ELEC4410

Control System Design

Lecture 19: Feedback from Estimated States and Discrete-Time Control Design

Julio H. Braslavsky

julio@ee.newcastle.edu.au

School of Electrical Engineering and Computer Science
The University of Newcastle

Outline

- Feedback from Estimated States
- Discrete-Time Control Design
- Dead-Beat Control

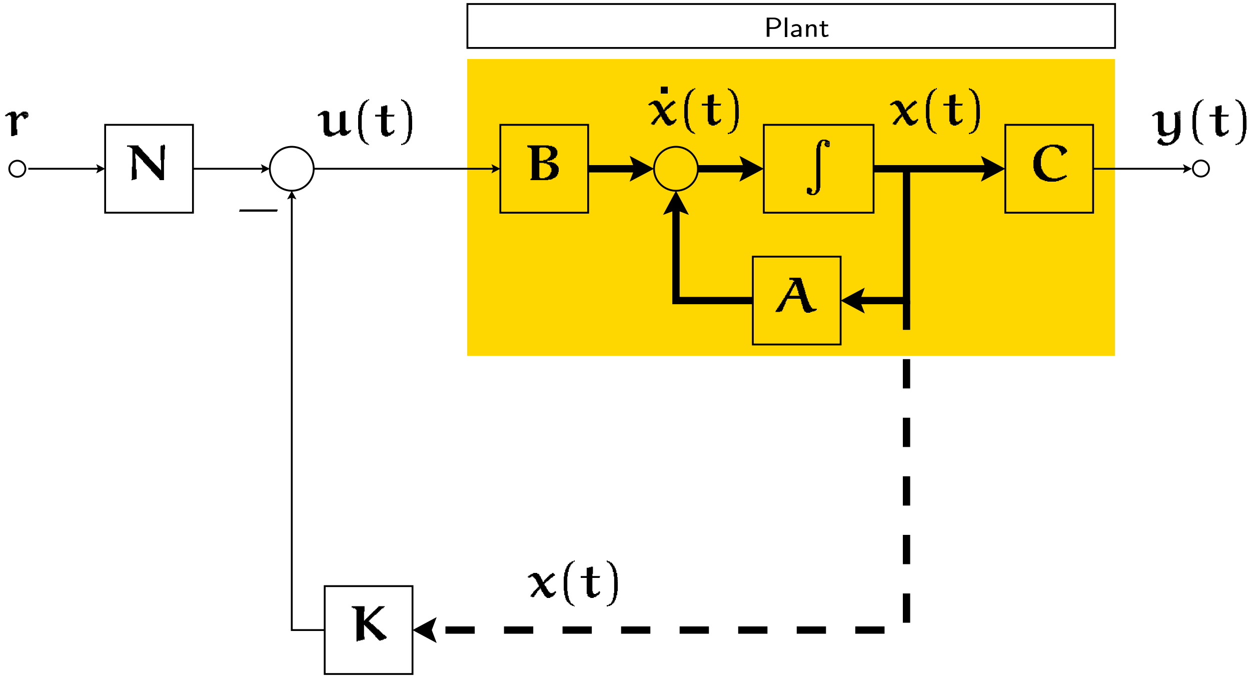
Feedback from Estimated States

In the last lecture we discussed the general scheme for feedback control from estimated states, involving

Feedback from Estimated States

In the last lecture we discussed the general scheme for feedback control from estimated states, involving

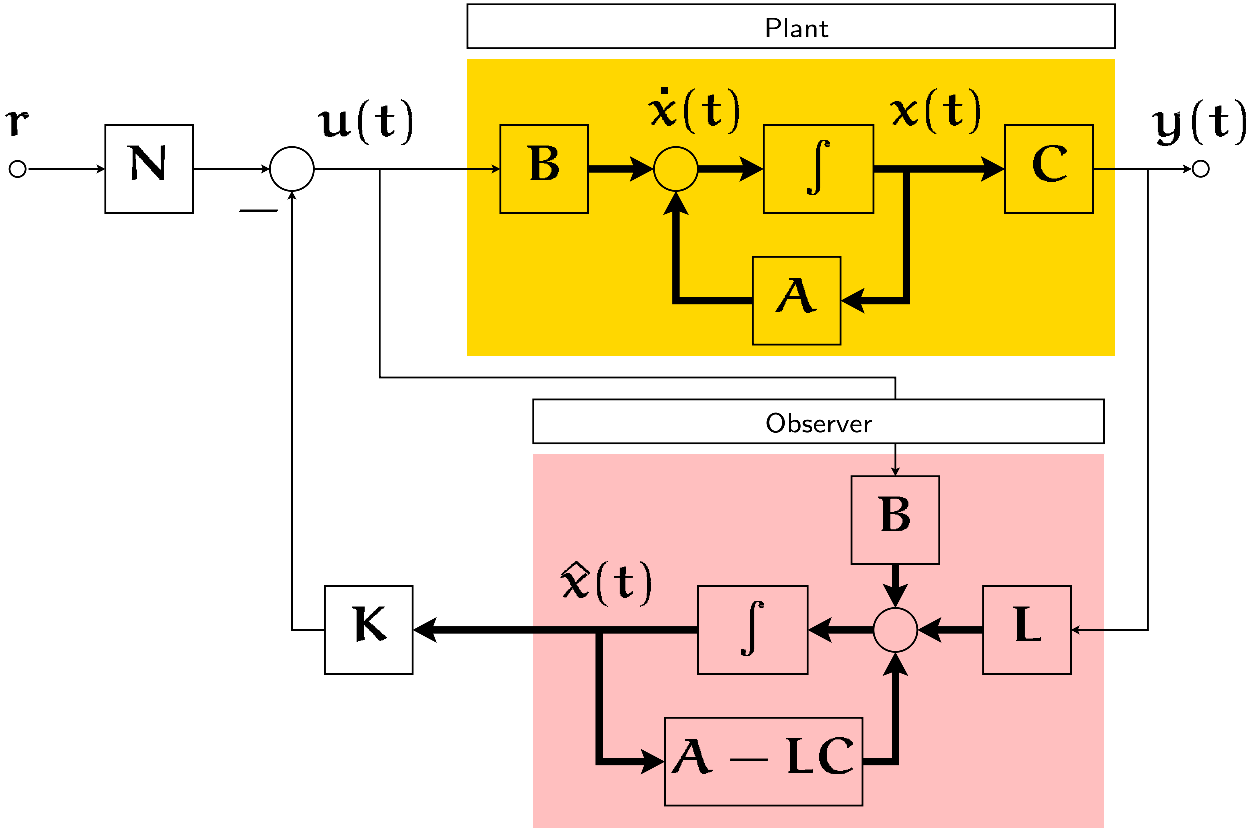
- a **state feedback gain** K



Feedback from Estimated States

In the last lecture we discussed the general scheme for feedback control from estimated states, involving

- a **state feedback gain** $\mathbf{K}$
- a **state estimator** (observer), with gain $\mathbf{L}$



Feedback from Estimated States

In the last lecture we discussed the general scheme for feedback control from estimated states, involving

