A Design Study Approach to Classical Control

Randal W. Beard Timothy W. McLain
Brigham Young University

Updated: December 1, 2016

Homework B.18

For this homework assignment we will use the loopshaping design technique to design a successive loop closure controller for the inverted pendulum.

(a) First consider the inner loop, where $P_{in}(s)$ is the transfer function of the inner loop derived in HW B.5. Using a proportional and phase-lead controller, design $C_{in}(s)$ to stabilize the system with a phase margin close to 60 deg, and to ensure that noise above $\omega_{no} = 200$ rad/s is rejected by $\gamma_{no} = 0.1$. We want the closed-loop bandwidth of the inner loop to be approximately 25 rad/s. Note that since the gain on $P_{in}(s)$ is negative, the proportional gain will also need to be negative.

(b) Now consider the design of the controller for the outer loop system. The 'plant' for the design of the outer loop controller is

$$P = P_{out} \frac{P_{in}C_{in}}{1 + P_{in}C_{in}}.$$

Design the outer loop controller C_{out} so that the system is stable with phase margin close to $PM = 60$ degrees, and so that reference signals with frequency below $\omega_r = 0.01$ radians/sec are tracked with error $\gamma_r = 0.0001$, and noise with frequency content above $\omega_{no} = 400$ radians/sec are rejected with $\gamma_{no} = 0.001$. Design the crossover frequency so that the rise time of the system is about 2 sec.

Solution

Figure 2 shows the Bode plot of the plant $P_{in}(s)$ together with the design specification on the noise attenuation.

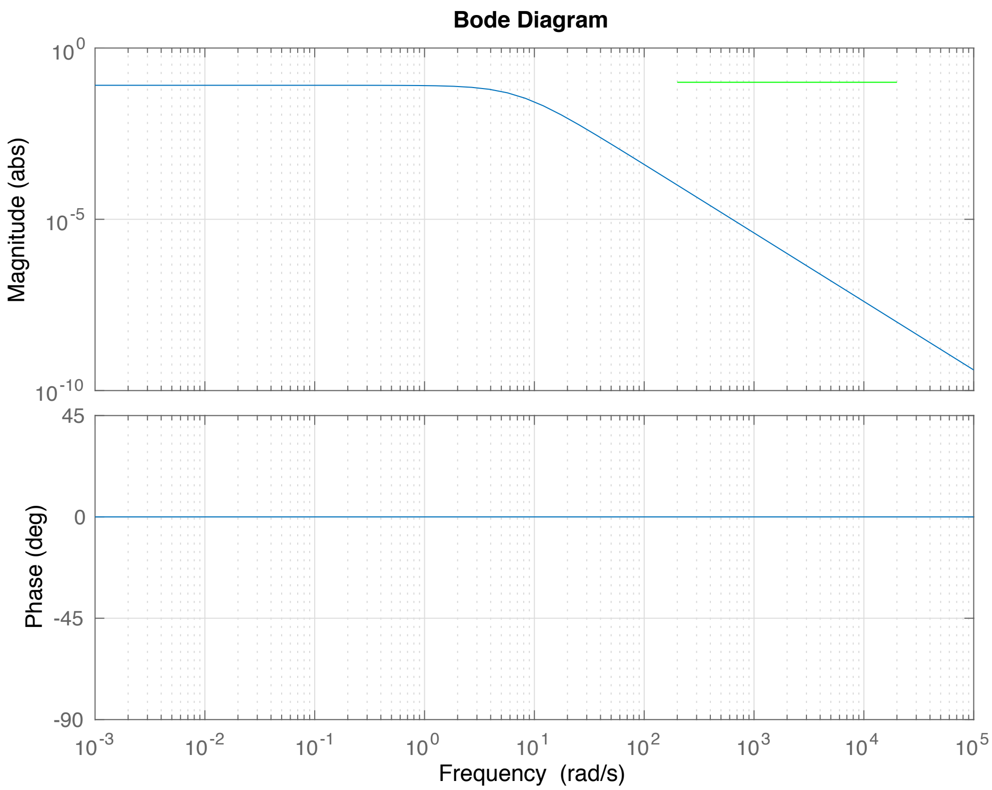

Figure 1: The Bode plot for the inner loop plant in HW B.18, together with the design specification.

