $\boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$
  - The function $V\big(\boldsymbol{x}(k)\big)$ is positive definite and radially unbounded since $\alpha_1 \|\boldsymbol{x}(k)\|_2^2 \leq V\big(\boldsymbol{x}(k)\big)\ \ \forall \boldsymbol{x}(k) \in \mathbb{R}^n$ with $\alpha_1 = \lambda_{\min}(\boldsymbol{P}) > 0$ due to $\boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$, cf. Lemma 2.1

Stability of Discrete-Time Linear Time-Invariant Systems

- **Proof (only Sufficiency)**
  - We must still prove when $\Delta V\big(\boldsymbol{x}(k)\big)$ along trajectories of the discrete-time LTI system (2.9), i.e.

    $$\Delta V\big(\boldsymbol{x}(k)\big) = V\big(\boldsymbol{x}(k+1)\big) - V\big(\boldsymbol{x}(k)\big) = \boldsymbol{x}^T(k+1)\boldsymbol{P}\boldsymbol{x}(k+1) - \boldsymbol{x}^T(k)\boldsymbol{P}\boldsymbol{x}(k)$$
    $$= \boldsymbol{x}^T(k)\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}\boldsymbol{x}(k) - \boldsymbol{x}^T(k)\boldsymbol{P}\boldsymbol{x}(k) = \boldsymbol{x}^T(k)(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P})\boldsymbol{x}(k),$$

    is negative definite
  - Obviously, $\Delta V\big(\boldsymbol{x}(k)\big)$ along trajectories of the discrete-time LTI system (2.9) is negative definite if $\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P} \prec \boldsymbol{0}$ since then $\Delta V\big(\boldsymbol{x}(k)\big) \leq \alpha_2 \|\boldsymbol{x}(k)\|_2^2 \ \forall \boldsymbol{x}(k) \in \mathbb{R}^n$ with $\alpha_2 = \lambda_{\max}(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}) < 0$, cf. Lemma 2.1

Corollary 2.1 The discrete-time linear time-invariant system (2.9) is globally asymptotically stable iff the discrete-time Lyapunov equation $\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A} - \boldsymbol{P} = -\boldsymbol{Q}$ has a solution $\boldsymbol{P} = \boldsymbol{P}^T \succ \boldsymbol{0}$ for any $\boldsymbol{Q} = \boldsymbol{Q}^T \succ \boldsymbol{0}$.

State Feedback Control

- **Assumptions**
  - All states can be measured, i.e. $\boldsymbol{C} = \boldsymbol{I}_{n\times n}$
  - There is no reference input, i.e. $\boldsymbol{r} = \boldsymbol{0}$ resulting controller usually called regulator
- **State Feedback Control Law**

$$\boldsymbol{u}(k) = \boldsymbol{K}\boldsymbol{x}(k) = (\boldsymbol{k}_1 \quad \boldsymbol{k}_2 \quad \cdots \quad \boldsymbol{k}_n)\boldsymbol{x}(k) \tag{2.10}$$

- **Closed-Loop System**
  - Substituting the state feedback control law (2.10) into the state equation (2.3) leads to

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{K}\boldsymbol{x}(k) = (\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})\boldsymbol{x}(k) \tag{2.11}$$

- **Characteristic Equation**

$$\boldsymbol{x}(k+1) = (\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})\boldsymbol{x}(k) \circ\!\!-\!\!\bullet \; z\boldsymbol{X}(z) = (\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K})\boldsymbol{X}(z) \Leftrightarrow (z\boldsymbol{I}_{n\times n} - \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\boldsymbol{X}(z) = \boldsymbol{0}$$

$$\det(z\boldsymbol{I}_{n\times n} - \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) = 0$$

State Feedback Control Design based on Pole Placement

- **Assumption**
  - Single-input single-output (SISO) system, i.e. $m = 1, p = 1$
- **Pole Placement**
  1. Specify desired poles $\{\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_n\}$ with $|\tilde{\lambda}_i| < 1$ and complex poles as conjugate complex pairs
  2. Compute the desired characteristic polynomial $(z - \tilde{\lambda}_1)(z - \tilde{\lambda}_2) \cdots (z - \tilde{\lambda}_n)$
  3. Compute the characteristic polynomial $\det(z\boldsymbol{I}_{n\times \mathrm{n}} - \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})$
  4. Set $\det(z\boldsymbol{I}_{n\times n} - \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}) = (z - \tilde{\lambda}_1)(z - \tilde{\lambda}_2) \cdots (z - \tilde{\lambda}_n)$
  5. Solve for $k_1, k_2, \dots, k_n$, e.g. by comparison of coefficients
- **Remark**
  - Pole placement can also be applied for multiple-input multiple-output (MIMO) systems.
    To this end, more complex methods like modal synthesis are required, see e.g. [Lun13, Sec. 6.3.2].
- **Can the poles of the closed-loop system be chosen arbitrarily using pole placement?**

Controllability

Definition and Analysis

Definition 2.7 The discrete-time linear time-invariant system (2.3) or equivalently the pair $(\boldsymbol{A}, \boldsymbol{B})$ is controllable if the system can be transferred from any initial state $\boldsymbol{x}(0)$ to any final state $\boldsymbol{x}(N)$ in finite time N by a suitable input sequence $\boldsymbol{u}(0), \boldsymbol{u}(1), \dots, \boldsymbol{u}(N-1)$.

Theorem 2.5 The following statements are equivalent:

(1) The pair $(\boldsymbol{A}, \boldsymbol{B})$ is controllable.

(2) The controllability matrix $\boldsymbol{\mathcal{C}} = (\boldsymbol{B} \quad \boldsymbol{A}\boldsymbol{B} \quad \cdots \quad \boldsymbol{A}^{n-1}\boldsymbol{B})$ has full rank n.

(3) For a state feedback control law (2.10) the eigenvalues of the resulting closed-loop system (2.11) can be selected arbitrarily by a suitable selection of the feedback matrix $\boldsymbol{K}$.

- **Remark**
  - The discrete-time LTI system (2.3) obtained by discretizing the continuous-time LTI system (2.1) using ZOH is controllable iff (2.1) is controllable and for any two different eigenvalues $\lambda_i \neq \lambda_j$ of $\boldsymbol{A}_\mathrm{c}$ with $\mathrm{Re}\{\lambda_i\} = \mathrm{Re}\{\lambda_j\}$ the relation $\mathrm{Im}\{\lambda_i\} - \mathrm{Im}\{\lambda_j\} \neq 2k\pi/h, k = \pm 1, \pm 2, \dots$ holds.

