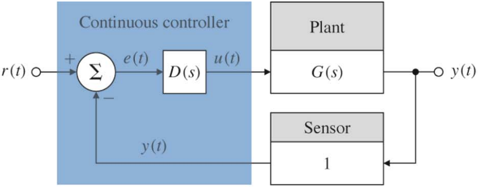
Control Systems
Lecture 8

Discrete-time Control

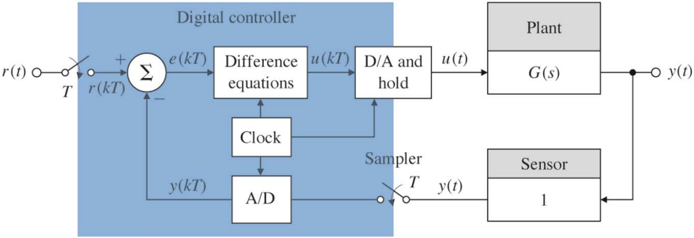
Digital (discrete-time) control

- Continuous-time controllers are implemented in discrete-time form (sampled-time rather than continuous-time)
- In digital control also the signal values are discretized; (we neglect this here and assume sufficiently accurate A/D converters)
- We focus on three questions:
  - How does the sample interval affect the control performance? (How to choose the sample-interval?)
  - How to implement a discrete-time controller on the basis of a designed continuous-time controller?
  - How to design a controller in discrete-time, on the basis of a discrete-time model

Continuous and digital controller



(a)



(b)

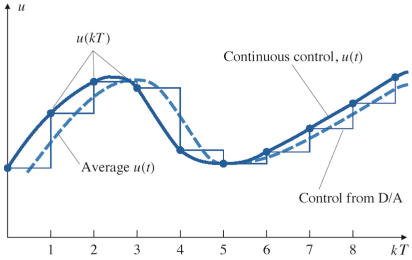
Basic ingredients of the digital controller:

From continuous to digital:

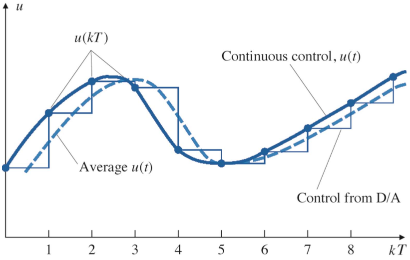
- Sampler: $y(t) \to y(kT)$ with $k \in \{0, 1, 2, \cdots\}$ and T the sampling interval

From digital to continuous:

- (Zero-order) hold: $u(t) = u(kT)$ for $kT \leq t < (k+1)T$



A first global effect:



A smoothened (averaged) version of the zero-order hold signal has a delay of $T/2$ compared to the continuous-time signal. This adds to the system as an additional time-delay.

Any computation time that the (digital) processor needs comes on top of this.

- At the bandwidth frequency ω_{BW} a delay of $T/2$ [sec] corresponds to a phase lag of

$$-(T/2)\ /\ \underbrace{[2\pi/\omega_{BW}]}_{T_{BW}} \times 2\pi = -\omega_{BW}\frac{T}{2} = -\omega_{BW}\frac{\pi}{\omega_s}\ [rad]$$

- For $\omega_s = 25\omega_{BW}$ the phase lag induced by the zero-order hold at ω_{WB} is

$$-\omega_{BW}\frac{\pi}{\omega_s} = -\frac{\pi}{25} = -7.2^o$$

  (see also time-delay slide in Lecture 7)
  This is often used as a rule-of-thumb lower bound for the sampling frequency
- With $\omega_c \leq \omega_{BW} \leq 2\omega_c$, the phase lag will be smaller than 7.2^o at ω_c.