The first step is to add a negative proportional gain of 1 to account for the negative sign in the plant transfer function. Once this is applied we can see that the phase fo the plant alone is -180 deg over all frequencies so that the open loop response at low frequency is above 0 dB. Figure 2 shows the corresponding Bode plot for a proportional gain of $K = -1$.

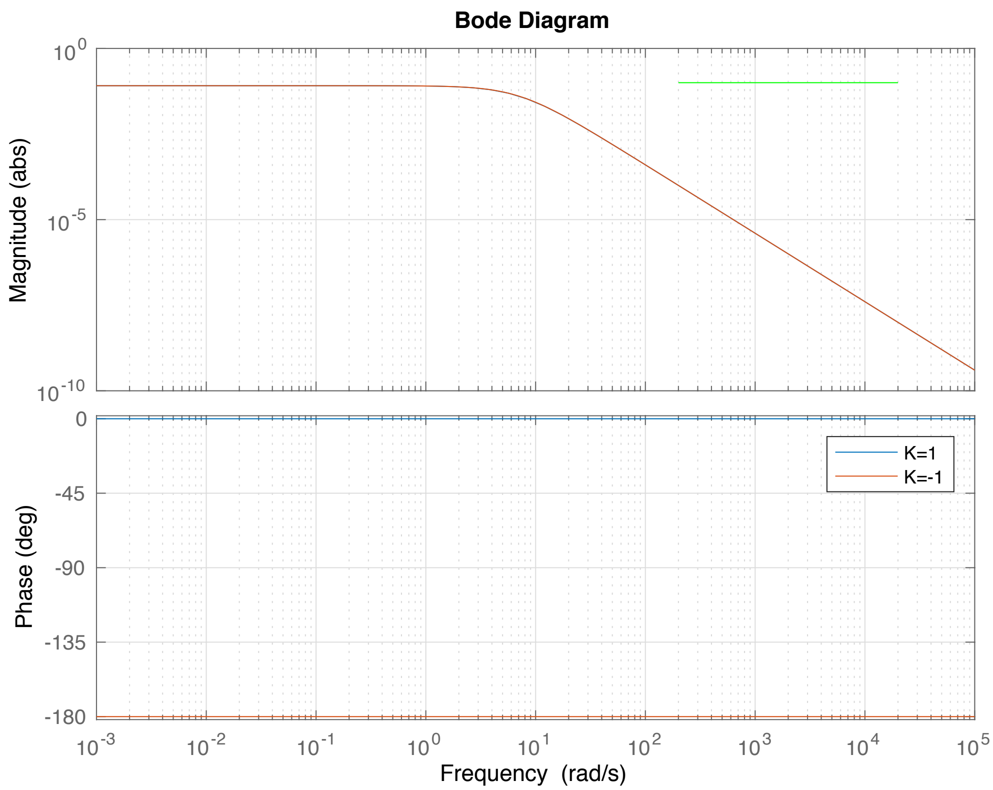

Figure 2: The Bode plot for the inner loop plant in HW B.18, with gain of -1 applied.

We want the bandwidth of the inner loop to be approximately 25 rad/s and so we will set the crossover frequency to be 25 rad/s. To stabilize the system, we will add lead compensation centered at 25 rad/s with a lead ratio of 13.9 that corresponds to the desired phase margin of 60 deg. Figure 3 shows the addition of the phase lead filter

$$C_{lead} = -45.1\left(\frac{\frac{s}{6.70}+1}{\frac{s}{93.3}+1}\right),$$

which has a phase margin of $PM = 60$ degrees. The gain 45.1 sets the crossover frequency to be 25 rad/s. The lead compensation alone also satisfies the noise specification.

