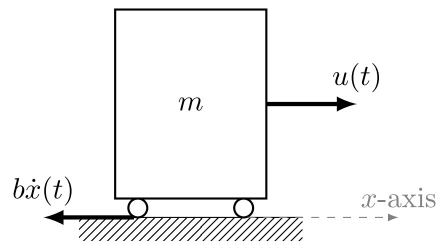
ELEC-E8101 Digital and Optimal Control

Homework 2 - Solution

1. A cruise control system of a car is often modeled (for simplicity) as a cart experiencing friction (with the average friction coefficient being b) and controller's action (denoted by u), as shown below:



a) Prove that the 2[nd] order ordinary differential equation (ODE) with respect to displacement x describing the evolution of the simplified cruise control system is given by

$$\dot{v}(t) + \frac{b}{m}v(t) = \frac{u(t)}{m},$$

where $v(t) = \dot{x}(t)$. [0.5p]

b) Consider a car, which has a weight $m = 1000$ kg. Assuming the average friction coefficient $b = 100$, design a cruise control system such that the car can reach 100 km/h from 0 km/h in 8 s with an overshoot less 20%.

i) Find a PI controller that achieves the design specifications. [0.5p]

ii) Verify the controller design via Simulink. [0.5p]

c) Design a discrete-time controller by discretizing the continuous-time controller with a sampling rate 30 times the bandwidth using the Tustin transformation and verify via Simulink that the specifications are met. [0.75p]

d) Design a discrete-time controller by

i) discretizing the continuous-time controller with a sampling rate 6 times the bandwidth using the Tustin transformation. Does the controller meet the specifications set? [0.75p]

ii) discretizing the plant with a sampling rate 6 times the bandwidth and designing a discrete-time PI controller. Does the controller meet the specifications set in this case? [1p]

Solution.

a) By the well-known Newton's Law of motion ($F = m\ddot{x}$):

$$m\ddot{x}(t) = u(t) - b\dot{x}(t) \Longrightarrow m\ddot{x}(t) + b\dot{x}(t) = u(t)$$

$$\Longrightarrow \ddot{x}(t) + \frac{b}{m}\dot{x}(t) = \frac{u(t)}{m}$$

$$\big(v(t) = \dot{x}(t)\big) \qquad \Longrightarrow \dot{v}(t) + \frac{b}{m}v(t) = \frac{u(t)}{m}.$$

b) Taking Laplace transform on both sides of the equation, we obtain

$$msV(s) + bV(s) = U(s) \Longrightarrow V(s) = \frac{1}{ms+b}U(s) \Longrightarrow G(s) = \frac{V(s)}{U(s)} = \frac{1}{ms+b}$$

The PI controller is $C(s) = K_p + K_i/s$. Hence, the closed-loop transfer function is

$$\begin{aligned} H(s) = \frac{C(s)G(s)}{1+C(s)G(s)} &= \frac{\frac{K_ps+K_i}{s}\frac{1}{ms+b}}{1+\frac{K_ps+K_i}{s}\frac{1}{ms+b}} \\ &= \frac{K_ps+K_i}{s(ms+b)+K_ps+K_i} \\ &= \frac{K_ps+K_i}{ms^2+(b+K_p)s+K_i} \\ &= \frac{K_ps+K_i}{1000s^2+(100+K_p)s+K_i} \\ &= \frac{0.001K_ps+0.001K_i}{s^2+(0.1+0.001K_p)s+0.001K_i} \end{aligned}$$

i) To find a PI controller, we first need to find ζ and ω_n from the design specifications.

First, we find a value for the damping ratio ζ:

$$\zeta \geq 0.6\left(1 - \frac{M_p \text{ in } \%}{100}\right) = 0.6\left(1 - \frac{20}{100}\right) = 0.48$$

Let's choose $\zeta = 0.7$.