Definition and Analysis

Definition 2.8 The discrete-time linear time-invariant system (2.3) or equivalently the pair $(\boldsymbol{A}, \boldsymbol{B})$ is stabilizable if there exists a feedback matrix $\boldsymbol{K}$ such that the closed-loop system (2.11) is globally asymptotically stable, i.e. if the unstable eigenvalues of (2.3) are controllable.

Theorem 2.6 Let $\Lambda = \{\lambda_i(\boldsymbol{A}) : |\lambda_i(\boldsymbol{A})| \geq 1\}$ be the set of eigenvalues on or outside the unit disk. Then the pair $(\boldsymbol{A}, \boldsymbol{B})$ is stabilizable iff $(\boldsymbol{A} - \lambda_i \boldsymbol{I}_{n\times n} \quad \boldsymbol{B})$ has full rank n for all $\lambda_i \in \Lambda$.

- **Remarks**
  - Controllability $\Rightarrow$ stabilizability, but stabilizability $\nRightarrow$ controllability
  - Stabilizability does in particular not guarantee that the eigenvalues of closed-loop system (2.11) can be selected arbitrarily by a suitable selection of the feedback matrix $\boldsymbol{K}$

Prediction Observer

- **Motivation**
  - A measurement of all states is often not possible, e.g. because the sensors are too expensive or because the states are not physical quantities (e.g. modes in active vibration control)
- **Approach**
  - Estimate the state $\hat{\boldsymbol{x}}(k)$ using the state equation (2.3), i.e.

    $$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k)$$

  - This will generally work since $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{u}(k)$ are known. The initial state $\boldsymbol{x}(0)$ is, however, unknown and furthermore disturbances may occur, leading to an estimation error $\tilde{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$ obeying

    $$\tilde{\boldsymbol{x}}(k+1) = \boldsymbol{x}(k+1) - \hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \cancel{\boldsymbol{B}\boldsymbol{u}(k)} - \boldsymbol{A}\hat{\boldsymbol{x}}(k) - \cancel{\boldsymbol{B}\boldsymbol{u}(k)} = \boldsymbol{A}\big(\boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)\big) = \boldsymbol{A}\tilde{\boldsymbol{x}}(k)$$

  - For an asymptotically stable $\boldsymbol{A}$ the estimation error $\tilde{\boldsymbol{x}}(k)$ converges, but not for an unstable
  - Introduce an output error feedback to obtain convergence, i.e.

    $$\hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{L}\big(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)\big) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{L}(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k)) \qquad (2.12)$$

Prediction Observer

- **Approach**
  - The estimation error $\tilde{\boldsymbol{x}}(k)$ then obeys

    $$\tilde{\boldsymbol{x}}(k+1) = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\tilde{\boldsymbol{x}}(k) \tag{2.13}$$

  - If $\boldsymbol{L}$ is chosen such that $\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ is asymptotically stable, then the estimation error $\tilde{\boldsymbol{x}}(k)$ converges
  - This can be achieved e.g. by pole placement
- **Remarks**
  - The equation (2.12) is called prediction observer or Luenberger observer since the state $\hat{\boldsymbol{x}}(k+1)$ at time $k+1$ is predicted based on the measured output $\boldsymbol{y}(k)$ at time k
  - Note that $\lambda_i(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}) = \lambda_i\big((\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})^T\big) = \lambda_i(\boldsymbol{A}^T - \boldsymbol{C}^T\boldsymbol{L}^T)\ \ \forall i \in \{1, \dots, n\}$
  - Since the estimation error $\tilde{\boldsymbol{x}}(k)$ must converge much faster than the state $\boldsymbol{x}(k)$, the poles of $\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ must be chosen much faster than the poles of $\boldsymbol{A}$ or the poles of $\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}$ if the prediction observer is combined with state feedback control

Prediction Observer

- **Remarks**
  - A large $\boldsymbol{L}$ results in a large observer input, which, different from a large control input, is not critical since the observer is realized entirely in a computer
  - A large $\boldsymbol{L}$ can be critical if the measurement noise $\boldsymbol{v}(k)$ is large since for $\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{v}(k)$ the estimation error obeys $\tilde{\boldsymbol{x}}(k+1) = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\tilde{\boldsymbol{x}}(k) - \boldsymbol{L}\boldsymbol{v}(k)$
  - Commonly the poles of $\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ are chosen 2 to 6 times faster than the poles of $\boldsymbol{A}$ or $\boldsymbol{A} + \boldsymbol{B}\boldsymbol{K}$, yielding a good compromise between fast convergence and noise rejection
  - Commonly the initial state $\hat{\boldsymbol{x}}(k) = \boldsymbol{0}$ is chosen for the observer if no further information is available
- **Implementation for State Feedback Control**
  1. Determine $\boldsymbol{u}(k) = \boldsymbol{K}\hat{\boldsymbol{x}}(k)$ — at time k for time k
  2. Determine $\hat{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{L}(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k))$ — at time k for time $k+1$
- **Can the poles of the observer be chosen arbitrarily using pole placement?**

Observability

Definition and Analysis

Definition 2.9 The discrete-time linear time-invariant system $(2.3)/(2.4)$ or equivalently the pair $(\boldsymbol{C}, \boldsymbol{A})$ is observable if any initial state $\boldsymbol{x}(0)$ can be determined from the finite known input sequence $\boldsymbol{u}(0), \boldsymbol{u}(1), \dots, \boldsymbol{u}(N-1)$ and the finite measured output sequence $\boldsymbol{y}(0), \boldsymbol{y}(1), \dots, \boldsymbol{y}(N-1)$

Theorem 2.7 The following statements are equivalent:

(1) The pair $(\boldsymbol{C}, \boldsymbol{A})$ is observable.

(2) The observability matrix $\boldsymbol{\mathcal{O}} = \begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^{n-1} \end{pmatrix}$ has full rank n.

(3) For the prediction observer (2.12) the eigenvalues of the resulting estimation error (2.13) can be selected arbitrarily by a suitable selection of the feedback matrix $\boldsymbol{L}$.

- **Remark**
  - The remark on Slide 2-35 analogously holds for observability

Detectability

Definition and Analysis

Definition 2.10 The discrete-time linear time-invariant system $(2.3)/(2.4)$ or equivalently the pair $(\boldsymbol{C}, \boldsymbol{A})$ is detectable if there exists a feedback matrix $\boldsymbol{L}$ such that the prediction observer (2.12) is globally asymptotically stable, i.e. if the unstable eigenvalues of (2.3) are observable.