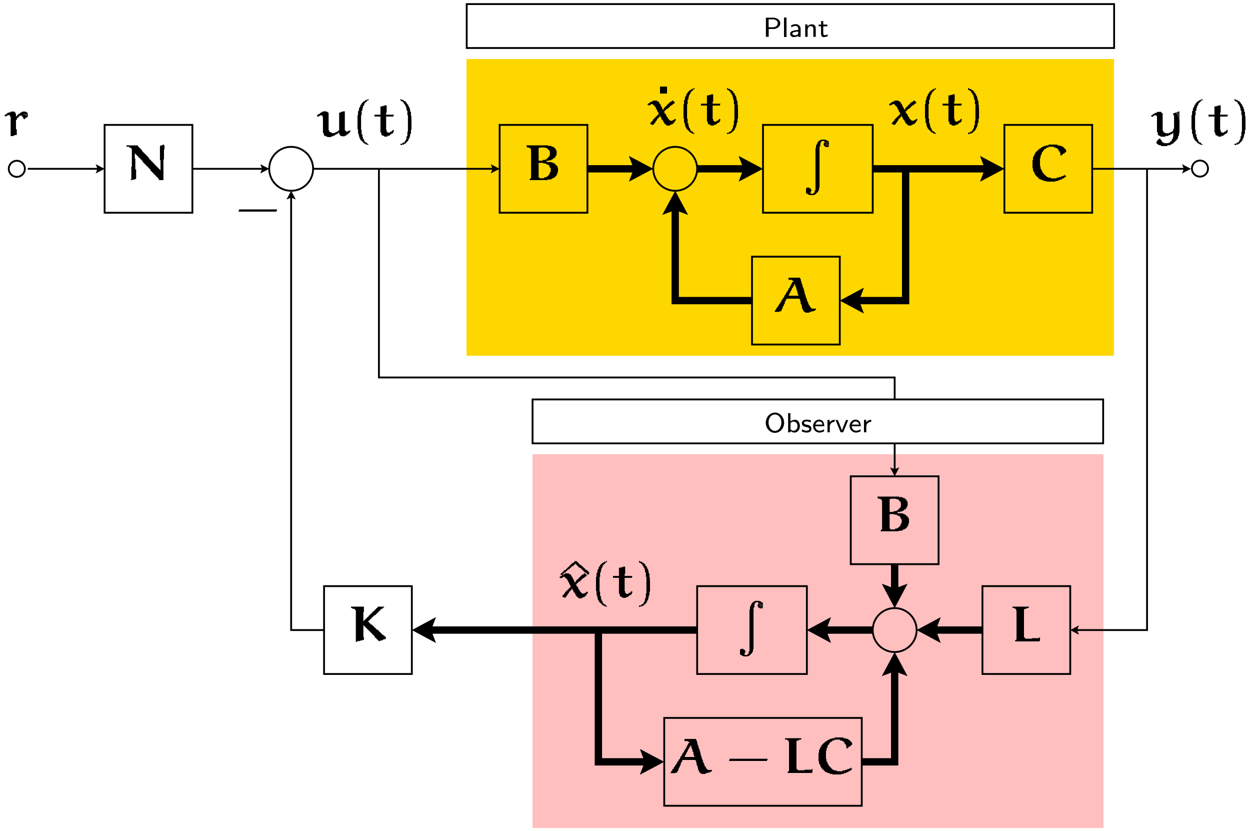
- a **state feedback gain** $\mathbf{K}$
- a **state estimator** (observer), with gain $\mathbf{L}$

To design $\mathbf{K}$ we used the **Bass-Gura** formula to make

$$(\mathbf{A} - \mathbf{BK}): \quad \text{Hurwitz}$$

To design $\mathbf{L}$ in the observer, we used **duality** and the same **Bass-Gura** formula to make

$$(\mathbf{A} - \mathbf{LC}): \quad \text{Hurwitz}$$



Feedback from Estimated States

- By designing $\mathbf{K}$ such that $(\boldsymbol{A} - \mathbf{B}\mathbf{K})$ is Hurwitz, with the desired eigenvalues, we can guarantee that the closed-loop system will be asymptotically and BIBO stable, and will have the specified dynamic response.

Feedback from Estimated States

- By designing $\mathbf{K}$ such that $(\mathbf{A} - \mathbf{BK})$ is Hurwitz, with the desired eigenvalues, we can guarantee that the closed-loop system will be asymptotically and BIBO stable, and will have the specified dynamic response.
- By designing $\mathbf{L}$ such that $(\mathbf{A} - \mathbf{LC})$ is Hurwitz, we guarantee that the observer will be asymptotically stable, and the estimate of the states $\hat{x}(t)$ will converge to the real states $x(t)$ as $t \to \infty$.

Feedback from Estimated States

- By designing $\mathbf{K}$ such that $(\mathbf{A}-\mathbf{BK})$ is Hurwitz, with the desired eigenvalues, we can guarantee that the closed-loop system will be asymptotically and BIBO stable, and will have the specified dynamic response.
- By designing $\mathbf{L}$ such that $(\mathbf{A}-\mathbf{LC})$ is Hurwitz, we guarantee that the observer will be asymptotically stable, and the estimate of the states $\hat{x}(t)$ will converge to the real states $x(t)$ as $t \to \infty$.
- But $\mathbf{K}$ and the observer are designed independently . . . Will they work the same when we put them together in a **feedback from estimated states** scheme?

Feedback from Estimated States

Three basic questions arise regarding **feedback from estimated states:**

- The closed-loop eigenvalues were set as those of $\boxed{(\mathbf{A}-\mathbf{B}\mathbf{K})}$ by using **state feedback**

$$\mathbf{u} = -\mathbf{K}x.$$

Would we still have the same eigenvalues if we do **feedback from estimated states**

$$\mathbf{u} = -\mathbf{K}\hat{x}\,?$$

Feedback from Estimated States

Three basic questions arise regarding **feedback from estimated states:**

- The closed-loop eigenvalues were set as those of $\boxed{(\mathbf{A}-\mathbf{B}\mathbf{K})}$ by using **state feedback**

$$\mathbf{u} = -\mathbf{K}x.$$

Would we still have the same eigenvalues if we do **feedback from estimated states**

$$\mathbf{u} = -\mathbf{K}\hat{x}\,?$$

- Would the interconnection affect the **observer** eigenvalues, those of $\boxed{(\mathbf{A}-\mathbf{L}\mathbf{C})}$?

Feedback from Estimated States

Three basic questions arise regarding **feedback from estimated states:**

- The closed-loop eigenvalues were set as those of $\boxed{(\mathbf{A}-\mathbf{BK})}$ by using **state feedback**

  $$\mathbf{u}=-\mathbf{K}x.$$

  Would we still have the same eigenvalues if we do **feedback from estimated states**

  $$\mathbf{u}=-\mathbf{K}\hat{x}\,?$$

- Would the interconnection affect the **observer** eigenvalues, those of $\boxed{(\mathbf{A}-\mathbf{LC})}$?
- What would be the effect of the observer in the **closed-loop transfer function**?

Feedback from Estimated States

To answer these questions we take a look at the state equations of the full system, putting together plant and observer, that is,

$$\begin{aligned}
\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{u} && \text{plant}\\
\dot{\hat{\mathbf{x}}} &= (\mathbf{A}-\mathbf{L}\mathbf{C})\hat{\mathbf{x}}+\mathbf{L}\mathbf{C}\mathbf{x}+\mathbf{B}\mathbf{u} && \text{observer}
\end{aligned}$$

Feedback from Estimated States

To answer these questions we take a look at the state equations of the full system, putting together plant and observer, that is,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{K}\hat{\mathbf{x}} + \mathbf{B}\mathbf{N}\mathbf{r} && \text{plant} \\ \dot{\hat{\mathbf{x}}} &= (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}} + \mathbf{L}\mathbf{C}\mathbf{x} - \mathbf{B}\mathbf{K}\hat{\mathbf{x}} + \mathbf{B}\mathbf{N}\mathbf{r} && \text{observer} \end{aligned}$$

after replacing $\mathbf{u} = \mathbf{N}\mathbf{r} - \mathbf{K}\hat{\mathbf{x}}$.