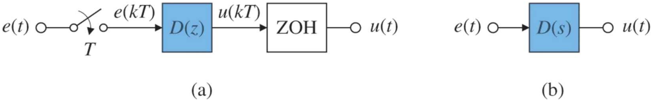
Choice of sampling interval

When implementing a designed continuous-time controller in discrete-time, the sampling frequency $\frac{2\pi}{T}$ should satisfy

$$\omega_s \geq 25\omega_{BW}$$

- In the considered situation the continuous-time controller can be "discretized". If a slower sampling frequency is chosen a -dedicated- discrete-time control design should be made.

How to find a discrete equivalent of a continuous controller?



The discrete-time equivalent of a differential equation:

Difference equation:

$$y(k)+a_1y(k-1)+\cdots+a_ny(k-n) = b_0u(k)+b_1u(k-1)+\cdots+b_nu(k-n)$$

By using the z-transform:

$$y(k) \to Y(z) := \sum_{k=0}^{\infty} y(k)z^{-k}, \qquad z \in \mathbb{C}$$

with the property $y(k-1) \to z^{-1}Y(z)$
we assign to this difference equation a (discrete-time) transfer function

$$\frac{Y(z)}{U(z)} = \frac{b_0 + b_1z^{-1} + \cdots b_nz^{-n}}{1 + a_1z^{-1} + \cdots a_nz^{-n}}.$$

Intermezzo: z-transform

Properties z-transform: $F(z) = \mathcal{Z}\{f(k)\}$[1]

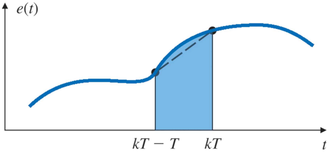
Discrete-time function	z-Transform
$f(k), k \geq 0$	$F(z)$
$f(k-1)$	$z^{-1}F(z)$
$f(k+1)$	$zF(z)$
unit pulse $f(0) = 1$	1
delayed pulse $f(k_0) = 1$	z^{-k_0}
unit step $f(k) = 1,\ k \geq 0$	$\frac{z}{z-1}$
a^k	$\frac{z}{z-a}$

[1] Note that there is a slight notational difference with $\mathcal{Z}\{f(kT)\}$

Tustin's Method (Trapezoidal integration)

Representing an integrator action in the discrete-time:
Continuous-time integrator $\frac{U(s)}{E(s)} = \frac{1}{s}$,

$$\begin{aligned} u(kT) &= \int_0^{kT} e(t)dt = \int_0^{kT-T} e(t)dt + \int_{kT-T}^{kT} e(t)dt \\ &= u(kT-T) + \text{area under } e(t) \text{ over last } T \end{aligned}$$



$$\approx \quad u(kT-T) + \frac{T}{2}[e(kT-T) + e(kT)]$$

Converting to discrete-time setting:

$$\begin{aligned} u(kT) &= u(kT-T) + \frac{T}{2}[e(kT-T)+e(kT)] \\ u(k) &= u(k-1) + \frac{T}{2}[e(k-1)+e(k)] \end{aligned}$$

This reflects a discrete-time system with transfer function

$$\frac{U(z)}{E(z)} = \frac{T}{2}\left(\frac{1+z^{-1}}{1-z^{-1}}\right) = \frac{1}{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}$$

The continuous-time to discrete-time mapping then is

$$\frac{1}{s} \to \frac{1}{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} \quad \text{or} \quad s \to \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)$$

Controller discretization

A designed continuous-time controller

$$D(s) = \frac{s\tau_1 + 1}{s\tau_a + 1}$$

can -by Tustin's approximation- be implemented in discrete-time by the variable subsitution

$$s \rightarrow \frac{2}{T}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)$$

i.e.

$$D(z) = \frac{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)\tau_1 + 1}{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)\tau_a + 1}$$

In Matlab:

```
Dd = c2d(D,T,'t')
```

where the 't' denotes the trapezoidal (Tustin) conversion
`D` is the continuous-time system, and
`Dd` is the discrete-time system.

Example 4.12 (Ed.6) - Speed Control

For a system with plant dynamics

$$G(s) = \frac{45}{(s+9)(s+5)}$$

a PI controller is designed with transfer function

$$D(s) = 1.4\frac{s+6}{s}.$$

The bandwidth of this control system is $\omega_{BW} \approx 10$ rad/sec.

