Homework #4

Reading: Chapter 4

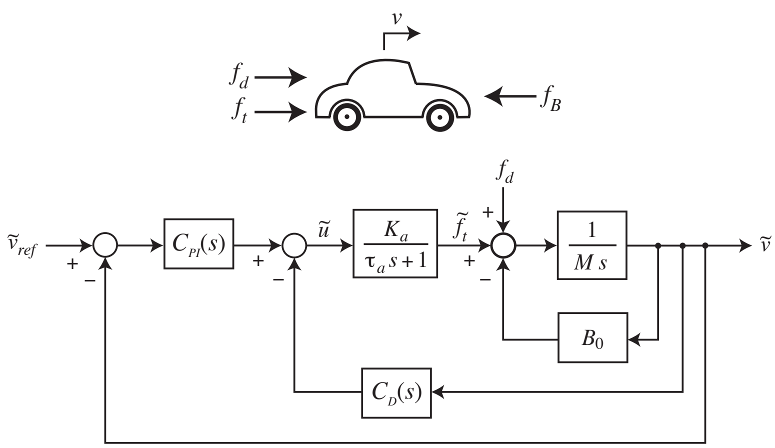


Figure 1. Speed control system.

1. The small-perturbation dynamics of the car speed control system that you dealt with in a previous homework assignment are represented in Figure 1. Here:

$v_0 = 25$ m/sec

$K_a = 1599$ N

$\tau_a = 0.5$ sec

$M = 1670$ kg

$B_0 = 27.80$ N/m/sec

$g = 9.806$ m/sec/sec (local acceleration due to gravity)

The proportional-integral control law is

$$C_{PI}(s) = \frac{K_P s + K_I}{s}$$

with

$K_P = 0.6$ 1/m/sec

$K_I = 0.01$ 1/m

The derivative control law is

$$C_D(s) = K_D s$$

with

$K_D = 0.08$ 1/m/sec^2

The sampled-data version of the Figure 1 system is shown in Figure 2.

Overshoot be higher, dynamic system respond slower.
Owing to discrete data sampling.
Got to change CTS to DTS.

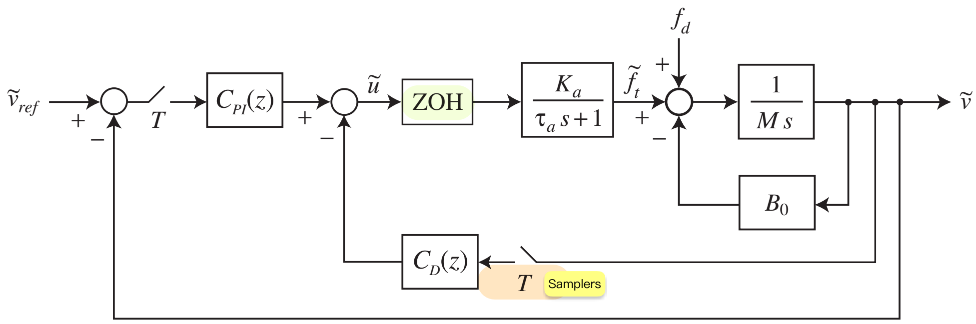


Figure 2. Speed control system.

(a) Using these finite-difference-based approximations to continuous-time proportional, integral, and derivative control laws:

$$u_P(k) = K_P\, e(k) \qquad u_I(k) = u_I(k-1) + K_I\, T\, e(k) \qquad u_D(k) = K_D\left[\frac{e(k) - e(k-1)}{T}\right]$$

Error

where T is the sampling period, create discrete-time controller transfer function systems $C_{PI}(z)$ and $C_D(z)$ for the Figure 2 system that correspond to the continuous-time controller transfer functions $C_{PI}(s)$ and $C_D(s)$ of the Figure 1 system.

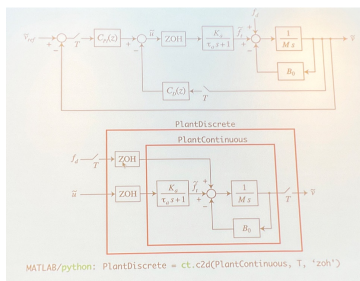
(b) Use Matlab or Python-control to create state-space models of the interconnected systems shown in Figures 1 and 2. In both cases, use the following as the input vector

$$u = \begin{bmatrix} \tilde{v}_{ref} \\ f_d \end{bmatrix}$$

and the output vector is

$$y = \begin{bmatrix} \tilde{v} \\ \tilde{u} \end{bmatrix}$$

MATLAB: c2d to connect the system



(c) Superimpose on one plot the $\tilde{v}$ responses of the Figure 1 system to a 1 m/sec $\tilde{v}_{ref}$ step and the discrete-time $\tilde{v}$ response of the Figure 2 system to the same input. For the Figure 2 system use $T = 2$ sec for the sampling period. Plot the Figure 2 system's response with symbols (rather than with a continuous curve) to emphasize that it is a discrete-time signal.