Feedback from Estimated States

To answer these questions we take a look at the state equations of the full system, putting together plant and observer, that is,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{K}\hat{\mathbf{x}} + \mathbf{B}\mathbf{N}\mathbf{r} \qquad \text{plant}$$

$$\dot{\hat{\mathbf{x}}} = (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}} + \mathbf{L}\mathbf{C}\mathbf{x} - \mathbf{B}\mathbf{K}\hat{\mathbf{x}} + \mathbf{B}\mathbf{N}\mathbf{r} \qquad \text{observer}$$

after replacing $\mathbf{u} = \mathbf{N}\mathbf{r} - \mathbf{K}\hat{\mathbf{x}}$. Packaging these equations in a more compact form we have

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\hat{\mathbf{x}}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}\mathbf{K} \\ \mathbf{L}\mathbf{C} & (\mathbf{A} - \mathbf{L}\mathbf{C} - \mathbf{B}\mathbf{K}) \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{B}\mathbf{N} \\ \mathbf{B}\mathbf{N} \end{bmatrix} \mathbf{r}$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix}$$

Feedback from Estimated States

Let's make a change of coordinates, so that the new coordinates are **the plant state** x and **the estimation error** $\varepsilon = x - \hat{x}$,

$$\begin{bmatrix} x \\ \varepsilon \end{bmatrix} = \begin{bmatrix} x \\ x - \hat{x} \end{bmatrix} = \underbrace{\begin{bmatrix} I & 0 \\ I & -I \end{bmatrix}}_{P} \begin{bmatrix} x \\ \hat{x} \end{bmatrix}$$

Note that $P^{-1} = \left[\begin{smallmatrix} I & 0 \\ I & -I \end{smallmatrix}\right]^{-1} = \left[\begin{smallmatrix} I & 0 \\ I & -I \end{smallmatrix}\right] = P$.

Feedback from Estimated States

Let's make a change of coordinates, so that the new coordinates are **the plant state** x and **the estimation error** $\varepsilon = x - \hat{x}$,

$$\begin{bmatrix} x \\ \varepsilon \end{bmatrix} = \begin{bmatrix} x \\ x - \hat{x} \end{bmatrix} = \underbrace{\begin{bmatrix} I & 0 \\ I & -I \end{bmatrix}}_{P} \begin{bmatrix} x \\ \hat{x} \end{bmatrix}$$

Note that $P^{-1} = \left[\begin{smallmatrix} I & 0 \\ I & -I \end{smallmatrix}\right]^{-1} = \left[\begin{smallmatrix} I & 0 \\ I & -I \end{smallmatrix}\right] = P$. With this equivalence transformation we get the matrices in the new coordinates as

$$\bar{A}_{KL} = PA_{KL}P^{-1} = \begin{bmatrix} (A - BK) & BK \\ 0 & (A - LC) \end{bmatrix}, \quad \bar{B}_{KL} = PB_{KL} = \begin{bmatrix} BN \\ 0 \end{bmatrix}$$

$$\bar{C}_{KL} = C_{KL}P^{-1} = \begin{bmatrix} C & 0 \end{bmatrix}$$

Feedback from Estimated States

The full system in the new coordinates is thus represented as

$$\begin{bmatrix} \dot{x} \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} (A - BK) & BK \\ 0 & (A - LC) \end{bmatrix} \begin{bmatrix} x \\ \varepsilon \end{bmatrix} + \begin{bmatrix} BN \\ 0 \end{bmatrix} r$$

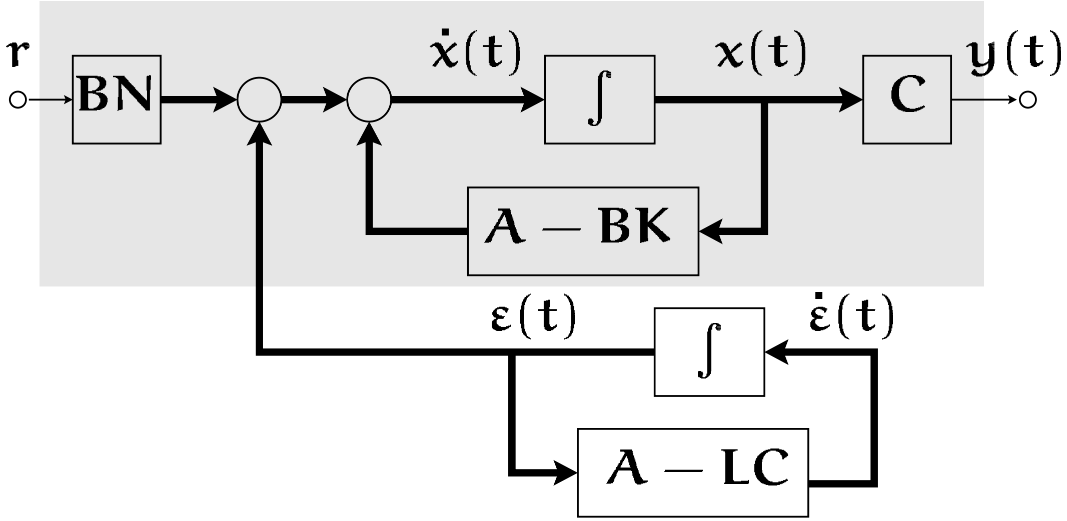
$$y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ \varepsilon \end{bmatrix}$$

Because $\bar{A}_{KL}$ is **triangular**, its eigenvalues are the union of those of $(A - BK)$ and $(A - LC)$.

Controller and observer **do not affect each other** in the interconnection.

Feedback from Estimated States

Note that the estimation error is **uncontrollable**, hence the observer eigenvalues will not appear in the closed loop transfer function.



Feedback from Estimated States

The property of independence between control and state estimation is called the **Separation Principle**

Separation Principle: The design of the state feedback and the design of the state estimator can be carried out independently.

- The eigenvalues of the closed-loop system are as designed by the state feedback law, unaffected by the use of a state estimator.

Feedback from Estimated States

The property of independence between control and state estimation is called the **Separation Principle**

Separation Principle: The design of the state feedback and the design of the state estimator can be carried out independently.

- The eigenvalues of the closed-loop system are as designed by the state feedback law, unaffected by the use of a state estimator.
- The eigenvalues of the observer are unaffected by the state feedback law.

Feedback from Estimated States

The closed-loop transfer function will only have the eigenvalues arising from $(\mathbf{A} - \mathbf{BK})$, since the estimation error is **uncontrollable**,

$$\mathbf{G_{cl}}(\mathbf{s}) = \mathbf{C}(\mathbf{sI} - \mathbf{A} + \mathbf{BK})^{-1}\mathbf{BN}.$$

Transients in state estimation, however, will be seen at the output, since the estimation error is **observable**.

Outline

- Feedback from Estimated States
- Discrete-Time Control Design
- Dead-Beat Control

Discrete-Time Control Design

For discrete-time state equations

$$\boxed{\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k], \qquad \mathbf{y}[k] = \mathbf{C}\mathbf{x}[k],}$$

the **design procedure** for a state feedback law $\mathbf{u}[k] = -\mathbf{K}\mathbf{x}[k]$ is **the same as for continuous-time systems**.