Note that $\omega_s = 25\omega_{BW}$ leads to $T_s \approx 0.025$ sec.

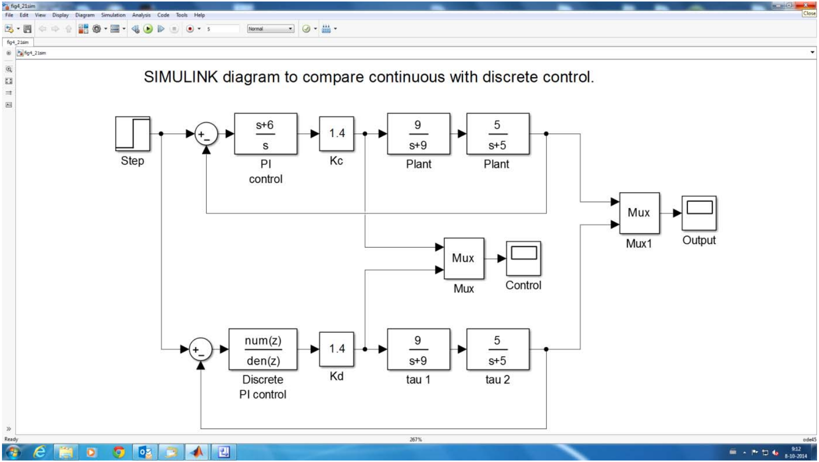
We are discretizing the control for two sample intervals:

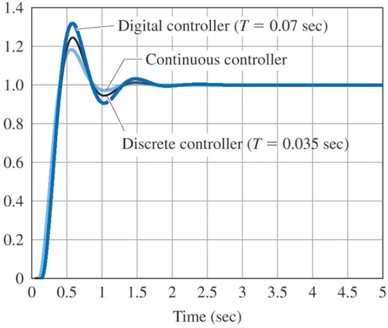
$$T_s = 0.07 \;\rightarrow\quad D_d(z) = 1.4\frac{1.21z - 0.79}{z-1}$$

and

$$T_s = 0.035 \;\rightarrow\quad D_d(z) = 1.4\frac{1.105z - 0.895}{z-1}$$

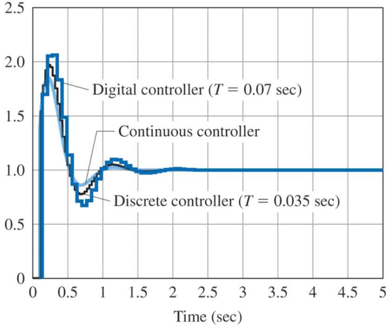
```
numDa = [1 6]; denDa = [1 0];
```
(gain of 1.4 is separate block in simulink)
```
sysDa = tf(numDa,denDa); sysDd = c2d(sysDa,T,'t');
[numD,denD,T] = tfdata(sysDd,'v');
```


SIMULINK diagram to compare continuous with discrete control.
Step
s+6 / s
PI control
1.4
Kc
9 / s+9
Plant
5 / s+5
Plant
Mux
Mux1
Output
Mux
Mux
Control
num(z) / den(z)
Discrete PI control
1.4
Kd
9 / s+9
tau 1
5 / s+5
tau 2



(a)

Output response to step input



(b)

Control output

Evaluate what happens for increasing sample intervals

Simulink file `fig4_21sim.mdl` is available (zipped) in the exercise folder of Canvas.