Theorem 2.8 Let $\Lambda = \{\lambda_i(\boldsymbol{A}) \colon |\lambda_i(\boldsymbol{A})| \geq 1\}$ be the set of eigenvalues on or outside the unit disk. Then the pair $(\boldsymbol{C}, \boldsymbol{A})$ is detectable iff $\begin{pmatrix} \boldsymbol{A} - \lambda_i \boldsymbol{I}_{n\times n} \\ \boldsymbol{C} \end{pmatrix}$ has full rank n for all $\lambda_i \in \Lambda$.

- **Remarks**
  - Observability $\Rightarrow$ detectability, but detectability $\not\Rightarrow$ observability
  - Detectability does in particular not guarantee that the eigenvalues of the estimation error (2.13) can be selected arbitrarily by a suitable selection of the feedback matrix $\boldsymbol{L}$

Current Observer

- **Motivation**
  - A prediction observer determines the current estimated state $\hat{\boldsymbol{x}}(k)$ based on the previous measured output $\boldsymbol{y}(k-1)$
  - This can be beneficial if computation time $\tau \approx$ sampling period h
  - This is undesirable if computation time $\tau \ll$ sampling period h
  - Then the current estimated state $\hat{\boldsymbol{x}}(k)$ is not as accurate as it could be
- **Approach**
  - Modify the prediction observer (2.12) to

$$\hat{\boldsymbol{x}}(k) = \underbrace{\boldsymbol{A}\hat{\boldsymbol{x}}(k-1) + \boldsymbol{B}\boldsymbol{u}(k-1)}_{\text{open-loop prediction } \breve{\boldsymbol{x}}(k)} + \boldsymbol{L}_{\mathrm{c}}\left[\boldsymbol{y}(k) - \boldsymbol{C}\big(\underbrace{\boldsymbol{A}\hat{\boldsymbol{x}}(k-1) + \boldsymbol{B}\boldsymbol{u}(k-1)}_{\text{open-loop prediction } \breve{\boldsymbol{x}}(k)}\big)\right] \tag{2.14}$$

  - The estimation error $\tilde{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$ then obeys

$$\tilde{\boldsymbol{x}}(k+1) = (\boldsymbol{A} - \boldsymbol{L}_{\mathrm{c}}\boldsymbol{C}\boldsymbol{A})\tilde{\boldsymbol{x}}(k)$$

Current Observer

- **Approach**
  - If $\boldsymbol{L}_\mathrm{c}$ is chosen such that $\boldsymbol{A} - \boldsymbol{L}_\mathrm{c}\boldsymbol{C}\boldsymbol{A}$ is asymptotically stable, then the estimation error $\tilde{\boldsymbol{x}}(k)$ converges
  - This can be achieved e.g. by pole placement
- **Remarks**
  - The equation (2.14) is called current observer since the state $\hat{\boldsymbol{x}}(k)$ at time k is estimated based on the measured output $\boldsymbol{y}(k)$ at time k
  - The feedback matrix $\boldsymbol{L}_\mathrm{c}$ of the current observer and the feedback matrix $\boldsymbol{L}$ of the prediction observer are related by $\boldsymbol{A}\boldsymbol{L}_\mathrm{c} = \boldsymbol{L}$
- **Implementation for State Feedback Control**
  1. Determine $\hat{\boldsymbol{x}}(k) = \check{\boldsymbol{x}}(k) + \boldsymbol{L}_\mathrm{c}[\boldsymbol{y}(k) - \boldsymbol{C}\check{\boldsymbol{x}}(k)]$ at time k for time k
  2. Determine $\boldsymbol{u}(k) = \boldsymbol{K}\hat{\boldsymbol{x}}(k)$ at time k for time k
  3. Determine $\check{\boldsymbol{x}}(k+1) = \boldsymbol{A}\hat{\boldsymbol{x}}(k) + \boldsymbol{B}\boldsymbol{u}(k)$ at time k for time $k+1$

Reduced-Order Observer

- **Motivation**
  - A measurement of some states is often possible
  - An observation of these states unnecessarily increases the computation time
  - An observation of these states may still be useful for smoothing due to noise
- **Assumption**
  - There is no feedthrough, i.e. $\boldsymbol{D} = \boldsymbol{0}$
- **Approach**
  - Partition the state vector $\boldsymbol{x}(k)$ into a measurable part $\boldsymbol{x}_1(k)$ and a non-measurable part $\boldsymbol{x}_2(k)$
  - The discrete-time LTI system (2.3)/(2.4) then becomes

$$\begin{pmatrix} \boldsymbol{x}_1(k+1) \\ \boldsymbol{x}_2(k+1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_1(k) \\ \boldsymbol{x}_2(k) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{pmatrix} \boldsymbol{u}(k) \tag{2.15}$$

$$\boldsymbol{y}(k) = \begin{pmatrix} \boldsymbol{C}_1 & \boldsymbol{C}_2 \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_1(k) \\ \boldsymbol{x}_2(k) \end{pmatrix} \tag{2.16}$$

Reduced-Order Observer

- **Approach**
  - Reorder the non-measurable part $\boldsymbol{x}_2(k)$ in (2.15) as

$$\boldsymbol{x}_2(k+1) = \boldsymbol{A}_{22}\boldsymbol{x}_2(k) + \underbrace{\boldsymbol{A}_{21}\boldsymbol{x}_1(k) + \boldsymbol{B}_2\boldsymbol{u}(k)}_{\text{known “input”}} \tag{2.18}$$

  - Reorder the measurable part $\boldsymbol{x}_1(k)$ in (2.15) as

$$\underbrace{\boldsymbol{x}_1(k+1) - \boldsymbol{A}_{11}\boldsymbol{x}_1(k) - \boldsymbol{B}_1\boldsymbol{u}(k)}_{\text{known “output”}} = \boldsymbol{A}_{12}\boldsymbol{x}_2(k) \tag{2.19}$$

  - Consider (2.18)/(2.19) as a new discrete-time LTI system by making the following substitutions

$$\boldsymbol{x}(k) \leftarrow \boldsymbol{x}_2(k) \qquad \boldsymbol{y}(k) \leftarrow \boldsymbol{x}_1(k+1) - \boldsymbol{A}_{11}\boldsymbol{x}_1(k) - \boldsymbol{B}_1\boldsymbol{u}(k)$$

$$\boldsymbol{A} \leftarrow \boldsymbol{A}_{22} \qquad \boldsymbol{B}\boldsymbol{u}(k) \leftarrow \boldsymbol{A}_{21}\boldsymbol{x}_1(k) + \boldsymbol{B}_2\boldsymbol{u}(k) \qquad \boldsymbol{C} \leftarrow \boldsymbol{A}_{12}$$

    in the prediction observer (2.12)