The same goes for a discrete-time state observer,

$$\boxed{\hat{\mathbf{x}}[k+1] = (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}\mathbf{y}[k].}$$

One difference is the **location of the desired eigenvalues**. E.g., for asymptotic stability, they should be **inside the unit circle**.

Discrete-Time Control Design

For discrete-time state equations

$$\boxed{x[k+1] = Ax[k] + Bu[k], \qquad y[k] = Cx[k],}$$

the **design procedure** for a state feedback law $u[k] = -Kx[k]$ is **the same as for continuous-time systems**.

The same goes for a discrete-time state observer,

$$\boxed{\hat{x}[k+1] = (A - LC)\hat{x}[k] + Bu[k] + Ly[k].}$$

One difference is the **location of the desired eigenvalues**. E.g., for asymptotic stability, they should be **inside the unit circle**.

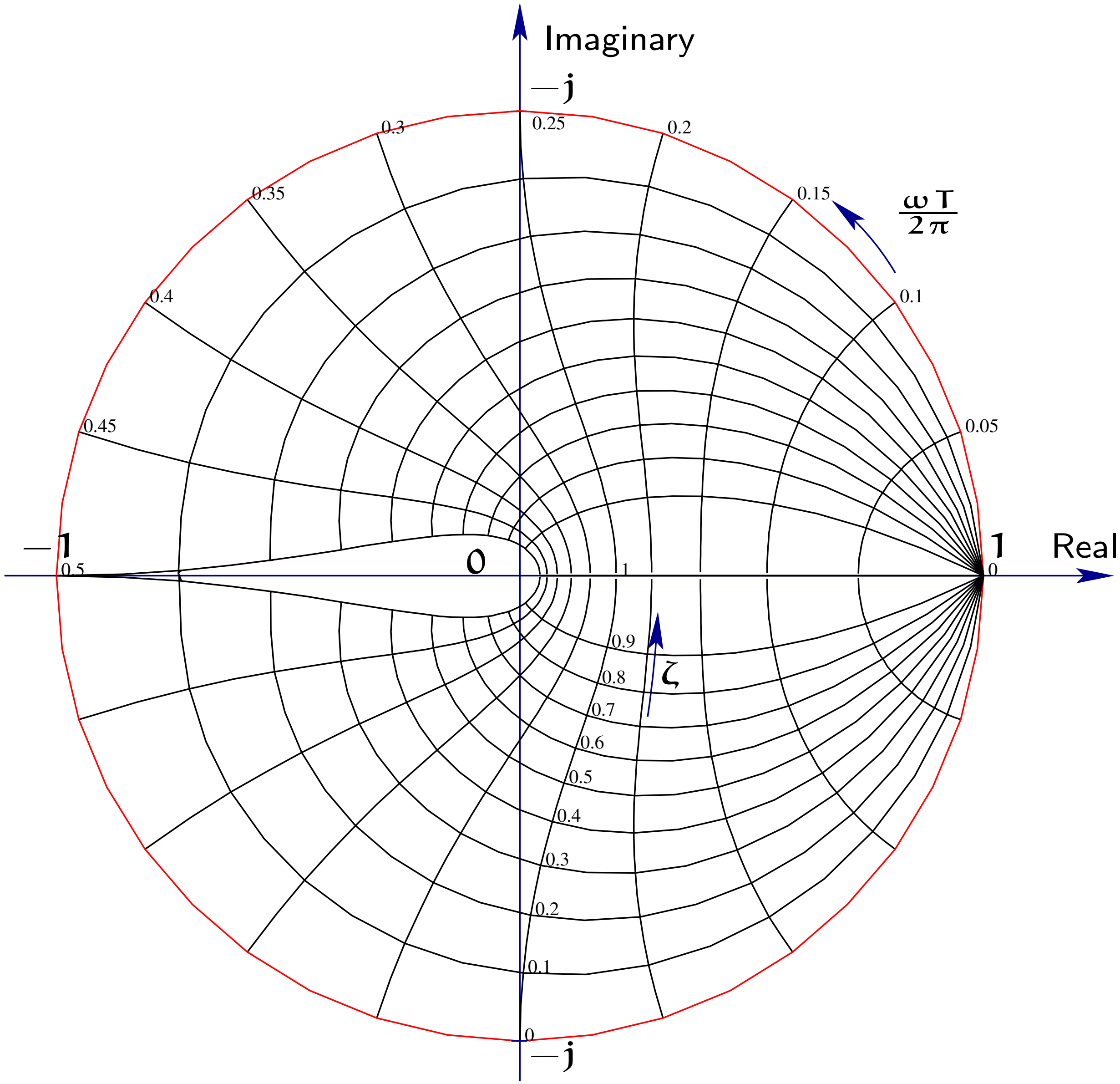
A practical rule to choose desired discrete-time eigenvalues is

1. choose a desired location in **continuous-time**, say p_i
2. translate it to discrete-time using the relation $\boxed{\lambda_i = e^{p_i T}}$.

Discrete-Time Control Design

Loci with constant damping ζ and constant frequencies ω in the discrete complex plane.

In MATLAB this grid may be obtained with the `zgrid` command.



Discrete-Time Control Design: Example

Example (Discrete-time speed control of a DC motor). We return to the DC motor we considered in the examples of the last lecture. We will suppose that the motor is to be controlled with a PC. Hence, the controller has to be **discrete-time**.

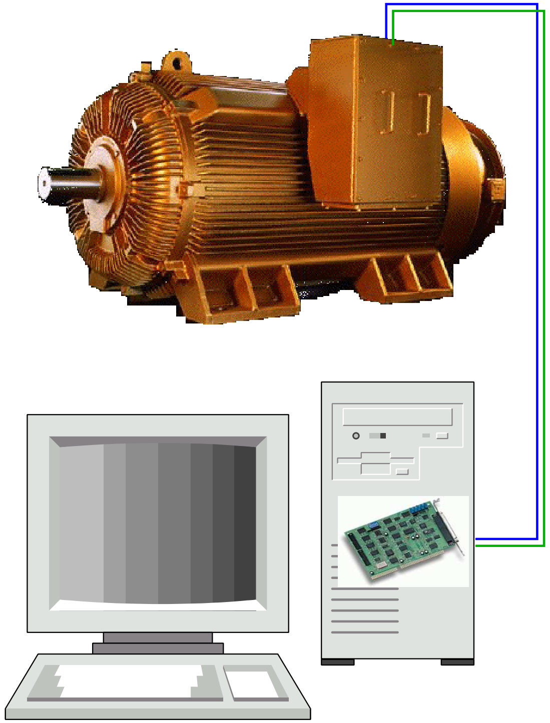
Discrete-Time Control Design: Example

Example (Discrete-time speed control of a DC motor). We return to the DC motor we considered in the examples of the last lecture. We will suppose that the motor is to be controlled with a PC. Hence, the controller has to be **discrete-time**.

This time, with a view to design a discrete-time control law, we first **discretise** the continuous-time model

$$\frac{d}{dt}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} = \begin{bmatrix}-10 & 1\\ -0.02 & -2\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix} + \begin{bmatrix}0\\ 2\end{bmatrix} V(t)$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\omega(t)\\ i(t)\end{bmatrix}$$



The first **design parameter** to define is the **sampling period** T.

Discrete-Time Control Design: Example

Example (Discrete design, step 1: choice of sampling period). From **Shannon's Sampling Theorem**, the sampling frequency $\omega_s = 2\pi/T$ should be **at least twice the bandwidth of the closed-loop system** (because we will **change** the system bandwidth with the control action).