Alternative relations between s and z: matched pole-zero method

Alternative reasoning (to Tustin) for relating s to z:
The continuous-time signal

$$f(t) = e^{-at}, \quad a \in \mathbb{R}, t \geq 0$$

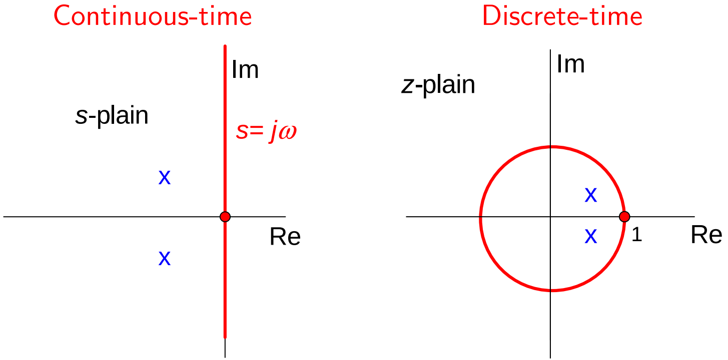
has the Laplace transform: $F(s) = \frac{1}{s+a}$, i.e. a pole at $s = -a$.
The z-transform of $f(kT)$ is

$$F(z) = \sum_{k=0}^{\infty} e^{-akT} z^{-k} = \frac{1}{1 - e^{-aT} z^{-1}} = \frac{z}{z - e^{-aT}}$$

i.e. a pole at $z = e^{-aT}$.

This suggests the $s \leftrightarrow z$ relation: $z = e^{sT}$.

$$z = e^{sT}$$



- Imaginary axis: $s = j\omega \rightarrow z = e^{j\omega T}$ (unit circle)
- $s = 0 \rightarrow z = 1$
- Region of stability: poles in Re $s < 0 \rightarrow |z| < 1$
- Pole location $s = p_i \rightarrow z = e^{p_i T}$

Approximation methods for s-z relations:

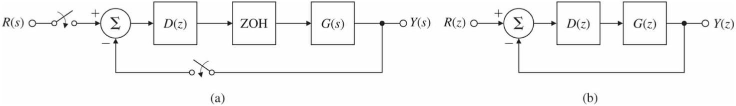
- Tustin
- Matched pole-zero

and alternatively also

- Zero-order hold (ZOH)
  Only applicable to the situation that the (continuous-time) controller input is assumed to be of zero-order hold.

Discrete design

Analyzing the control system in discrete time



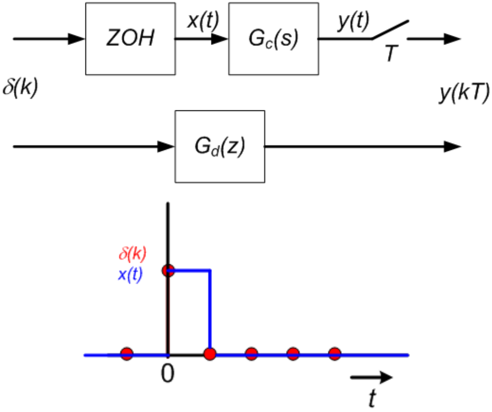
If a continuous-time system $G(s)$ is preceded by a ZOH (zero-order hold), then its output $y(kT)$ at the sample intervals can be exactly described by the discrete equivalent system:

$$G(z) = (1 - z^{-1})\mathcal{Z}\left\{\frac{G(s)}{s}\right\}$$

where $\mathcal{Z}(F(s))$ refers to Z-transform of the time sequence $f(kT)$.

Explanation

Find $G_d(z)$ such that $y(kT)$ of both systems are the same:



with $\delta(k)$ a discrete pulse signal.

Explanation (cont'd)

With $u(t)$ a unit-step function, it follows that

$$x(t) = u(t) - u(t - T).$$

Then

$$y(t) = m(t) - m(t - T)$$

with $m(t)$ the step response of G_c.