(d) Superimpose on one plot the $\tilde{v}$ responses of the Figure 1 system to a long hill that has a constant 5 percent grade and the discrete-time $\tilde{v}$ response of the Figure 2 system to the same input. For the Figure 2 system use $T = 2$ sec for the sampling period. Plot the Figure 2 system's response with symbols to emphasize that it is a discrete-time signal. In both cases, assume that the car starts up the hill at time $t = 0$ with $v = v_0$ and $dv/dt = 0$.

(e) Use a z-plane plot of the zeros and poles of the appropriate transfer function to explain why the peak $\tilde{v}$ overshoot of the Figure 2 system that you obtained for part (c) is what it is. Show also, on your z-plane plot, the $z = e^{sT}$ mapping of the zeros and poles of the corresponding Figure 1 system transfer function.

(f) Use a z-plane plot of the zeros and poles of the appropriate transfer function to explain why the time constant of the exponential decay of the $\tilde{v}$ response of the Figure 2 system that you obtained for part (c) is what it is. Show also, on your z-plane plot, the $z = e^{sT}$ mapping of the zeros and poles of the corresponding Figure 1 system transfer function.

(g) Use a z-plane plot of the zeros and poles of the appropriate transfer function to explain why the time constant of the exponential decay of the $\tilde{v}$ response of the Figure 2 system that you obtained for part (d) is what it is. Show also, on your z-plane plot, the $z = e^{sT}$ mapping of the zeros and poles of the corresponding Figure 1 system transfer function.

Note: The Matlab/python-control commands

```
poles = pole(sys)
zeros = zero(sys)
```

return the `zeros` and `poles` of the transfer function `sys`. Choose an input-output pair using `sys(output_num, input_num)` (Matlab) or `sys[output_num, input_num]` (Python).

Hint 1: For 1e, 1f, and 1g) "explain why the step response is what it is" means to describe the response to the locations of poles and zeros.

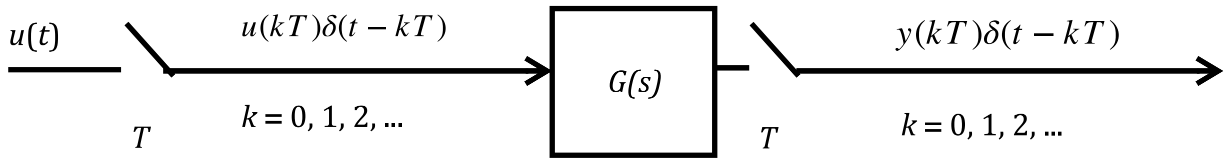
Hint 2: plotting pole and zero locations

if python add this

```
poles = ct.pole(sys) # s- or z-plane poles, depending on sys type
plt.plot(np.real(poles), np.imag(poles), 'x') # use 'o' for zeros
```

or

```
ct.pzmap(sys)
```



2. A continuous-time transfer function $G(s)$ consists of a pole pair at

$$p = -0.3 \pm 1.2j.$$

Suppose it is preceded by, and followed by, a sampler operating at intervals of

$$T = \pi$$

seconds, as shown above. The input sampler produces a continuous-time signal consisting of a stream of delta functions ("impulse functions") separated by time intervals T, each of whose area is equal to the magnitude of the input signal at the sampling instants. The sampler at the output performs the same operation.

This forms the "impulse-sampled" system $G^*(s)$, which can be represented exactly by the discrete-time transfer function $G(z)=G^*(s)|_{s = \ln(z)/T}$

a) Give the pole locations of $G(z)$
b) What is the natural frequency of the pole pair in $G(z)$?
c) What is the damping ratio of the pole pair in $G(z)$?
d) What is the time constant of the pole pair in $G(z)$?
e) What is the expected percent overshoot in response to a step input?

3. The system

$$G(z) = \frac{1}{z + 0.5},\ dt = 0.1$$

is excited by a sinusoid with a frequency $\omega = 20$ rad/s that is sampled at intervals equal to its dt.
a) Is this system stable?
b) What is the magnitude of the output sampled sinusoid in steady-state?

Potentially useful information:

Time-Domain Performance Parameters Using the Simple Oscillator Model

Performance Parameter	**Expression**
Rise Time	$T_r = \frac{\pi - \phi}{\omega_d}$ with $\cos\phi = \zeta$
Peak Time	$T_p = \frac{\pi}{\omega_d}$
Peak Value	$M_p = 1 - e^{-\pi\zeta/\sqrt{1-\zeta^2}}$
Percentage Overshoot (PO)	$PO = 100\, e^{-\pi\zeta/\sqrt{1-\zeta^2}}$
Time Constant	$\tau = \frac{1}{\zeta\omega_n}$
Settling Time (2%)	$T_s = -\frac{\ln\left[0.02\sqrt{1-\zeta^2}\right]}{\zeta\omega_n} \approx 4\tau = \frac{4}{\zeta\omega_n}$

Simple Oscillator Model:

$$\ddot{y} + 2\zeta\omega_n\dot{y} + \omega_n^2 y = \omega_n^2 u \iff u \longrightarrow \boxed{\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}} \longrightarrow y$$

$$\omega_d = \omega_n\sqrt{1-\zeta^2}$$