Discrete-Time Control Design: Example

Example (Discrete design, step 1: choice of sampling period). From **Shannon's Sampling Theorem**, the sampling frequency $\omega_s = 2\pi/T$ should be **at least twice the bandwidth of the closed-loop system** (because we will **change** the system bandwidth with the control action).

The specification that we had for the previous continuous-time design was a settling time t_s of about 1s. The rule based on Shannon´s Theorem would then give a sampling time of less than $T = 0.5$s. In practice T is chosen at least 10 to 20 times faster than the desired closed-loop settling time. Here we choose

$$\boxed{T = 0.1s}$$

Discrete-Time Control Design: Example

Example (Discrete design, step 1: choice of sampling period). From **Shannon's Sampling Theorem**, the sampling frequency $\omega_s = 2\pi/T$ should be **at least twice the bandwidth of the closed-loop system** (because we will **change** the system bandwidth with the control action).

The specification that we had for the previous continuous-time design was a settling time t_s of about 1s. The rule based on Shannon´s Theorem would then give a sampling time of less than $T = 0.5$s. In practice T is chosen at least 10 to 20 times faster than the desired closed-loop settling time. Here we choose

$$\boxed{T = 0.1s}$$

Note that the maximum sampling speed is limited by the available computer clock frequency, and thus the time required for computations and signal processing operations.

Discrete-Time Control Design: Example

Example (Discrete design, step 2: discretisation of the plant). Having defined the sampling-time T, we compute the discrete-time state matrices

$$A_d = e^{At}, \quad \text{and} \quad B_d = \int_0^T e^{A\tau} B \, d\tau$$

From MATLAB, we do `[Ad,Bd]=c2d(A,B,0.1)` and obtain

$$A_d = \begin{bmatrix} 0.3678 & 0.0563 \\ -0.0011 & 0.8186 \end{bmatrix}, \quad B_d = \begin{bmatrix} 0.0068 \\ 0.1812 \end{bmatrix}$$

As we can verify, the open loop **discrete-time** eigenvalues are

$$0.3679 = e^{(-9.9975 \times 0.1)} \quad \text{and} \quad 0.8185 = e^{(-2.0025 \times 0.1)}$$

Discrete-Time Control Design: Example

Example (Discrete design, step 3: design of discrete feedback gain). The **discrete-time** characteristic polynomial is

$$\Delta(z) = (z - e^{-9.9975\mathrm{T}})(z - e^{-2.0025\mathrm{T}}) = z^2 - 1.1864z + 0.3011$$

Discrete-Time Control Design: Example

Example (Discrete design, step 3: design of discrete feedback gain). The **discrete-time** characteristic polynomial is

$$\Delta(z) = (z - e^{-9.9975T})(z - e^{-2.0025T}) = z^2 - 1.1864z + 0.3011$$

For the desired discrete characteristic polynomial, we first obtain the **discrete mapping** $p \to e^{pT}$ of the eigenvalues $p_{1,2} = -5 \pm j$ specified for the continuous-time system, i.e.,

$$\delta \pm j\gamma = e^{(-5.\;\;j).\;\;0.1} = 0.6035 \pm j0.0605$$

which yield the desired discrete characteristic polynomial

$$\Delta_K(z) = z^2 - 1.2070z + 0.3678.$$

Discrete-Time Control Design: Example

Example (Discrete design, step 3: design of discrete feedback gain). The **discrete-time** characteristic polynomial is

$$\Delta(z) = (z - e^{-9.9975T})(z - e^{-2.0025T}) = z^2 - 1.1864z + 0.3011$$

For the desired discrete characteristic polynomial, we first obtain the **discrete mapping** $p \to e^{pT}$ of the eigenvalues $p_{1,2} = -5 \pm j$ specified for the continuous-time system, i.e.,

$$\delta \pm j\gamma = e^{(-5.\ j).\ 0.1} = 0.6035 \pm j0.0605$$

which yield the desired discrete characteristic polynomial

$$\Delta_K(z) = z^2 - 1.2070z + 0.3678.$$

From the coefficients of $\Delta_K(z)$ and $\Delta(z)$ we get the discrete state feedback gain

$$\bar{K}_d = \begin{bmatrix}(-1.2070 + 1.1864) & (0.3678 - 0.3011)\end{bmatrix} = \begin{bmatrix}-0.0206 & 0.0667\end{bmatrix}$$

Discrete-Time Control Design: Example

Example (Discrete design, step 3, continuation). We then compute, just following the same procedure used in continuous-time, from the **discrete** matrices $\mathbf{A_d}$ and $\mathbf{B_d}$, the **discrete controllability matrices** $\mathcal{C}_\mathbf{d}$ and $\bar{\mathcal{C}}_\mathbf{d}$,

$$\mathcal{C}_\mathbf{d} = \begin{bmatrix} 0.0068555 & 0.0127368 \\ 0.1812645 & 0.1483875 \end{bmatrix}, \quad \text{and} \quad \bar{\mathcal{C}}_\mathbf{d} = \begin{bmatrix} 1 & -1.1864 \\ 0 & 1 \end{bmatrix}^{-1}$$

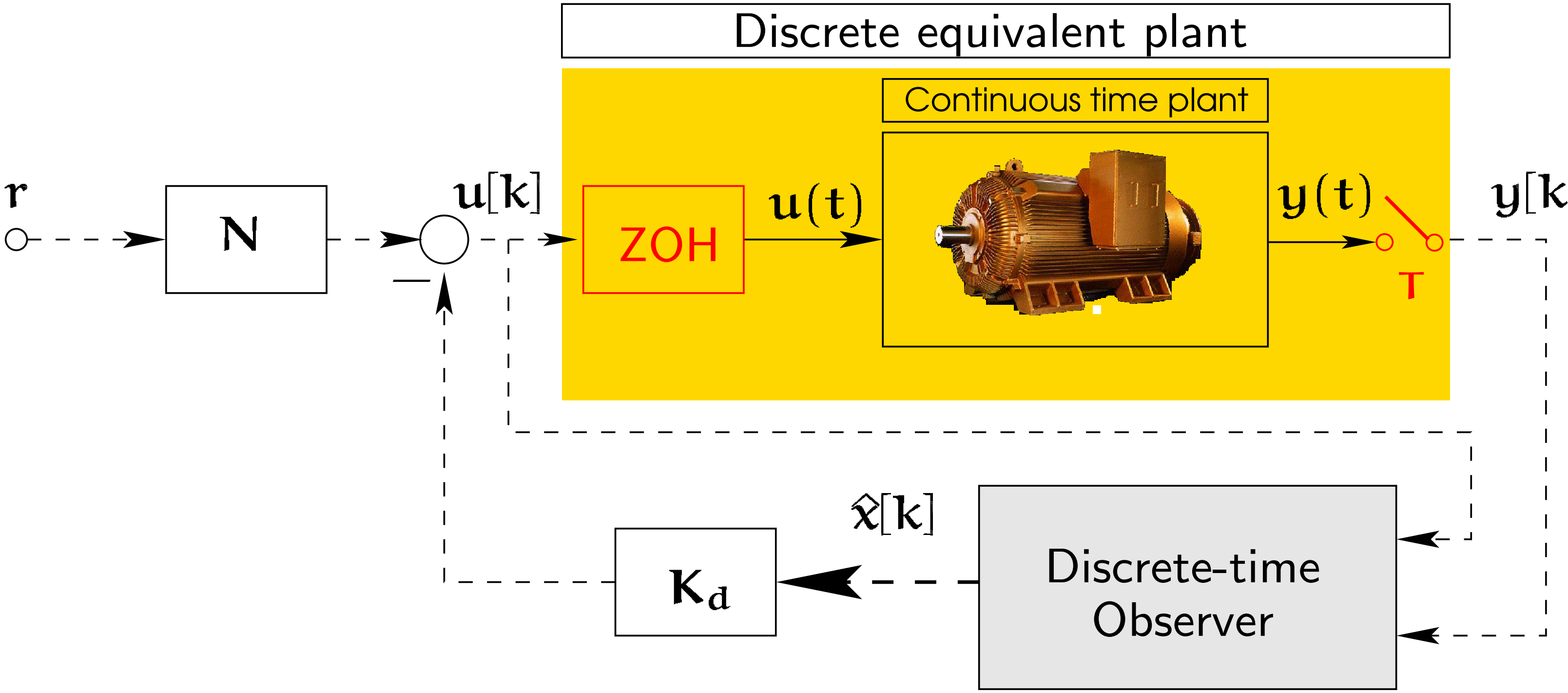
We thus obtain, in the original coordinates, the **discrete-time** feedback gain

