Exercises for lectures
20 – Digital Control

Michael Šebek
Automatic control 2016
Sampling: $s$ and $z$ relationship for complex poles

Continuous signal

- $y(t) = e^{-\alpha t} \sin \beta t, t > 0$
- Laplace transform
  
  \[ y(s) = \frac{\beta}{(s + \alpha)^2 + \beta^2} \]
- With poles
  \[ s_{1,2} = -\alpha \pm j\beta \]

Discrete signal

- $z$-Transform
  \[ y(k) = e^{-\alpha kh} \sin(\beta kh) \]
  \[ y(z) = \frac{z^{-1} e^{-\alpha h} \sin(\beta h)}{1 - z^{-1} 2e^{-\alpha h} \cos(\beta h) + z^{-2} e^{-2\alpha h}} = \frac{z^{-1} e^{-\alpha h} \sin(\beta h)}{1 - z^{-2} 2e^{-\alpha h} \cos(\beta h) + e^{-2\alpha h}} \]
- With poles
  \[ z_{1,2} = e^{-\alpha h} \left( \cos(\beta h) \pm j \sin(\beta h) \right) = e^{(-\alpha \pm j\beta)h} \]
- There is a relationship between the continuous and sampled system poles.

\[ z_{1,2} = e^{s_{1,2}h} \]
• Signals with frequencies from 0 to $\pi/h$ are displayed to unit circle after sampling. Where are displayed signals with higher frequencies?

• Consider sine signal with L-transform and poles

\[
y(t) = \sin \omega_1 t \quad y(s) = \frac{\omega_1}{s^2 + \omega_1^2} \quad s_{1,2} = \pm j\omega_1
\]

• The sampling period $h$

\[
y(k) = \sin(\omega_1 hk) \quad y(z) = \frac{z \sin \omega_1 h}{z^2 - z2\cos \omega_1 h + 1} \quad z_{1,2} = e^{\pm j\omega_1 h}
\]

• In case

\[
\omega_1 > \pi/h \leftrightarrow h > \pi/\omega_1
\]

in Hz

\[
f_1 = \frac{\omega_1}{2\pi} \text{ Hz, } f_s = \frac{1}{h} < 2f_1 \quad \omega_1 h > \pi
\]

• for $\omega_1 h > \pi$ is $e^{-j\omega_1 h} = e^{j(2\pi - \omega_1 h)}$, $e^{j\omega_1 h} = e^{-j(2\pi - \omega_1 h)}$, where $(2\pi - \omega_1 h) \in [0,180^\circ]$

• And position of the poles corresponds to the frequency

\[
(\omega_2 h = 2\pi - \omega_1 h) \rightarrow \omega_2 = 2\pi/h - \omega_1 = \omega_s - \omega_1
\]
We don't know the correct frequency band in a reverse transform (signal reconstruction).

To prevent this, we have to sample with a higher sample rate. Or filter out a higher frequencies than $\omega_N = \omega_s / 2$ (anti-aliasing filter).
Disk drive Arm

- simplified (normalized to 1)
- more detail ÅW, s13, ex1.2
- Transfer function from voltage to arm position

\[ G(s) = \frac{1}{s^2} \]

- goal: follow the track
- Accurate position control
- Important dynamics – reading speed
- Control structure
Example: continuous control

- Continuous controller (designed by “continuous methods“)

\[ u(s) = \frac{1}{2} u_C(s) - 2 \frac{s + 0.5}{s + 2} y(s) \]

- CL characteristic polynomial

\[ c_{CL}(s) = (s + 1)(s^2 + s + 1) \]

\[ s_1 = 1 \]

\[ s_{2,3} = -\frac{1}{2} \pm j \frac{\sqrt{3}}{2} \]

- CL transfer function

\[ y(s) = \frac{1}{2} \frac{s + 2}{(s + 1)(s^2 + s + 1)} u_C(s) \]

- simulation `AW_1_2.mdl`

- Setting time for 5% is 5.5, overshoot to 10% - OK

- How to realize digitally?

\[ (s + 1.0000)(s^2 + 1.0000s + 1.0000) \]
Example: Naive controller approximation

• Continuous controller is

\[ u(s) = 0.5u_c(s) - 2 \frac{s + 0.5}{s + 2} y(s) = 0.5u_c(s) - 2y(s) + 2 \frac{1.5}{s + 2} y(s) = 2[0.25u_c(s) - y(s) + x(s)] \]

where \( x(s) = \frac{1.5}{s + 2} y(s) \)

• We get a continuous time domain algorithm (control law)

\[ u(t) = 2[0.25u_c(t) - y(t) + x(t)] \]

\[ \frac{dx}{dt} = -2x(t) + 1.5y(t) \]

• Discrete algorithm – we sample signal with a period \( h \)

• And the derivative is approximated by difference

\[ \frac{x(t + h) - x(t)}{h} = -2x(t) + 1.5y(t) \]
So we get a discrete approximation

\[
x(t_k + h) = x(t_k) + h\left[1.5y(t_k) - 2x(t_k)\right]
\]

\[
u(t_k) = 2\left[0.25u_c(t_k) - y(t_k) + x(t_k)\right]
\]

It can be realized by algorithm (where \(u_c\) is discrete)

\[
y := \text{adin}(\text{in2})\quad \{\text{read a process value}\}
\]

\[
u := 2 \times (0.25 \times \text{uc} - y + x)\quad \{\text{compute a control value}\}
\]

\[
dout(u)\quad \{\text{send out a control value}\}
\]

\[
x := x + h \left(1.5y - 2x\right)\quad \{\text{compute the new x value}\}
\]