Reduced-Order Observer

- **Approach**
  - This leads to a reduced-order observer

    Not available at time k!
    Regard during implementation (see below)
    or substitute $\hat{\boldsymbol{x}}_2(k) = \hat{\boldsymbol{x}}_2'(k) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k)$ (see Slide 2-62)

    $$\hat{\boldsymbol{x}}_2(k+1) = \boldsymbol{A}_{22}\hat{\boldsymbol{x}}_2(k) + \boldsymbol{A}_{21}\boldsymbol{x}_1(k) + \boldsymbol{B}_2\boldsymbol{u}(k) + \boldsymbol{L}_\mathrm{r}[\boldsymbol{x}_1(k+1) - \boldsymbol{A}_{11}\boldsymbol{x}_1(k) - \boldsymbol{B}_1\boldsymbol{u}(k) - \boldsymbol{A}_{12}\hat{\boldsymbol{x}}_2(k)]$$

  - The estimation error $\tilde{\boldsymbol{x}}_2(k) = \boldsymbol{x}_2(k) - \hat{\boldsymbol{x}}_2(k)$ then obeys

    $$\tilde{\boldsymbol{x}}_2(k+1) = (\boldsymbol{A}_{22} - \boldsymbol{L}_\mathrm{r}\boldsymbol{A}_{12})\tilde{\boldsymbol{x}}_2(k)$$

  - If $\boldsymbol{L}_\mathrm{r}$ is chosen such that $\boldsymbol{A}_{22} - \boldsymbol{L}_\mathrm{r}\boldsymbol{A}_{12}$ is asymptotically stable, then the estimation error $\tilde{\boldsymbol{x}}(k)$ converges
  - This can be achieved e.g. by pole placement

- **Implementation for State Feedback Control**
  1. Determine $\breve{\boldsymbol{x}}_2(k) \coloneqq \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k) + \breve{\boldsymbol{x}}_2(k)$
  2. Determine $\boldsymbol{u}(k) = \boldsymbol{K}\hat{\boldsymbol{x}}(k) = (\boldsymbol{K}_1 \quad \boldsymbol{K}_2)\begin{pmatrix}\boldsymbol{x}_1(k) \\ \breve{\boldsymbol{x}}_2(k)\end{pmatrix}$
  3. Determine $\breve{\boldsymbol{x}}_2(k+1) = \boldsymbol{A}_{22}\hat{\boldsymbol{x}}_2(k) + \boldsymbol{A}_{21}\boldsymbol{x}_1(k) + \boldsymbol{B}_2\boldsymbol{u}(k) + \boldsymbol{L}_\mathrm{r}[-\boldsymbol{A}_{11}\boldsymbol{x}_1(k) - \boldsymbol{B}_1\boldsymbol{u}(k) - \boldsymbol{A}_{12}\hat{\boldsymbol{x}}_2(k)]$

Combination of Controller and Observer

Separation Theorem

- **What happens if state feedback control is realized with the estimated state?**
  - Consider the state feedback control law (2.10) using the estimated state $\hat{\boldsymbol{x}}(k)$, i.e.

    $$\boldsymbol{u}(k) = \boldsymbol{K}\hat{\boldsymbol{x}}(k)$$

  - The closed-loop system then becomes

    $$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{K}\hat{\boldsymbol{x}}(k)$$

    which can also be written in terms of the estimation error $\tilde{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$ as

    $$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{K}\big(\boldsymbol{x}(k) - \tilde{\boldsymbol{x}}(k)\big)$$

  - Combining with (2.13) yields an augmented state equation of the overall system, i.e.

    $$\begin{pmatrix} \tilde{\boldsymbol{x}}(k+1) \\ \boldsymbol{x}(k+1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} & \boldsymbol{0}_{n\times n} \\ -\boldsymbol{B}\boldsymbol{K} & \boldsymbol{A} + \boldsymbol{B}\boldsymbol{K} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(k) \\ \boldsymbol{x}(k) \end{pmatrix}$$

Combination of Controller and Observer

Separation Theorem

- **What happens if state feedback control is realized with the estimated state?**
  - The characteristic equation of the overall system results as

$$\det\begin{pmatrix} z\boldsymbol{I}_{n\times n} - \boldsymbol{A} + \boldsymbol{LC} & \boldsymbol{0}_{n\times n} \\ \boldsymbol{BK} & z\boldsymbol{I}_{n\times n} - \boldsymbol{A} - \boldsymbol{BK} \end{pmatrix} = 0$$

which due to the zero matrix can also be written as

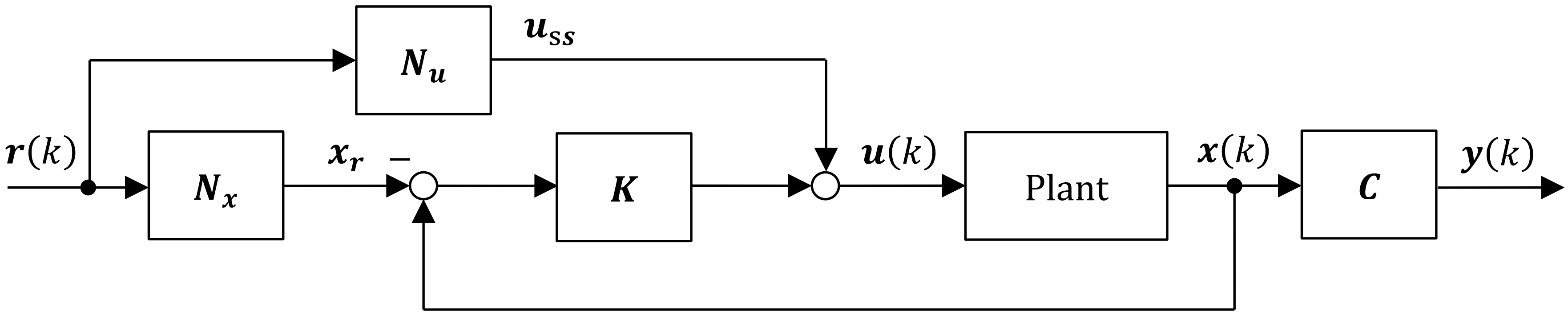
$$\det(z\boldsymbol{I}_{n\times n} - \boldsymbol{A} + \boldsymbol{LC})\det(z\boldsymbol{I}_{n\times n} - \boldsymbol{A} - \boldsymbol{BK}) = 0$$

Theorem 2.9 The poles of a state feedback control realized with an estimated state consist of the poles of $\boldsymbol{A} + \boldsymbol{BK}$, i.e. the poles of the state feedback control without observer, and the poles of $\boldsymbol{A} - \boldsymbol{LC}$, i.e. the poles of the observer.