$$K_\mathbf{d} = \bar{K}_\mathbf{d} \bar{\mathcal{C}}_\mathbf{d} \mathcal{C}_\mathbf{d}^{-1} = \begin{bmatrix} 8.3011164 & -0.4270676 \end{bmatrix}$$

As can be verified with MATLAB, $(\mathbf{A_d} - \mathbf{B_d} K_\mathbf{d})$ will have the desired discrete eigenvalues.

Discrete-Time Control Design: Example

Example (Discrete design: continuation). In a similar fashion, we can carry out the design of the discrete-time observer, based on the discrete-time model of the plant. The output feedback design is finally implemented on the **continuous-time** plant through a Zero Order Hold and a Sampler.



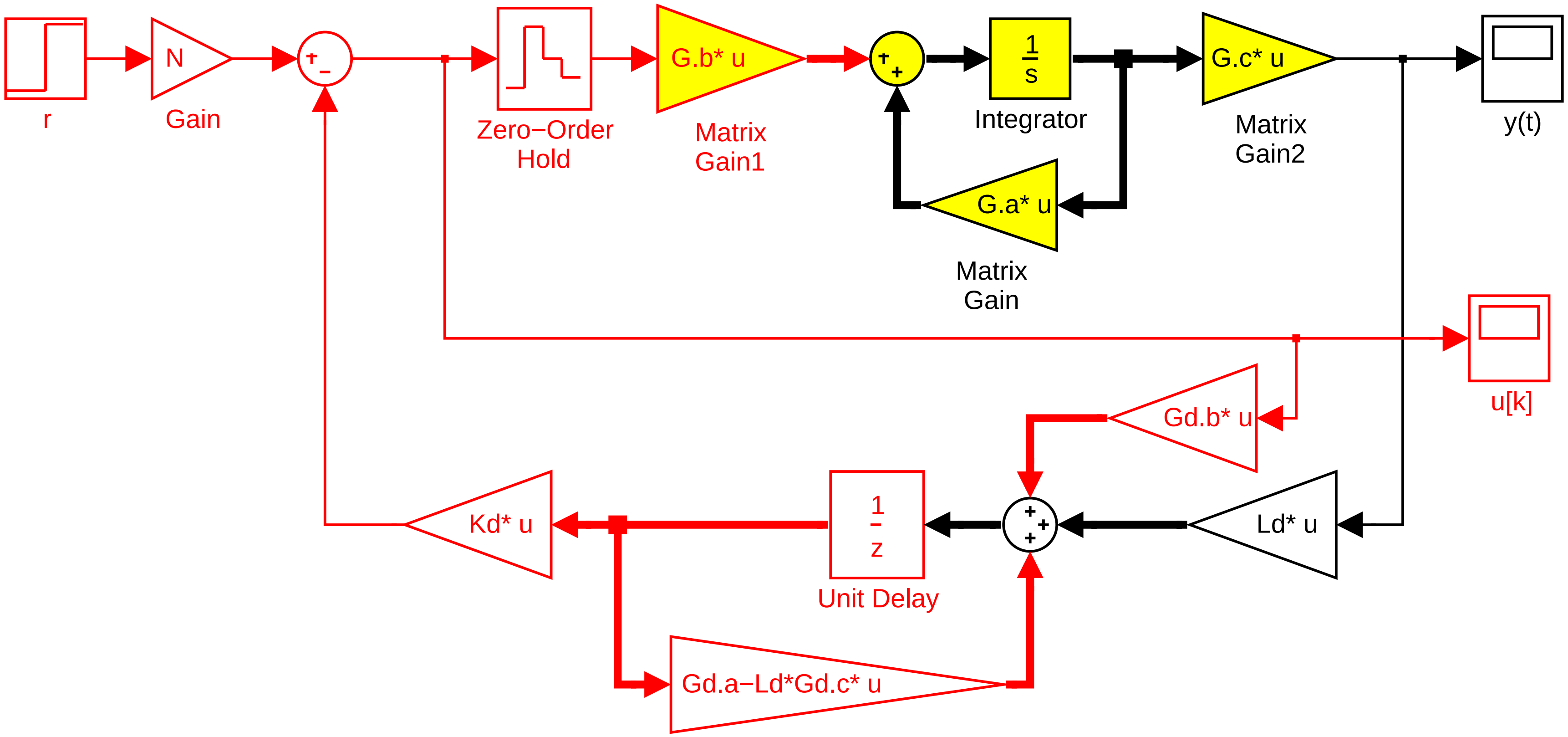
Discrete-Time Control Design: Example

Example (Discrete design: MATLAB script). We used the following MATLAB script to compute the gains and run the simulations.

```
% Continuous-time matrices
A=[-10 1;-0.02 -2];B=[0;2];C=[1 0];D=0;
G=ss(a,B,C,D);    % state space system definition
T=0.1;            % Sampling time
Gd=c2d(G,T,’zoh’) % discretisation
% Discrete-time feedback gain
Kd=place(Gd.a,Gd.b,exp([-5-i,-5+i]*T))
% Discrete-time observer gain
Ld=place(Gd.a’,Gd.c’,exp([-6-i,-6+i]*T))’
% steady state error compensation
N=inv(Gd.c*inv(eye(2)-Gd.a+Gd.b*Kd)*Gd.b)
% Run simulink diagram
sim(’dmotorOFBK’)
% Plots (after simulations have been run)
subplot(211),plot(y(:,1),y(:,2));grid
subplot(212),stairs(u(:,1),u(:,2));grid
```