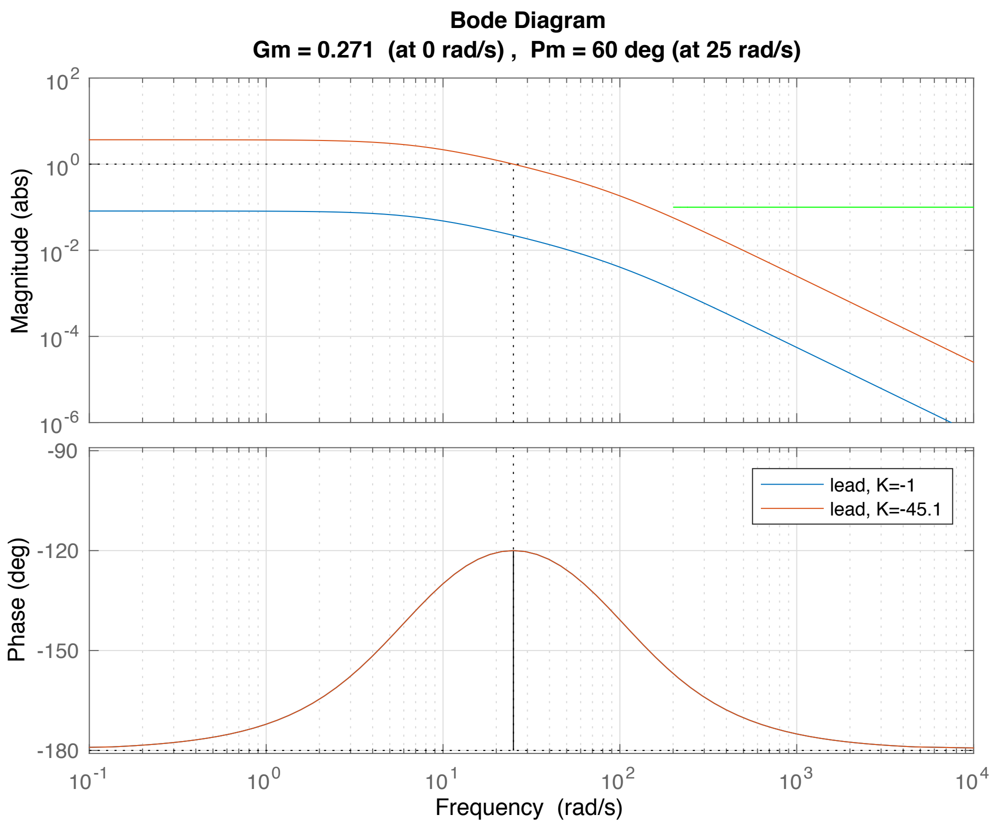

Figure 3: The Bode plot for the inner loop system in HW B.18, with proportional gain and phase lead compensation.

Notice that the DC-gain is not equal to one. The closed-loop magnitude response for the inner-loop subsystem

$$\frac{P_{in}C_{in}}{1+P_{in}C_{in}},$$

as well as the unit step response for the output and control signal are all show in Figure 4.