- **Remarks**
  - Theorem 2.9 implies that the state feedback control and the observer can be designed separately
  - Theorem 2.9 has been derived for the prediction observer, but also applies for the other observers

Reference Tracking based on State Feedback Control

- **Motivation**
  - Many control problems require a tracking of the reference input $\boldsymbol{r}(k)$
- **Approach**
  - Consider the so-called state-command structure



  - The state command matrix $\boldsymbol{N_x}$ defines the desired state $\boldsymbol{x_r}$
  - The feedforward matrix $\boldsymbol{N_u}$ provides a steady-state control input $\boldsymbol{u}_{ss}$ for eliminating a steady-state error for plants without integral action

Reference Tracking based on State Feedback Control

- **Approach**
  - Assume that the number of inputs m is equal to the number of outputs p. Only in this case a unique and exact solution exists. Otherwise an approximate solution must be constructed (not considered).
  - Require that the output $\boldsymbol{y}(k)$ is equal to the reference input $\boldsymbol{r}(k)$ in steady state $\boldsymbol{x}_{ss}$, i.e.

$$\boldsymbol{N}_{\boldsymbol{x}}\boldsymbol{r}(k) = \boldsymbol{x}_{\boldsymbol{r}} = \boldsymbol{x}_{ss} \tag{2.20}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}_{ss} = \boldsymbol{r}(k) \tag{2.21}$$

  - Substituting (2.20) into (2.21) leads to

$$\boldsymbol{C}\boldsymbol{N}_{\boldsymbol{x}}\boldsymbol{r}(k) = \boldsymbol{r}(k) \Leftrightarrow \boldsymbol{C}\boldsymbol{N}_{\boldsymbol{x}} = \boldsymbol{I}_{p\times p} \tag{2.22}$$

  - Consider that the plant is in steady state $\boldsymbol{x}_{ss}$, i.e.

$$\boldsymbol{x}_{ss} = \boldsymbol{A}\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss} \Leftrightarrow (\boldsymbol{A} - \boldsymbol{I}_{n\times n})\boldsymbol{x}_{ss} + \boldsymbol{B}\boldsymbol{u}_{ss} = \boldsymbol{0}_{n\times 1} \tag{2.23}$$

  - Substituting (2.20) and $\boldsymbol{u}_{ss} = \boldsymbol{N}_{\boldsymbol{u}}\boldsymbol{r}(k)$ into (2.23) results in

$$(\boldsymbol{A} - \boldsymbol{I}_{n\times n})\boldsymbol{N}_{\boldsymbol{x}}\boldsymbol{r}(k) + \boldsymbol{B}\boldsymbol{N}_{\boldsymbol{u}}\boldsymbol{r}(k) = \boldsymbol{0}_{n\times p} \Leftrightarrow (\boldsymbol{A} - \boldsymbol{I}_{n\times n})\boldsymbol{N}_{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{N}_{\boldsymbol{u}} = \boldsymbol{0}_{n\times p} \tag{2.24}$$

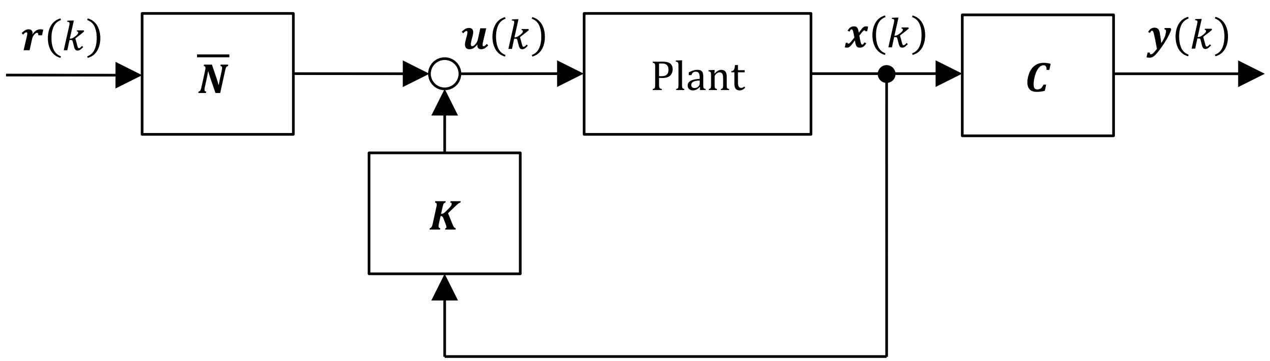
Reference Tracking

Reference Tracking based on State Feedback Control

- **Approach**
  - Combining (2.22) and (2.24) finally leads to

$$\begin{pmatrix} \boldsymbol{A}-\boldsymbol{I}_{n\times n} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{0}_{p\times p} \end{pmatrix} \begin{pmatrix} \boldsymbol{N_x} \\ \boldsymbol{N_u} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0}_{n\times p} \\ \boldsymbol{I}_{p\times p} \end{pmatrix} \Leftrightarrow \begin{pmatrix} \boldsymbol{N_x} \\ \boldsymbol{N_u} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}-\boldsymbol{I}_{n\times n} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{0}_{p\times p} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{0}_{n\times p} \\ \boldsymbol{I}_{p\times p} \end{pmatrix}$$

- **Remarks**
  - The matrices $\boldsymbol{N_x}$ and $\boldsymbol{N_u}$ are sometimes combined to $\overline{\boldsymbol{N}} = -\boldsymbol{K}\boldsymbol{N_x} + \boldsymbol{N_u}$, yielding the structure