Second, we find the natural frequency ω_n

$$t_s = \frac{4.6}{\zeta\omega_n} \Rightarrow \omega_n = \frac{4.6}{(0.7)8} \approx 0.82.$$

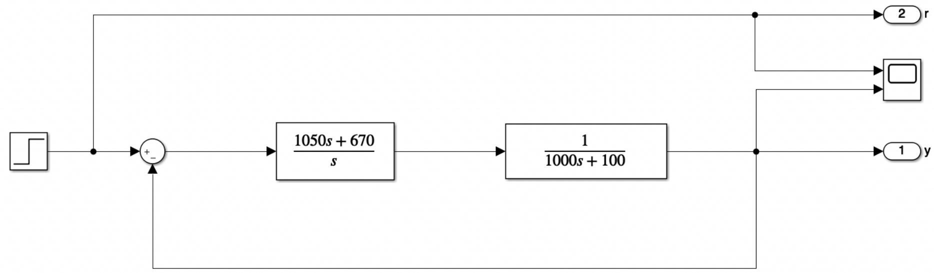
Hence, the *desired* closed-loop function $H(s)$ is given by

$$\begin{aligned} H(s) &= \frac{\omega_n^2}{s^2+2\zeta\omega_ns+\omega_n^2} \\ &= \frac{0.67}{s^2+1.15s+0.67} \end{aligned}$$

Comparing the coefficients in the denominator, we get

$$\begin{cases} 0.001K_i = 0.67 \Rightarrow K_i = 670 \\ 0.1 + 0.001K_p = 1.15 \Rightarrow K_p = 1050 \end{cases}$$

ii) In what follows, we verify the controller design via Simulink. We first build the system:



and then we run the simulation for a step of 100:

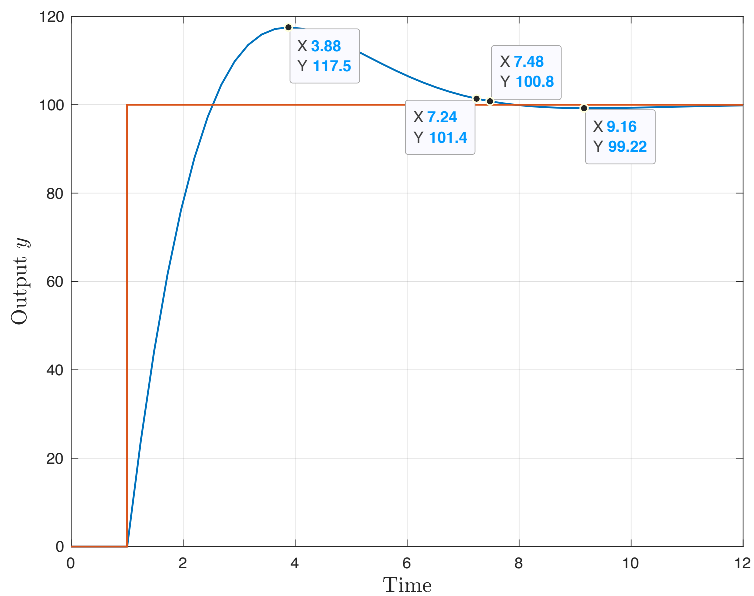


Figure 1: Overshoot is around 17.5% *and the settling time is around* 7.4*s*.

MATLAB Code:

```
% Solution code for Homework 2: Item (b) ii). 2019 %

simOut = sim('car',12);
outputs = simOut.get('yout');
y = (outputs.get('y').Values);
r = (outputs.get('r').Values);

figure(1);
set(gca,'FontSize',14);
plot(y, 'LineWidth',1.2); hold on; grid on;
xlabel('Time','fontsize',14,'interpreter','latex');
ylabel('Output $y$','fontsize',14,'interpreter','latex');
title('');
```

c) The bandwidth frequency is defined as the frequency at which the closed-loop magnitude drops 3 dB below its magnitude at DC (magnitude as the frequency approaches 0). In the following Bode plot we find the *closed-loop* bandwidth:

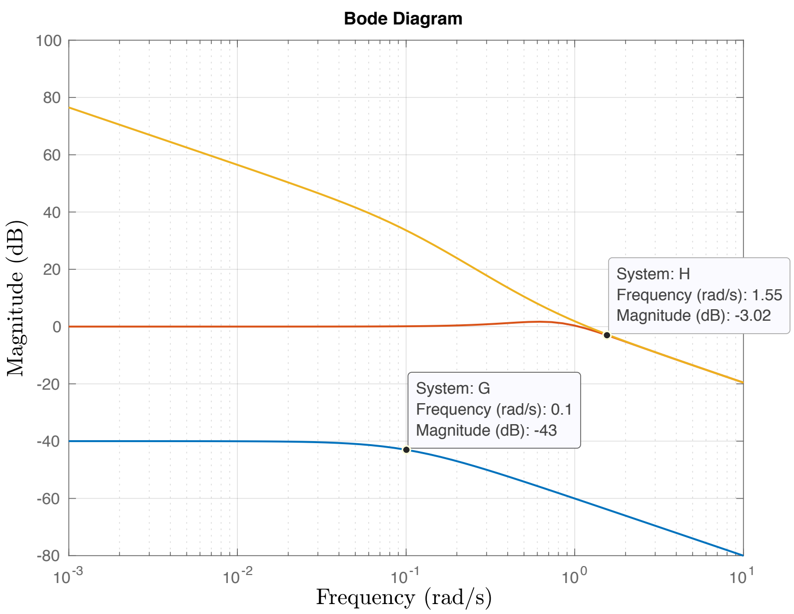


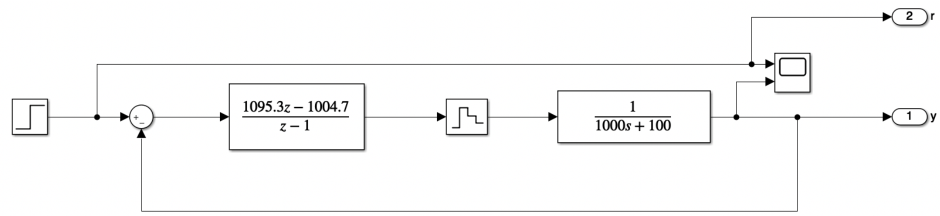
Figure 2: The blue line shows the bode plot of the plant, the yellow line the bode plot of the open-loop system, and the red line the bode plot of the closed-loop system. The bandwidth of the closed-loop system is around 1.55 *rad/sec* $= 0.2467Hz$.

```
% Solution code for Homework 2: Item (c). 2019 %

G = tf(1,[1000 100]);
C = tf([1050 670],[1 0]);
dbdrop = -3;
fb_G = bandwidth(G,dbdrop);
CG=C*G;
H=CG/(1+CG);
fb_H = bandwidth(H,dbdrop);

bodemag(G,H,CG);grid on;
set(findall(gcf,'type','line'),'linewidth',1.2)
xlabel('Frequency','fontsize',14,'interpreter','latex');
ylabel('Magnitude','fontsize',14,'interpreter','latex');
```

We first discretize the continuous-time PI control law with $T = 1/(30 \times 0.2467) = 0.1351$ seconds using a bilinear transformation method, i.e.,

$$C(z) = C(s)\Big|_{s=\frac{2}{T}\frac{z-1}{z+1}} = \frac{1050\left(\frac{2}{0.1351}\frac{z-1}{z+1}\right) + 670}{\frac{2}{0.1351}\frac{z-1}{z+1}} = \frac{1095.3z - 1004.7}{z-1}$$

In what follows, we check the performance of the controller design via Simulink. We first build the system:



Then, we run the simulation for a step of 100:

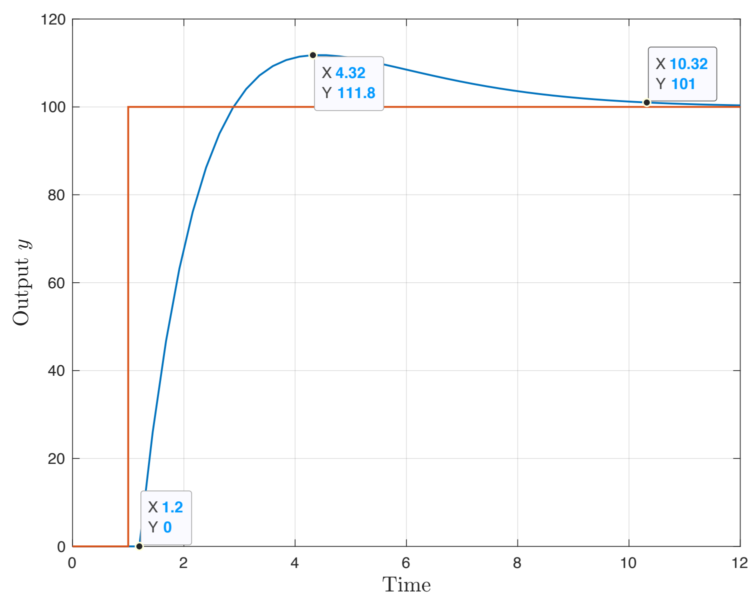
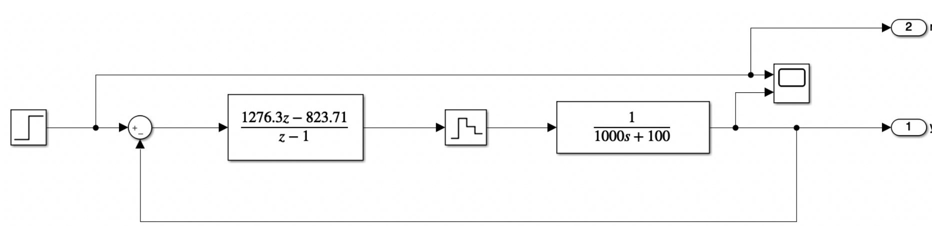


Figure 3: Overshoot is around 11.8%*, but the settling time is more than* 8*s.*

d) i) We first discretize the continuous-time PI control law with $T = 1/(6 \times 0.2467) = 0.6755$ seconds using a bilinear transformation method, i.e.,

$$C(z) = C(s)\Big|_{s=\frac{2}{T}\frac{z-1}{z+1}} = \frac{1050\left(\frac{2}{0.6755}\frac{z-1}{z+1}\right) + 670}{\frac{2}{0.6755}\frac{z-1}{z+1}} = \frac{1276.3z - 823.71}{z-1}$$

In what follows, we check the performance of the controller design via Simulink. We first build the system:



Then, we run the simulation for a step of 100:

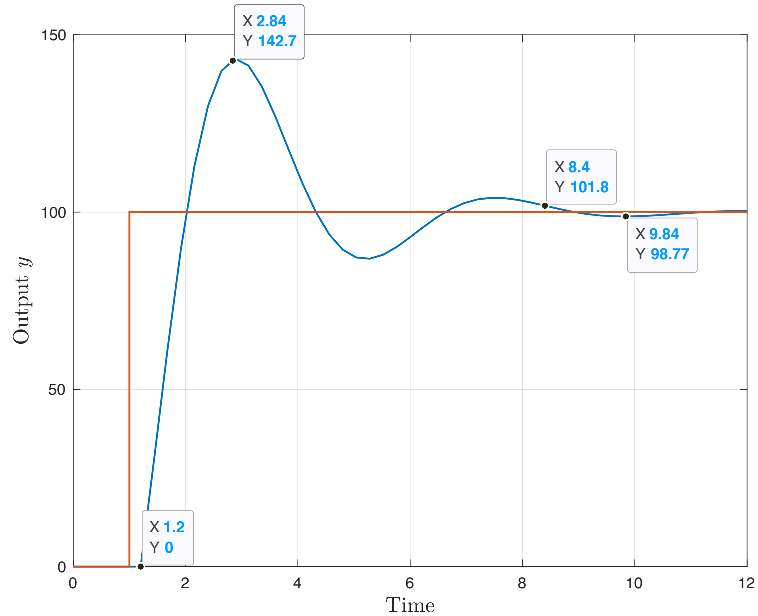


Figure 4: Overshoot is around 42.7% and the settling time is more than 10s.

ii) In this case, we discretize the continuous-time plant first and then design a digital controller. Since we want the system to meet certain specifications in the step response, we choose the step-invariance method, i.e.,

$$
\begin{aligned}
G(z) &= \frac{z-1}{z}\mathcal{Z}\left\{\frac{G(s)}{s}\right\} \\
&= \frac{z-1}{z}\mathcal{Z}\left\{\frac{0.01}{s(s+0.1)}\right\} \\
&= \frac{z-1}{z}\mathcal{Z}\left\{\frac{0.01}{s} - \frac{0.01}{(s+0.1)}\right\} \\
&= 0.01\frac{z-1}{z}\left(\frac{z}{z-1} - \frac{z}{z-e^{-(0.1\times 0.6)}}\right) \\
&= \frac{0.00058}{z-0.942}
\end{aligned}
$$

