All students should complete the following problems:

1. Consider the block diagram for the following second order system
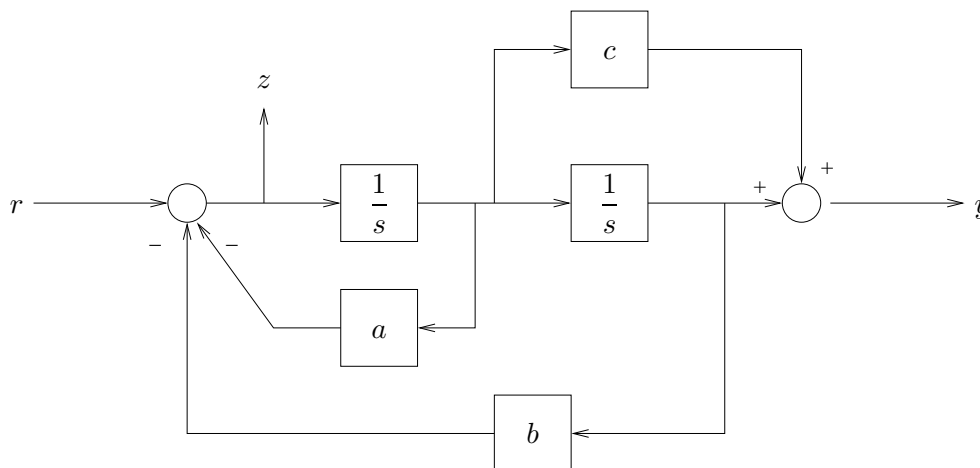
   (a) Compute the transfer function H_{yr} between the input r and the output y.

   (b) Show that the following state space system has the same transfer function, with the appropriate choice of parameters:

   $$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r$$

   $$y = \begin{bmatrix} b_2 & b_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + dr$$

   Give the values of a_i, b_i, and d that correspond to the transfer function you computed in (a).

   (c) Compute the transfer function H_{zr} between the input r and the output z. (Hint: it is *not* $H_{zr} = 1$.)

2. Consider the cruise control system from homework #2. The equations of motion for the system were given by

   $$m\dot{v} = -bv + K\tau + F_{\text{hill}}$$
   $$\dot{\tau} = -a\tau + u_e \tag{1}$$

   (see HW #2 for details on the variables and parameters). The simplest controller for this system is a proportional control, $u_e = -K_p e$, where $e = (v - r)$ (r is the reference speed).

   (a) Draw a block diagram for the system, with the engine dynamics and the vehicle dynamics in separate blocks and represented by transfer functions. Label the reference input to the closed loop system as r, the disturbance due to the hill as d, and the output as y ($= v$).

   (b) (MATLAB) Construct the transfer functions H_{er} and H_{yd} for the closed loop system and use MATLAB to generate the step response and frequency response for the each. Make sure to use the transfer function computation.

(c) Consider a more sophisticated control law of the form

$$\dot{x}_c = e$$
$$y_c = K_i x_c + K_p e.$$

This control law contains an "integral" term, which uses the controller state x_c to integrate the error. Compute the transfer functions for this control law and redraw your block diagram from part (a) with the default controller replaced by this one.

(d) (MATLAB) Using the default gains from previous homeworks ($K_i = 100$, $K_p = 500$), use MATLAB to compute the transfer function from r to y and plot the step response and frequency response for the system. This should match your answers in previous homework sets, but make sure to use the transfer function computation.

(e) Using the results from part (c), compute the bandwidth and steady state error of your control system (from r to y).

Only CDS 110a/ChE 105 students need to complete the following additional problems:

3. In this exercise we will explore the use of state feedback to generate a cruise controller for the system (1) in problem 2.

(a) Construct the linear state space system corresponding to the open loop system, in the form $\dot{x} = Ax + Bu$. Verify that the system is controllable.

(b) Design a pole placement control law that places the poles of the closed loop system at -1 and -2. Plot the step response and frequency response of the closed loop system with this control law.

(c) Design an LQR control law that minimizes the quadratic cost

$$J = \int_0^\infty \left(\|x\|^2 + \rho u^2 \right) dt$$

where ρ is chosen as 1, 10, and 100. Plot the step and frequency responses for each of these weights.

(d) In one paragraph, describe your impressions of the different control laws in problems 2 and 3. Is one better than the others?

4. (MATLAB) Consider the Caltech ducted fan example from homework #4. The linearized dynamics of the system were given in the solutions.

(a) Use MATLAB to compute the transfer function between the input u_2 and the vertical position of the fan, y. Compute the poles and zeros of the transfer function.

(b) Consider a unity feedback control system with controller given by

$$C(s) = \frac{k}{s + a}$$

Choose $k = 10$ and $a = 50$ and show that the resulting closed loop system is stable (by computing the poles of the closed loop transfer function).

(c) Plot the frequency response for your controller, from the reference input r to the output y. Compute the steady state error and bandwidth of your controller from the Bode plot. Plot the step response for your controller (use the MATLAB `step` command).

(d) By modifying the default gains, design a controller that has two times smaller steady state error and two times greater bandwidth (you should actually be able to do *much* better than 2X). Plot the frequency response and step response of the modified system.