Reference Tracking based on State Feedback Control

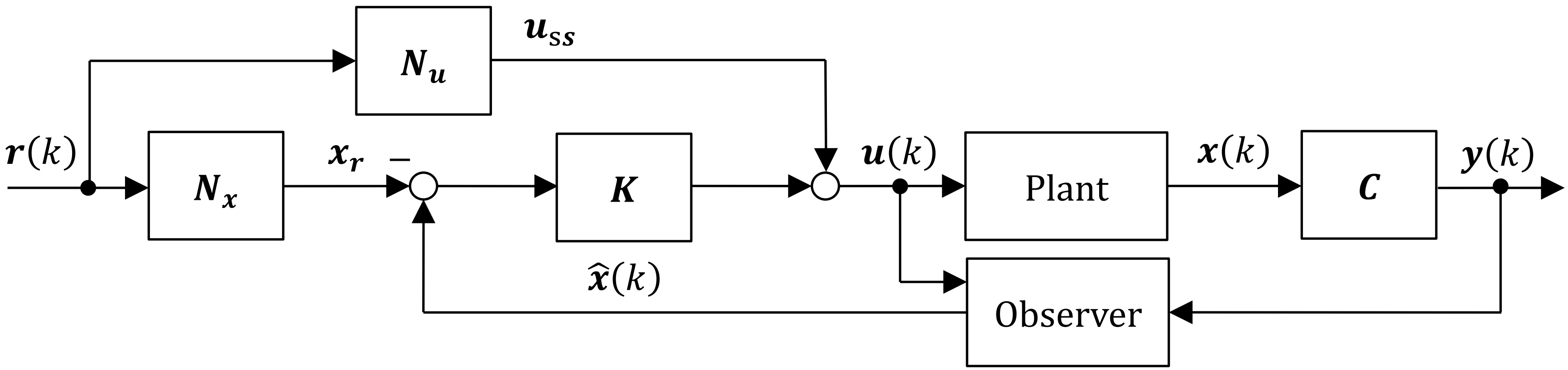
- **Remarks**
  - The advantage of this structure is a low computation time, the disadvantage is a high sensitivity w.r.t. computation errors in the feedback matrix $\boldsymbol{K}$
  - Note that the measured output $\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k)$ with $\boldsymbol{C} \in \mathbb{R}^{p \times n}$ used for the observer and the controlled output $\boldsymbol{y_r}(k) = \boldsymbol{C_r}\boldsymbol{x}(k)$ with $\boldsymbol{C_r} \in \mathbb{R}^{p_r \times n}$ considered for reference tracking are sometimes different. Reference tracking is then formulated w.r.t. $\boldsymbol{y_r}(k)$.

Reference Tracking based on Output Feedback Control

- **Approach**
  - Reference tracking can be realized analogously with an observer
  - The input vector $\boldsymbol{u}(k)$ to the plant and to the observer must, however, be equal, i.e.

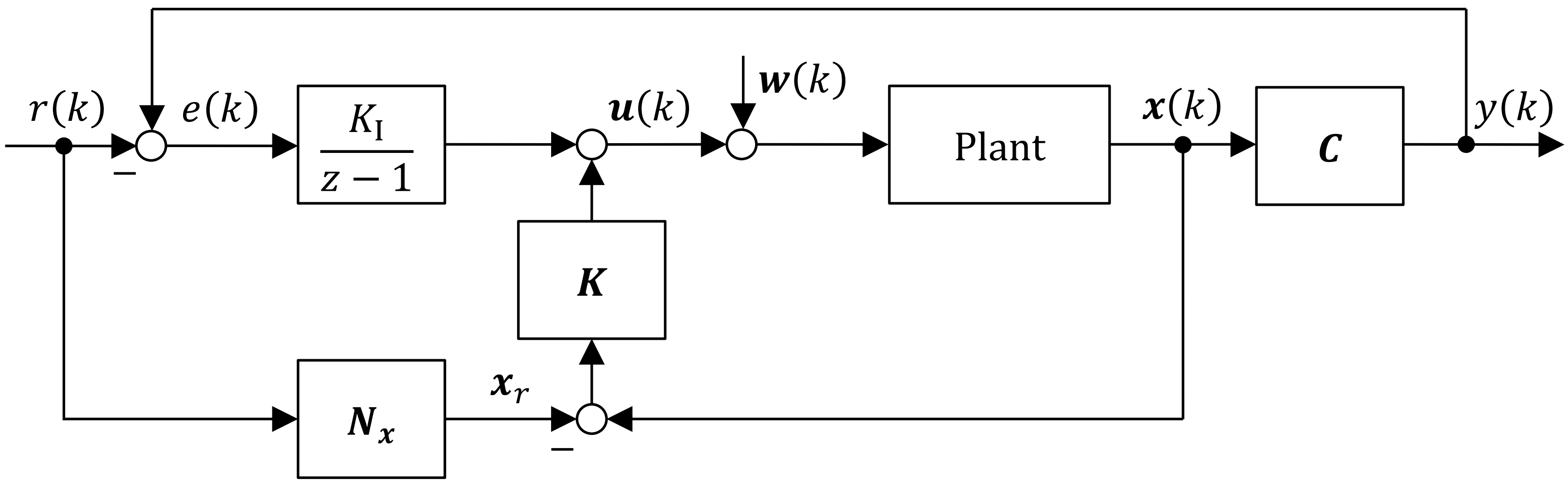
$$\boldsymbol{u}(k) = \boldsymbol{K}(\hat{\boldsymbol{x}}(k) - \boldsymbol{x}_r) + \boldsymbol{N}_u \boldsymbol{r}(k) = \boldsymbol{K}\hat{\boldsymbol{x}}(k) + \overline{\boldsymbol{N}}\boldsymbol{r}(k)$$ for prediction and current observer

$$\boldsymbol{u}(k) = (\boldsymbol{K}_1 \quad \boldsymbol{K}_2)\left(\begin{pmatrix}\boldsymbol{x}_1(k)\\ \hat{\boldsymbol{x}}_2(k)\end{pmatrix} - \boldsymbol{x}_r\right) + \boldsymbol{N}_u \boldsymbol{r}(k) = (\boldsymbol{K}_1 \quad \boldsymbol{K}_2)\begin{pmatrix}\boldsymbol{x}_1(k)\\ \hat{\boldsymbol{x}}_2(k)\end{pmatrix} + \overline{\boldsymbol{N}}\boldsymbol{r}(k)$$ for red.-ord. observer



Integral Control

- **Motivation**
  - Integral control is useful for eliminating steady-state errors due to constant disturbances or reference inputs and for automatically providing a setpoint for the control input
  - Integral control is, different from feedforward control, robust w.r.t. uncertainties
- **Approach**



Integral Control

- **Approach**
  - Augment the discrete-time LTI state equation (2.3) by the state $x_\mathrm{I}(k)$ obeying the state equation

    $$x_\mathrm{I}(k+1) = x_\mathrm{I}(k) + e(k) = x_\mathrm{I}(k) + \boldsymbol{C}\boldsymbol{x}(k) - r(k)$$