We choose a digital PI controller via bilinear transformation:

$$
C(z) = K_p + \frac{K_i}{s}\bigg|_{s=\frac{2}{T}\frac{z+1}{z-1}} \equiv \frac{a_0 z - a_1}{z-1}
$$

The resulting closed-loop transfer function from r to y is given by

$$
\begin{aligned}
H(z) &= \frac{C(z)G(z)}{1+C(z)G(z)} = \frac{\frac{a_0 z - a_1}{z-1}\frac{0.00058}{z-0.942}}{1+\frac{a_0 z - a_1}{z-1}\frac{0.00058}{z-0.942}} \\
&= \frac{0.00058(a_0 z - a_1)}{z^2 + (0.00058a_0 - 1.942)z + (0.942 - 0.00058a_1)} \qquad (1)
\end{aligned}
$$

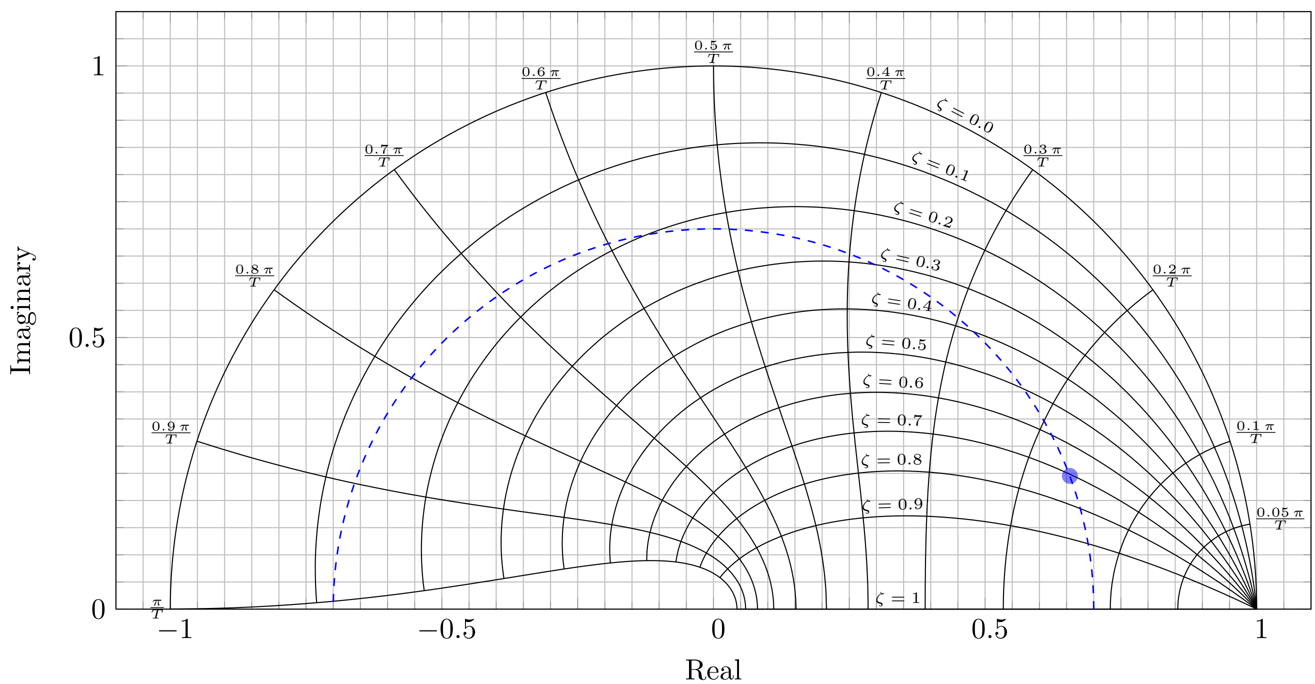
From the design in part b) i), we obtained the desired parameters $\zeta = 0.7$ and $\omega_n = 0.82$ in continuous setting. In the discrete setting, we have:

$$\zeta \geq 0.6\left(1 - \frac{M_p \text{ in } \%}{100}\right) = 0.6\left(1 - \frac{20}{100}\right) = 0.6(0.8) = 0.48,$$

as before. For consistency, let's choose $\zeta = 0.7$ as before.
The condition for the settling time, with $T = 0.6$, gives us

$$|z| < 0.01^{T/t_s} = 0.01^{0.6/8} = 0.7079.$$

We use this information on the following chart: we draw a dashed arc of radius 0.7 (< 0.7079) and follow the line of $\zeta = 0.7$.



