Computerized control - homework 2

Kjartan Halvorsen

Due 2016-02-05

Exercises

Sample the continuous-time transfer function

Consider the harmonic oscillator with transfer function

$$G(s) = \frac{\omega^2}{s^2 + \omega^2}. \tag{1}$$

Compute the pulse-transfer function by sampling the transfer function $G(s)$.

Simulation of the continuous- and discrete-time harmonic oscillator

Use matlab's control toolbox or the python control module to simulate the system and verify your calculations.

Define systems

First, define the continuous-time system in (1)

```
omega = 1; % Just a suggestion
h = 0.2/omega; % Completely undamped system. This gives about 30 samples per period
sys_c = tf([omega^2],[1 0 omega^2])
```

Sample the system using the function `c2d`

```
sys_c2d = c2d(sys_c, h)
```

Define the discrete-time system you calculated in the first part of the homework

```
den = [1 a1 a2];
num = [b1 b2];
sys_d = tf(num, den, h)
```

Verify that the two discrete-time systems `sys_c2d` and `sys_d` are identical.

Simulate step responses

Simulate for 4 complete periods

```
Tc = linspace(0, 4*(2*pi/omega), 800);
[yc,tc] = step(sys_c, Tc);

Td = h*(0:120);
```

```
[yd,td] = step(sys_d, Td);

figure()
clf
plot(tc,yc)
hold on
plot(td,yd, 'r*')
```

Verify that the step response of the discrete-time system is exactly equal to that of the continuous-time system at the sampling instants. Explain why this is so!

Compute the discrete step response yourself

Write some lines of code that solves the difference equation

$$y(k+2) = -a_1 y(k+1) - a_2 y(k) + b_1 u(k+1) + b_2 u(k)$$

for the harmonic oscillator. Use the initial state $y(-1) = y(0) = 0$ and compute the response to a step sequence

$$u(k) = \begin{cases} 1, & k \geq 0 \\ 0, & \text{otherwise} \end{cases}.$$

Verify that your solution is the same as when using the `step` function in the previous exercise in this homework.

Solutions

Sampling the transfer function

1. Calculate the step response

$$\begin{aligned} Y(s) &= G(s)\frac{1}{s} = \frac{\omega^2}{(s^2+\omega^2)s} \\ &= \frac{1}{s} - \frac{s}{s^2+\omega^2}. \end{aligned}$$

2. Transform to time domain (using transform table)

$$y(t) = u(t) - u(t)\cos\omega t$$

3. Calculate z-transform of sampled output

$$\begin{aligned} Y(z) &= \mathcal{Z}\{y(kh)\} = \mathcal{Z}\{u(kh) - u(kh)\cos(\omega kh)\} \\ &= \mathcal{Z}\{u(k)\} - \mathcal{Z}\{u(k)\cos(\omega hk)\} \\ &= \frac{z}{z-1} - \frac{z(z-\cos(\omega h))}{z^2 - 2\cos(\omega h)z + 1} \end{aligned}$$

4. Divide by the z-transform of the step input

$$\begin{aligned} H(z) = \frac{Y(z)}{U(z)} &= \frac{z-1}{z}\left(\frac{z}{z-1} - \frac{z(z-\cos(\omega h))}{z^2 - 2\cos(\omega h)z + 1}\right) \\ &= 1 - \frac{(z-1)(z-\cos(\omega h))}{z^2 - 2\cos(\omega h)z + 1} \\ &= \frac{z^2 - 2\cos(\omega h)z + 1 - z^2 + \cos(\omega h)z + z - \cos(\omega h)}{z^2 - 2\cos(\omega h)z + 1} \\ &= \frac{\big(1 - \cos(\omega H)\big)z + 1 - \cos(\omega h)}{z^2 - 2\cos(\omega h)z + 1}. \end{aligned}$$

Simulations

The code that is provided does most of the job. You just have to define the numerator and denominator of the discrete-time system from the sampled model you obtained:

```
a1 = -2*cos(omega*h);
a2 = 1;
b1 = 1-cos(omega*h);
b2 = b1;

den = [1 a1 a2];
num = [0 b1 b2];
sys_d = tf(num, den, h)
```

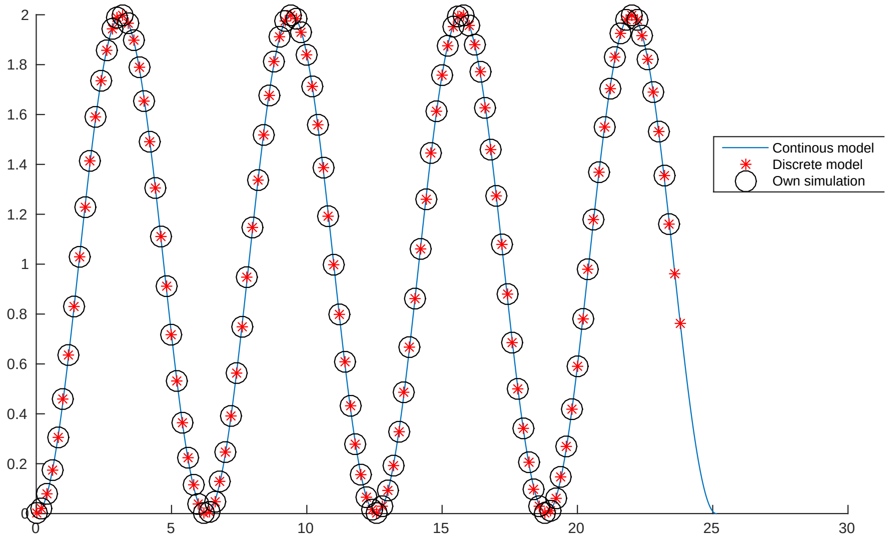
Own code for simulating system response

There are many ways to do this. Here is one

```
% Calculate the step response by hand
N = 120;
u = ones(N,1);
y = nan(N,1);
% Pad with two zeros at beginning, corresponding to u(-2), u(-1), y(-2) and
% y(-1)
u = [zeros(2,1); u];
y = [zeros(2,1); y];

for kplustwo = 3:N
    y(kplustwo) = -a1*y(kplustwo-1) -a2*y(kplustwo-2) + b1*u(kplustwo-1) + b2*u(kplustwo-2);
end

yd2 = y(3:end);
plot(td,yd2, 'ko', 'markersize', 14)
legend('Continous model', 'Discrete model', 'Own simulation')
```



2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
5
10
15
20
25
30
Continous model
Discrete model
Own simulation