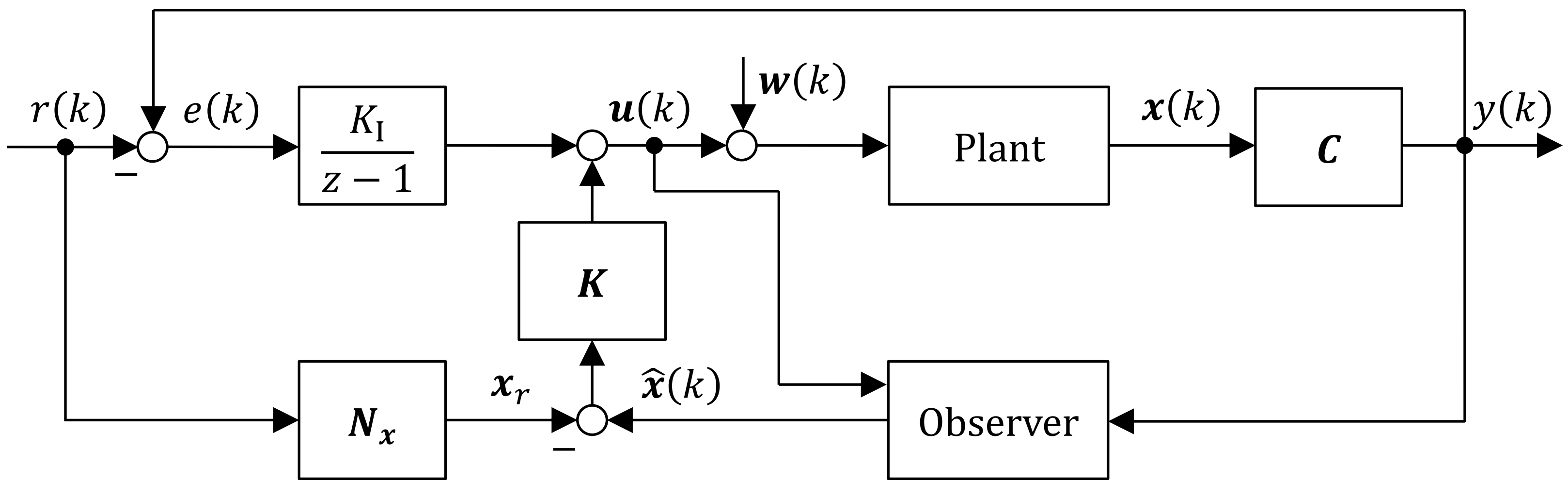
    to obtain the augmented state equation

    $$\begin{pmatrix} x_\mathrm{I}(k+1) \\ \boldsymbol{x}(k+1) \end{pmatrix} = \begin{pmatrix} 1 & \boldsymbol{C} \\ \boldsymbol{0} & \boldsymbol{A} \end{pmatrix} \begin{pmatrix} x_\mathrm{I}(k) \\ \boldsymbol{x}(k) \end{pmatrix} + \begin{pmatrix} 0 \\ \boldsymbol{B} \end{pmatrix} u(k) - \begin{pmatrix} 1 \\ \boldsymbol{0} \end{pmatrix} r(k)$$
  - The state feedback control law then becomes

    $$u(k) = \begin{pmatrix} K_\mathrm{I} & \boldsymbol{K} \end{pmatrix} \begin{pmatrix} x_\mathrm{I}(k) \\ \boldsymbol{x}(k) \end{pmatrix} - \boldsymbol{K}\boldsymbol{N}_{\boldsymbol{x}} r(k)$$
  - The state feedback control law can be designed based on methods developed on the previous slides
- **Remarks**
  - Integral control has been derived here for SISO systems, but can be easily extended to MIMO systems

Integral Control

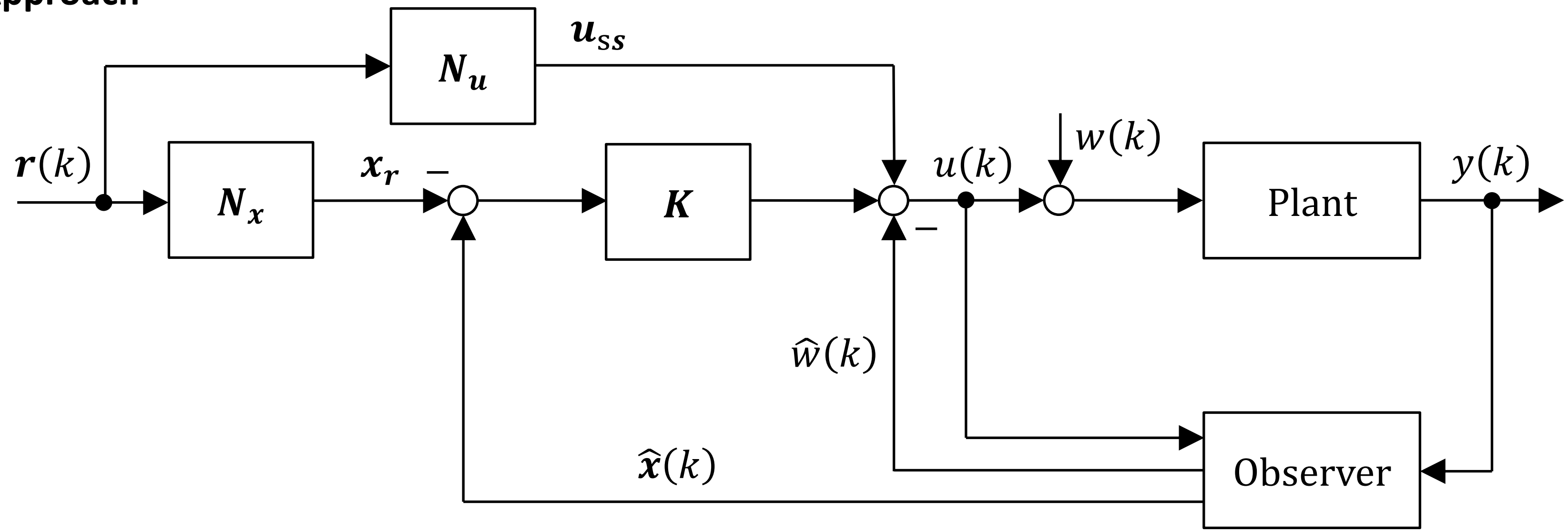
- **Remarks**
  - The state command matrix $\boldsymbol{N}_x$ is often retained to react to references inputs rapidly and restrict integral control to disturbance rejection and uncertainty handling
  - Integral control can also be combined with an observer. This observer is based on the original discrete-time LTI system $(2.3)/(2.4)$ since only the state $\boldsymbol{x}(k)$ must be estimated.