Discrete-Time Control Design

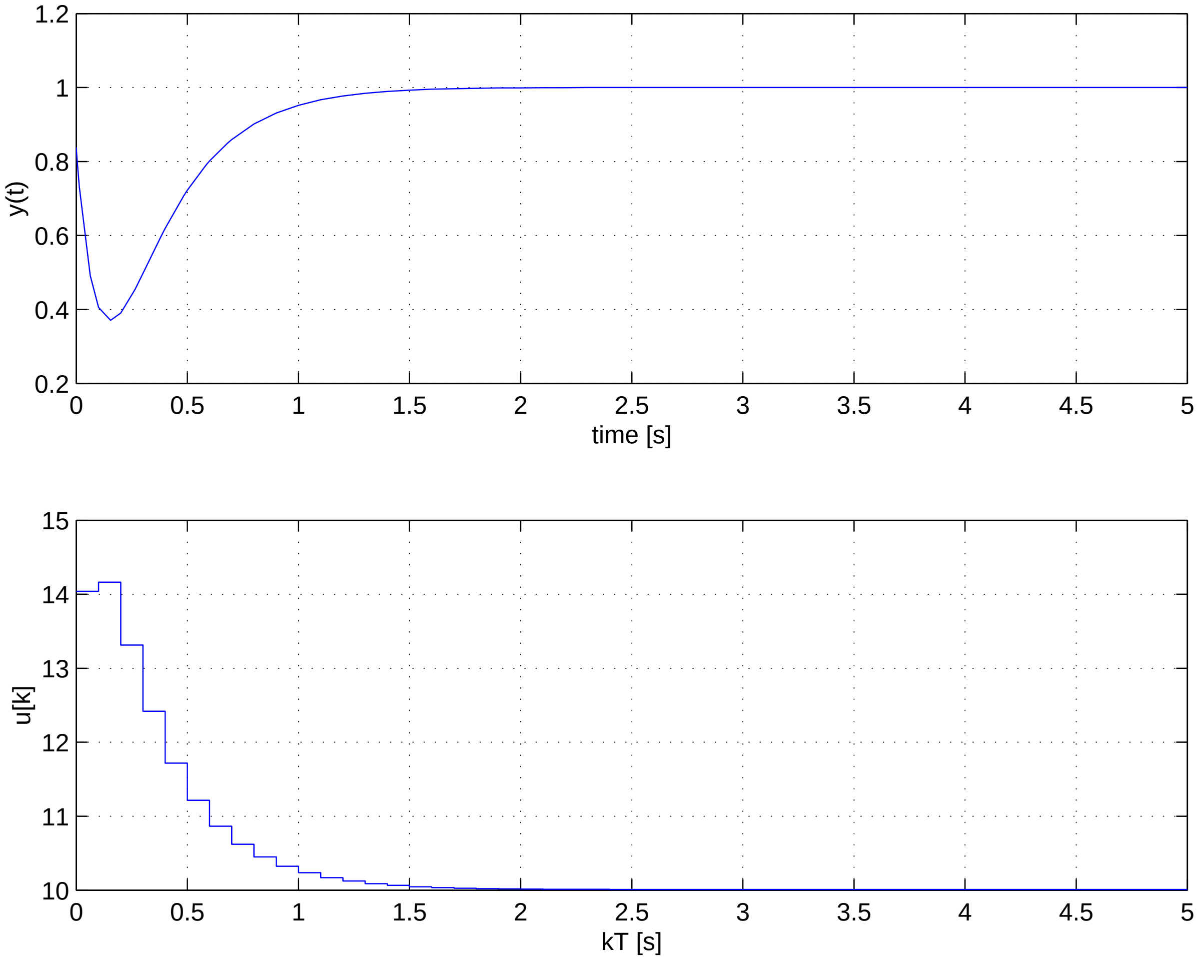
Example (Discrete design: SIMULINK diagram). We used the following SIMULINK diagram to run the simulations.



Notice the **sampled** signals coloured in red (check the option "Sample Time Colors" in the *Format* menu). All blocks with discrete-time signals include a sampler at their input.

Discrete-Time Control Design: Example

Example (Discrete design: simulations). The following plots show the response of the closed-loop sampled-data controlled system: continuous-time output $\mathbf{y}(\mathbf{t})$ and discrete-time control signal $\mathbf{u}[\mathbf{k}]$.



Discrete-Time Design: Peculiarities

Two $\boxed{\textbf{special differences}}$ in the discrete-time design procedure:

- The gain $\mathbf{N}$ for steady state error compensation is, as in continuous-time, the inverse of the steady-state DC gain of the **closed-loop** transfer function. **Notice** though, that in discrete-time

$$\mathbf{N} = \frac{1}{\mathbf{C}(z\mathbf{I} - \mathbf{A_d} + \mathbf{B_d K_d})^{-1}\mathbf{B_d}\Big|_{z=1}} = \frac{1}{\mathbf{C}(\mathbf{I} - \mathbf{A_d} + \mathbf{B_d K_d})^{-1}\mathbf{B_d}}$$

Discrete-Time Design: Peculiarities

Two **special differences** in the discrete-time design procedure:

- The gain $\mathbf{N}$ for steady state error compensation is, as in continuous-time, the inverse of the steady-state DC gain of the **closed-loop** transfer function. **Notice** though, that in discrete-time

$$\mathbf{N} = \frac{1}{\mathbf{C}(z\mathbf{I} - \mathbf{A_d} + \mathbf{B_d K_d})^{-1}\mathbf{B_d}\big|_{z=1}} = \frac{1}{\mathbf{C}(\mathbf{I} - \mathbf{A_d} + \mathbf{B_d K_d})^{-1}\mathbf{B_d}}$$

- If we implement robust tracking by adding **Integral Action** into the state feedback design, notice that the plant augmentation is different in discrete-time,

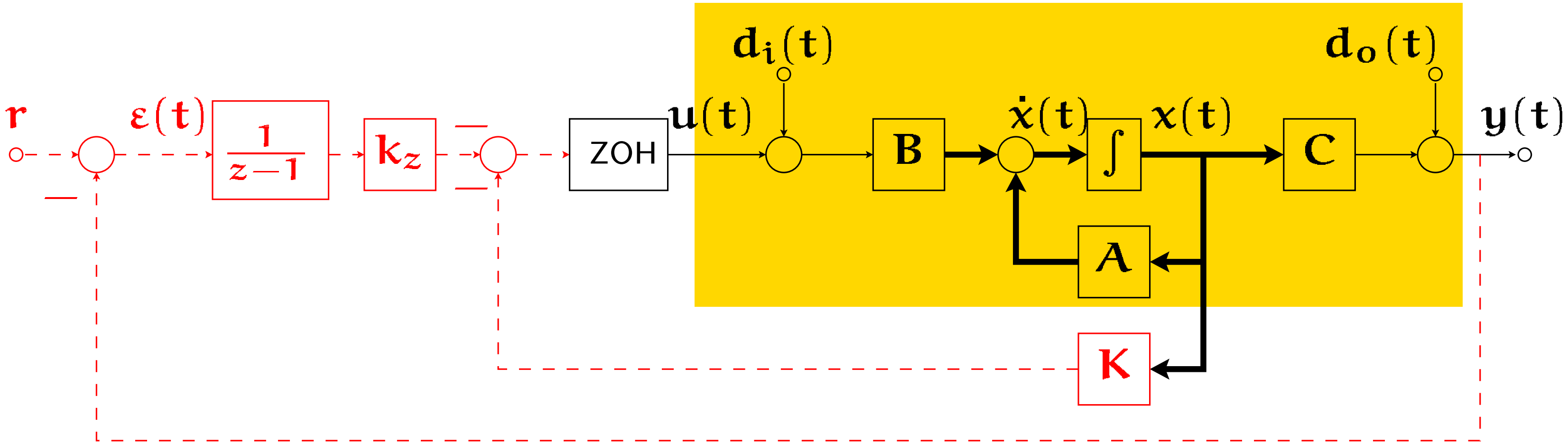
$$\mathbf{A_a} = \begin{bmatrix} \mathbf{A_d} & 0 \\ -\mathbf{C} & 1 \end{bmatrix}$$

since the **discrete-integration** of the tracking error has to be implemented as

$$z[k+1] = z[k] + r - \mathbf{C}x[k]$$

Discrete-Time Design: Peculiarities

The implementation of the **discrete-time integral action** in the diagram should be consistent, i.e. **discrete-integration** of the tracking error.