The intersection gives us the desired pole location: $p = 0.65 \pm j0.25$. Hence, the characteristic equations of $H(z)$ is:

$$\begin{aligned}
\chi(z) &= \big(z - (0.65 - j0.25)\big)\big(z - (0.65 + j0.25)\big) \\
&= z^2 - z(0.65 - j0.25) - z(0.65 + j0.25) + (0.65 - j0.25)(0.65 + j0.25) \\
&= z^2 - 1.3z + 0.65^2 + 0.25^2 \\
&= z^2 - 1.3z + 0.485 \qquad (2)
\end{aligned}$$

Equating the coefficients of the denominator of (1) and the coefficients of (2), we get:

$$\begin{cases} 0.00058a_0 - 1.942 = -1.3 \Rightarrow 0.00058a_0 = 1.942 - 1.3 = 0.642 \Rightarrow \underline{a_0 = 1106.9} \\ 0.942 - 0.00058a_1 = 0.485 \Rightarrow 0.00058a_1 = 0.942 - 0.485 = 0.457 \Rightarrow \underline{a_1 = 787.931} \end{cases}$$

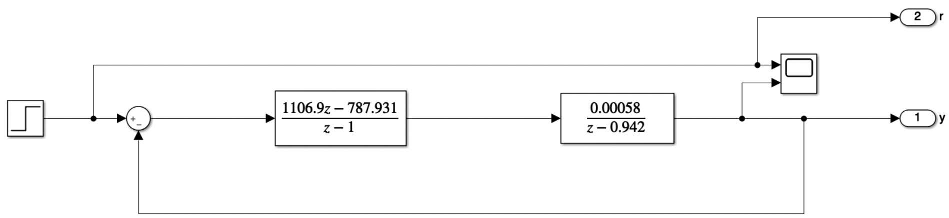
Therefore, the PI controller $C(z)$ is given by

$$C(z) = \frac{1106.9z - 787.931}{z - 1}$$

and the closed loop system $H(z)$ is given by

$$H(z) = \frac{0.642z - 0.457}{z^2 - 1.3z + 0.485}$$

We simulate the digital controller with the discretized plant to see whether the specifications are fulfilled in the discrete-time setting:



We run the simulation for a step of 100:

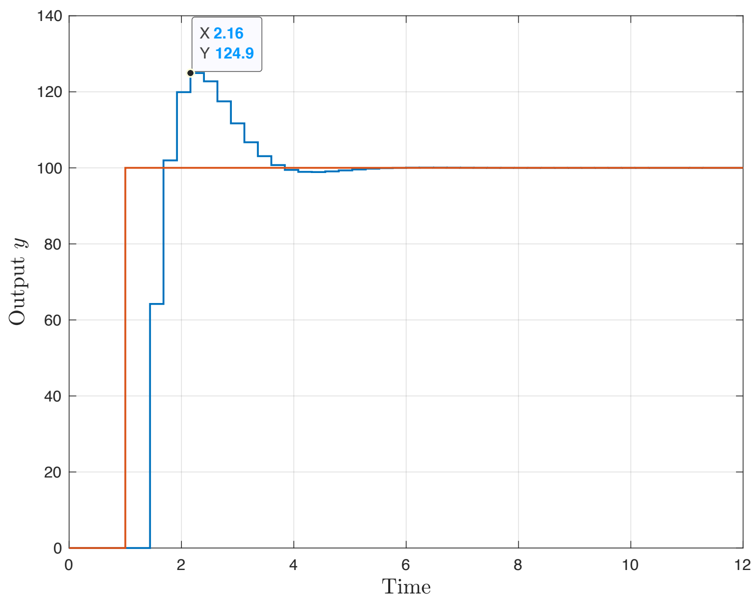


Figure 5: Overshoot is around 24.9%. The settling time is (a lot) less than 8s.