Disturbance Rejection

Disturbance Estimation

- **Motivation**
  - Sometimes the disturbance $w(k)$ can be modeled
  - The disturbance $w(k)$ can then be estimated by an observer, included in the control input $u(k)$, and compensated in this way
- **Approach**



Disturbance Estimation

- **Approach**
  - Introduce a disturbance model, e.g.

$\dot{w}(t) = 0$	constant disturbance
$\ddot{w}(t) + \omega^2 w(t) = 0$	sinusoidal disturbance with frequency $f = {}^{\omega}/_{2\pi}$
$\dot{\boldsymbol{x}}_w(t) = \boldsymbol{A}_{wc}\boldsymbol{x}_w(t), w(t) = \boldsymbol{C}_{wc}\boldsymbol{x}_w(t)$	general disturbance

  - Discretize the disturbance model using ZOH, i.e.

$$\boldsymbol{x}_w(k+1) = \boldsymbol{A}_w\boldsymbol{x}_w(k), w(k) = \boldsymbol{C}_w\boldsymbol{x}_w(k) \text{ with } \boldsymbol{A}_w = \mathrm{e}^{\boldsymbol{A}_{wc}h}, \boldsymbol{C}_{wc} = \boldsymbol{C}_w$$

  - Augment the discrete-time LTI system $(2.3)/(2.4)$ by discrete-time disturbance model, i.e.

$$\begin{pmatrix} \boldsymbol{x}(k+1) \\ \boldsymbol{x}_w(k+1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}\boldsymbol{C}_w \\ \boldsymbol{0} & \boldsymbol{A}_w \end{pmatrix} \begin{pmatrix} \boldsymbol{x}(k) \\ \boldsymbol{x}_w(k) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{pmatrix} u(k) \tag{2.25}$$

$$y(k) = \begin{pmatrix} \boldsymbol{C} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}(k) \\ \boldsymbol{x}_w(k) \end{pmatrix} \tag{2.26}$$

Disturbance Estimation

- **Approach**
  - Note that for a constant disturbance the augmented discrete-time system $(2.25)/(2.26)$ reduces to

$$\begin{pmatrix} \boldsymbol{x}(k+1) \\ w(k+1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{x}(k) \\ w(k) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} \\ 0 \end{pmatrix} u(k)$$

$$y(k) = \begin{pmatrix} \boldsymbol{C} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{x}(k) \\ w(k) \end{pmatrix}$$

  - Design an observer for the augmented discrete-time system $(2.25)/(2.26)$ based on the methods developed on the previous slides
- **Remarks**
  - The state feedback control law must still be designed for the discrete-time LTI system $(2.3)/(2.4)$ since the disturbance can not be controlled
  - Disturbance estimation has been derived here for SISO systems, but can be easily extended to MIMO systems

Literature

Discrete-Time Systems

[Fra07] Gene F. Franklin. Rational Rate. *IEEE Control Systems Magazine*, 27(4):19–19, 2007.

[GHJ+14] Lars Grüne, Sandra Hirche, Oliver Junge, Péter Koltai, Daniel Lehmann, Jan Lunze, Adam Molin, Rudolf Sailer, Manuela Sigurani, Christian Stöcker, and Fabian Wirth. Event-based control. In Jan Lunze, editor, *Control Theory of Digitally Networked Dynamic Systems*, chapter 5, pages 169–261. Springer, Cham, 2014.

[HNX07] João P. Hespanha, Payam Naghshtabrizi, and Yonggang Xu. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.

Matrix Computations

[GV96] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996. – MAT Golub

[HJ85] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985. – MAT Horn

Literature

Matrix Computations

[Mey00] Carl Dean Meyer. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000. – www.matrixanalysis.com

[MV78] Cleve Moler and Charles F. Van Loan. Nineteen dubious ways to compute the exponential of a matrix. *SIAM Review*, 20(4):801–836, 1978.

[Van78] Charles F. Van Loan. Computing integrals involving the matrix exponential. *IEEE Transactions on Automatic Control*, 23(3):395–404, 1978.

Miscellaneous

[Eng04] Hans-Georg Engel. *Kraftfahrzeuge II*. Lecture Notes, Department of Mechanical and Process Engineering, University of Kaiserslautern, 2004.

[PSD+08] C. Poussot-Vassal, O. Sename, L. Dugard, P. Gáspár, Z. Szabó, and J. Bokor. A new semi-active suspension control strategy through LPV technique. *Control Engineering Practice*, 16(12):1519–1534, 2008.

Reduced-Order Observer

- **Substitution**
  - Substituting $\hat{\boldsymbol{x}}_2(k) = \hat{\boldsymbol{x}}_2'(k) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k)$ into the reduced-order observer yields

$$\hat{\boldsymbol{x}}_2'(k+1) + \cancel{\boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k+1)} = \boldsymbol{A}_{22}\left(\hat{\boldsymbol{x}}_2'(k) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k)\right) + \boldsymbol{A}_{21}\boldsymbol{x}_1(k) + \boldsymbol{B}_2\boldsymbol{u}(k) + \cancel{\boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k+1)}$$
$$+\boldsymbol{L}_\mathrm{r}\left[-\boldsymbol{A}_{11}\boldsymbol{x}_1(k) - \boldsymbol{B}_1\boldsymbol{u}(k) - \boldsymbol{A}_{12}\left(\hat{\boldsymbol{x}}_2'(k) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k)\right)\right]$$

  or equivalently

$$\hat{\boldsymbol{x}}_2'(k+1) = (\boldsymbol{A}_{22} - \boldsymbol{L}_r\boldsymbol{A}_{11})\hat{\boldsymbol{x}}_2'(k) + (\boldsymbol{A}_{22}\boldsymbol{L}_r - \boldsymbol{A}_{21} - \boldsymbol{L}_r\boldsymbol{A}_{11} - \boldsymbol{L}_\mathrm{r}\boldsymbol{A}_{12}\boldsymbol{L}_\mathrm{r})\boldsymbol{x}_1(k) + (\boldsymbol{B}_2 - \boldsymbol{B}_1)\boldsymbol{u}(k)$$

  - The estimated state then results from $\hat{\boldsymbol{x}}_2(k) = \hat{\boldsymbol{x}}_2'(k) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(k)$
- **Remark**
  - The initial estimated state results from $\hat{\boldsymbol{x}}_2(0) = \hat{\boldsymbol{x}}_2'(0) + \boldsymbol{L}_\mathrm{r}\boldsymbol{x}_1(0)$
  - The feedback matrix $\boldsymbol{L}_\mathrm{r}$ should therefore not be chosen too large
  - Otherwise the initial estimation error $\tilde{\boldsymbol{x}}_2(0)$ may be very large