Apart from these two differences, and the locations for the eigenvalues/poles, the discrete-time design is obtained by performing the same computations for continuous-time state feedback and observer design.

Outline

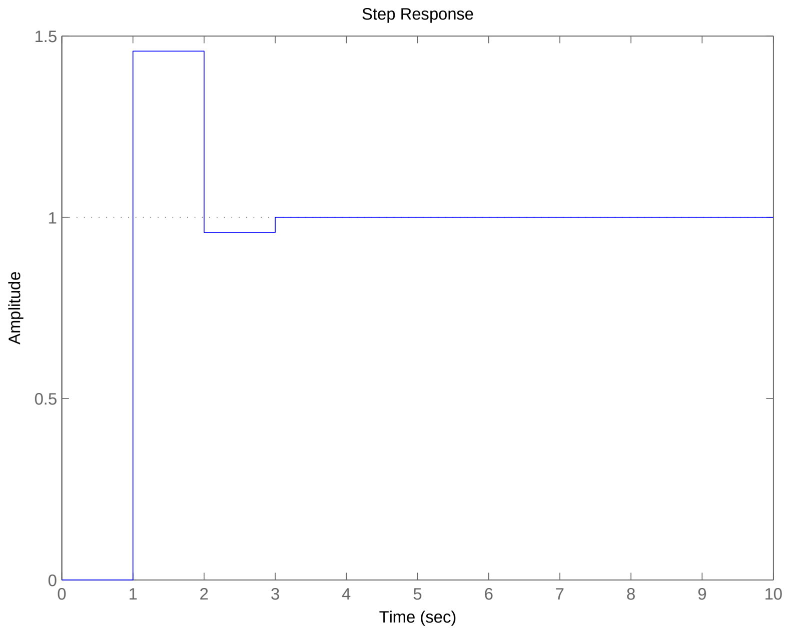
- Feedback from Estimated States
- Discrete-Time Control Design
- Dead-Beat Control

Discrete-Time Design: Dead-Beat Control

A special discrete-time design that **has no correlate** in continuous-time is **dead-beat control**.

A **dead-beat** response is a response that settles in its final value at a **finite time**. This feature arises in a discrete-time system which has all its poles at $z = 0$, e.g.,

$$G(z) = \frac{(7z - 1)(5z - 1)}{24z^3},$$



which will settle in 3 sampling periods (the dynamics is just a **delay** of 3 sample periods).

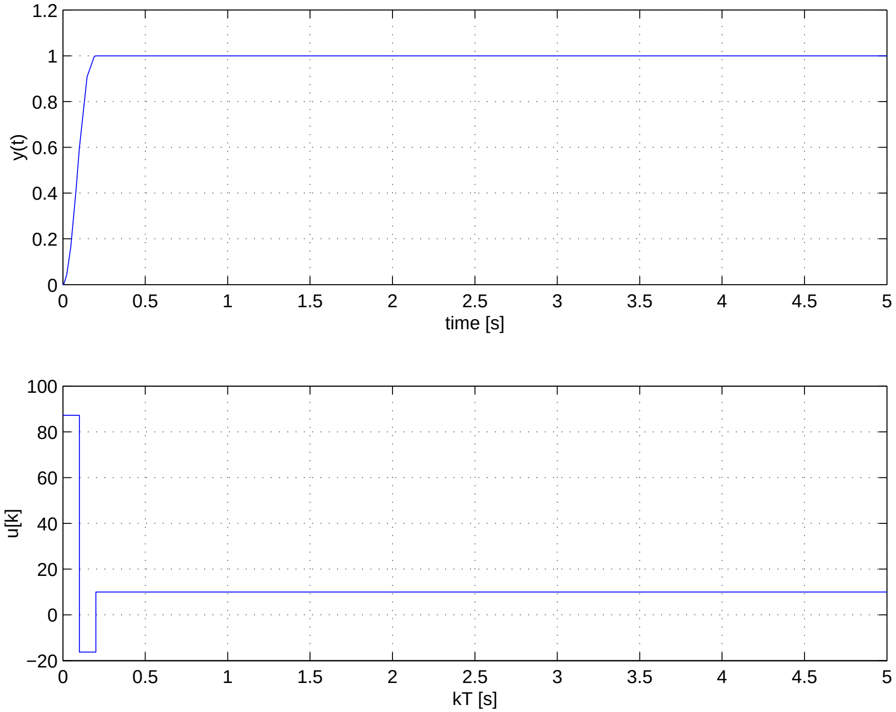
Discrete-Time Design: Dead-Beat Control

To design a **dead-beat** controller, we just have to find $\mathbf{K_d}$ to place all the closed-loop poles at $z = 0$. The discrete-time observer can also be designed **dead-beat**, with $\mathbf{L_d}$ to place all the observer poles at $z = 0$.

The plot shows the response of the DC motor of the example controlled to have **dead-beat** response (state feedback — no observer).

There is not much flexibility in dead-beat control, the only parameter to change the response is the sampling period.

Dead-beat usually requires **large control action**, which may saturate actuators.



Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently

Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently
- We have discussed the design of **discrete-time** state feedback controllers and observers. The procedures are the same as in continuous-time, save for
  - the choice of desired eigenvalues (which have to be in the unit circle for stability)
  - the compensation of steady state tracking error via the feedforward gain $\mathbf{N}$ (be careful with the formula used)
  - the implementation of integral action to achieve robust tracking and disturbance rejection (be careful, use **discrete integration**)

Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently
- We have discussed the design of **discrete-time** state feedback controllers and observers. The procedures are the same as in continuous-time, save for
  - the choice of desired eigenvalues (which have to be in the unit circle for stability)
  - the compensation of steady state tracking error via the feedforward gain $\mathbf{N}$ (be careful with the formula used)
  - the implementation of integral action to achieve robust tracking and disturbance rejection (be careful, use **discrete integration**)
- We presented a special discrete design: **dead-beat control**.

Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently
- We have discussed the design of **discrete-time** state feedback controllers and observers. The procedures are the same as in continuous-time, save for
  - the choice of desired eigenvalues (which have to be in the unit circle for stability)
  - the compensation of steady state tracking error via the feedforward gain $\mathbf{N}$ (be careful with the formula used)
  - the implementation of integral action to achieve robust tracking and disturbance rejection (be careful, use **discrete integration**)
- We presented a special discrete design: **dead-beat control**.

Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently
- We have discussed the design of **discrete-time** state feedback controllers and observers. The procedures are the same as in continuous-time, save for
  - the choice of desired eigenvalues (which have to be in the unit circle for stability)
  - the compensation of steady state tracking error via the feedforward gain $\mathbf{N}$ (be careful with the formula used)
  - the implementation of integral action to achieve robust tracking and disturbance rejection (be careful, use **discrete integration**)
- We presented a special discrete design: **dead-beat control**.

Conclusions

- We have reviewed the **state feedback** and **state estimator** design procedure, and showed that the **Separation Principle** holds: the two designs can be carried out independently
- We have discussed the design of **discrete-time** state feedback controllers and observers. The procedures are the same as in continuous-time, save for
  - the choice of desired eigenvalues (which have to be in the unit circle for stability)
  - the compensation of steady state tracking error via the feedforward gain $\mathbf{N}$ (be careful with the formula used)
  - the implementation of integral action to achieve robust tracking and disturbance rejection (be careful, use **discrete integration**)
- We presented a special discrete design: **dead-beat control**.