Then

$$G_d(z) = \mathcal{Z}\{y(kT)\} = \mathcal{Z}\{m(kT)\} - z^{-1}\mathcal{Z}\{m(kT)\}$$

Since $m(t) = \mathcal{L}^{-1}\{\frac{G_c(s)}{s}\}$ it follows that

$$G_d(z) = (1 - z^{-1})\mathcal{Z}\left(\frac{G_c(s)}{s}\right)$$

On the basis of $G(z)$ we can now design a discrete controller $D(z)$ directly in the discrete domain

- Most of the CT design tools remain intact (closed-loop transfers, root loci, ...)
- Characteristic equation: $1 + D(z)G(z) = 0$ remains to have the same interpretation (closed-loop poles)
- Only the region of stability changes (inside unit circle for z)

Example 8.3

Consider $G(s) = \dfrac{a}{s+a}$, with $a > 0$, and proportional controller

$$D(z) = K$$

Then $G(z) = (1 - z^{-1})\mathcal{Z}\left[\dfrac{a}{s(s+a)}\right]$

By using Laplace/Z-transform table 8.1, and realizing that

$$\frac{a}{s(s+a)} = \frac{1}{s} - \frac{1}{s+a}$$

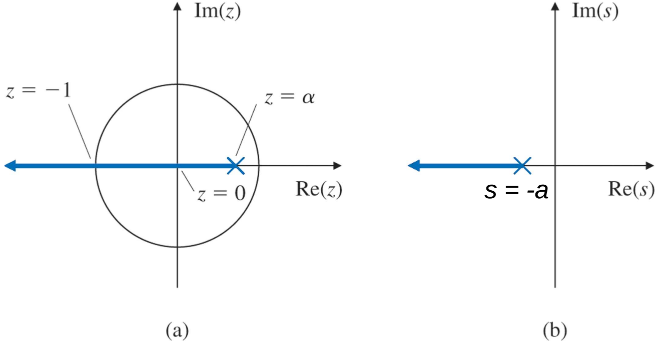
it follows that

$$\begin{aligned} G(z) &= (1 - z^{-1})\left[\frac{(1 - e^{-aT})z^{-1}}{(1 - z^{-1})(1 - e^{-aT}z^{-1})}\right] \\ &= \frac{1-\alpha}{z-\alpha} \quad \text{with } \alpha = e^{-aT}. \end{aligned}$$

TABLE 8.1 - Laplace Transforms and z-Transforms of Simple Discrete Time Functions

No.	$F(s)$	$f(kT)$	$F(z)$
3	$\frac{1}{s}$	$1(kT)$	$\frac{z}{z-1}$
4	$\frac{1}{s^2}$	kT	$\frac{Tz}{(z-1)^2}$
8	$\frac{1}{s+a}$	e^{-akT}	$\frac{z}{z-e^{-aT}}$
12	$\frac{a}{s(s+a)}$	$1-e^{-akT}$	$\frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$

Root locus

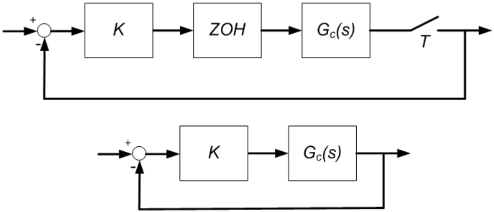


In continuous-time the closed-loop systems stays stable for increasing K.

In discrete-time the locus also goes to the left, but this leads to instability for sufficiently large K

The discrete design directly reflects the effect of the ZOH (phase lag) that introduces instability for sufficiently high controller gains

Compare discrete-time situation (top) and continuous-time situation (bottom):



Discrete design principles include:

- Placement of closed-loop poles:
  - sufficient distance from unit circle for good damping
  - typically close to/on positive real axis
  - not too close to $z = 1$ for high bandwidth
- Including integrator action

$$k_I \frac{z}{z-1}$$

for steady state reference tracking

Summary

- Discretization of controller introduces phase lag in the continuous-time closed-loop
- In order to avoid problems: choose $\omega_s \geq 25\omega_{BW}$
- Controller discretization can be done through Tustin's approximation
- Analysis of discrete-time control system is done in the z-domain
- In general case (also for small sampling intervals) controller can be designed in the discrete (z)-domain.