Or discrete transfer function

\[
u(z) = 0.5u_c(z) - 2 \frac{z + 0.5h - 1}{z + 2h - 1} y(z)
\]

\[
zx(z) = x(z) + h \left[1.5y(z) - 2x(z)\right]
\]

Corresponds to continuous transfer function substitution

\[
s = \frac{z - 1}{h}
\]
Example: comparison

- Comparin continuous and discrete control for $h = 0.2$

$$u(z) = 0.5u_C(z) - 2 \frac{z - 0.9}{z - 0.6} y(z)$$
• Various sampling periods $h = 0.1, 0.5, 1, 1.5$
Example: another solution

- We find a discrete transfer function of a system and a shaper.
  \[ G(s) = \frac{1}{s^2} \rightarrow G(z) = \frac{h^2}{2} \frac{z + 1}{(z-1)^2} \]

- We use discreté methods for descreet controller design
  \[(z^2 - 2z + 1)p(z) + h^2/2(z + 1)q(z) = z^3\]

- Solve the equation
  \[
  \begin{bmatrix}
  p_0 & p_1 & q_0 & q_1 \\
  0 & 1 & -2 & 1 \\
  h^2/2 & h^2/2 & 0 & 0 \\
  0 & h^2/2 & h^2/2 & 0
  \end{bmatrix}
  \begin{bmatrix}
  1 & -2 & 1 & 0 \\
  0 & 1 & -2 & 1 \\
  h^2/2 & h^2/2 & 0 & 0 \\
  0 & h^2/2 & h^2/2 & 0
  \end{bmatrix}
  = \begin{bmatrix}
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]

- We get
  \[
  \begin{bmatrix}
  p_0 & p_1 & q_0 & q_1
  \end{bmatrix}
  = \begin{bmatrix}
  3/4 & 1 & -3/2h^2 & 5/2h^2
  \end{bmatrix}
  \]

  \[p(z) = 3/4 + z\]
  \[q(z) = -\frac{3}{2h^2} + \frac{5}{2h^2}z\]
Example: another solution

- This "pure discrete" controller

\[ u(z) = \frac{4}{7h^2}u_C(z) - \frac{5}{2h^2} \frac{z - 3/5}{z + 3/4} y(z) \]

- It gives a transfer function

\[ y(z) = \frac{2}{7} \frac{(z+1)(z+3/4)}{z^3} u_C(z) \]

- and CL characteristic polynomial

\[ c_{CL}(z) = z^3 \]

- simulation

ARI_20_2_AW_1_3.mdl

for \( h = 1.4 \)
- Simulation \texttt{ARI\_20\_2\_AW\_1\_3.mdl} for $h = 1.4$
- \textbf{output:} \hspace{2cm} \textbf{input:} \hspace{2cm} \textbf{speed:}

- The output value is the same as required value in the 4th sample step.
- This pure discrete solution is better than continuous and
- There is no parallel in continuous system.
- So what happens with decreasing $h$?
Example: second another solution

• Is it possible to decrease the number of steps even more? Apparently yes:
  \[
  \left( z^2 - 2z + 1 \right) p(z) + h^2/2(z + 1) q(z) = z^2(z + 1)
  \]

• Solve
  \[
  \begin{bmatrix}
  p_0 & p_1 & q_0 & q_1 \\
  1 & -2 & 1 & 0 \\
  0 & 1 & -2 & 1 \\
  h^2/2 & h^2/2 & 0 & 0 \\
  0 & h^2/2 & h^2/2 & 0 
  \end{bmatrix}
  =
  \begin{bmatrix}
  0 & 0 & 1 & 1
  \end{bmatrix}
  \]

• we get
  \[
  \begin{bmatrix}
  p_0 & p_1 & q_0 & q_1 
  \end{bmatrix}
  =
  \begin{bmatrix}
  1 & 1 & -2/h^2 & 4/h^2
  \end{bmatrix}
  \]

• controller
  \[
  u = \frac{1}{h^2} u_c - \frac{4}{h^2} \frac{z-1/2}{z+1} y
  \]

  \[
  y = \frac{1}{2} \frac{(z+1)}{z^2} u_c
  \]

  with CL transfer function

  and CL characteristic polynomial

  \[
  c_{CL}(z) = z^2(z + 1)
  \]
Simulation – second model in `ARI_20_2_AW_1_3.mdl` it looks OK

But for nonzero initial conditions reveals a problém.

Note that in moments of sampling behaves perfectly
Solution with state feedback

State equations of double integrator \( G(s) = 1/s^2 \) are

\[
\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \ y = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)
\]

Its discrete version (with ZOH and sampling period \( h \))

\[
x(k + 1) = \begin{bmatrix} 1 & h \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} h^2/2 \\ h \end{bmatrix} u(k), \ y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(k)
\]

State controller

\[
u(k) = -\begin{bmatrix} 1/h^2 \\ 3/(2h) \end{bmatrix} x(k) + 1/h^2 \ u_c(k)
\]

The system change to with polynomial \( c_{CL}(z) = z^2 \)

\[
x(k + 1) = \begin{bmatrix} 1/2 & 1/4h \\ -1/h & -1/2 \end{bmatrix} x(k) + \begin{bmatrix} 1/2 \\ 1/h \end{bmatrix} u_c(k), \ y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(k)
\]
Simulation

- Simulation `ARI_20_3_AW_4_5.mdl` for $h = 1.4$

- Starting from the third step, the set point is exactly set, and control is zero.
- And for each initial condition
- system is internally stable.