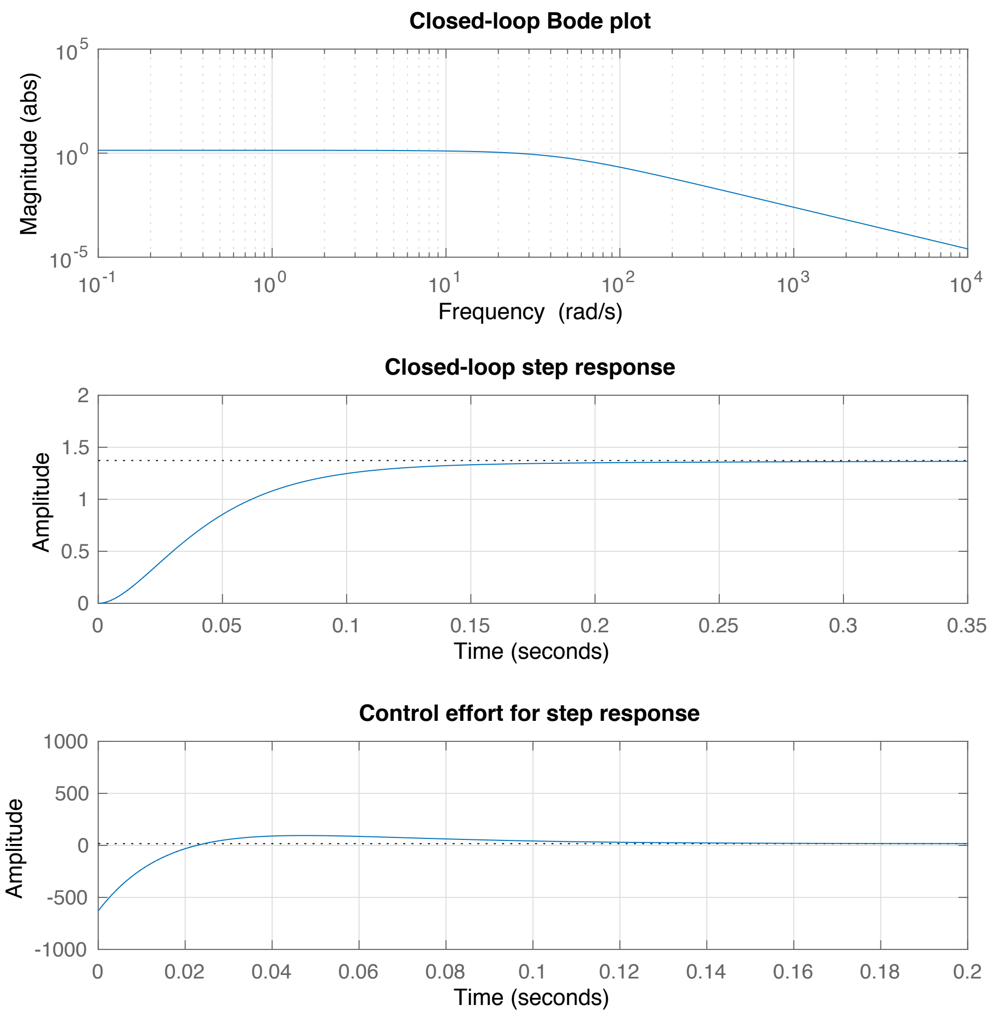

Figure 4: The closed loop bode response, the unit step response for the output, and the unit step response for the input of the inner loop design in HW B.18.

The Matlab code used to design the inner loop is shown below.

```
param

Plant = P_in;
figure(2), clf, hold on

```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Define Design Specifications
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %--- noise specification ---
        omega_n = 200;   % attenuate noise above this frequency
        gamma_n = 0.1;   % attenuate noise by this amount
        w = logspace(log10(omega_n),2+log10(omega_n));
        plot(w,gamma_n*ones(size(w)),'g')
        figure(2), bode(Plant,logspace(-3,5)), grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Control Design
  C = 1;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% proportional control: correct for negative sign in plant
     K = -1;
     C = C*K;
     figure(2), bode(Plant*C)
     legend('K=1','K=-1')

% phase lead: increase PM (stability)
    w_max = 25; % location of maximum frequency bump (desired crossover)
    phi_max = 60*pi/180;
    M = (1+sin(phi_max))/(1-sin(phi_max)); % lead ratio
    z = w_max/sqrt(M)
    p = w_max*sqrt(M)
    Lead = tf([1/z 1],[1/p 1]);
    C = C*Lead;
    figure(2), bode(Plant*C), hold on, grid % update plot

% find gain to set crossover at w_max = 25 rad/s
    [m,p] = bode(Plant*C,25);
    K = 1/m;
    C = K*C;
    figure(2), margin(Plant*C)  % update plot
    legend('lead, K=-1','lead, K=-45.1')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Create plots for analysis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Open-loop tranfer function
  OPEN = Plant*C;
% closed loop transfer function from R to Y
```

```
  CLOSED_R_to_Y = minreal((Plant*C/(1+Plant*C)));
% closed loop transfer function from R to U
  CLOSED_R_to_U = minreal((C/(1+C*Plant)));

figure(3), clf
    subplot(3,1,1),
        bodemag(CLOSED_R_to_Y)
        title('Closed-loop Bode plot'), grid on
    subplot(3,1,2),
        step(CLOSED_R_to_Y)
        title('Closed-loop step response'), grid on
    subplot(3,1,3),
        step(CLOSED_R_to_U)
        title('Control effort for step response'), grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Convert controller to state space equations for implementation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    [num,den] = tfdata(C,'v');
    [P.Ain_C,P.Bin_C,P.Cin_C,P.Din_C]=tf2ss(num,den);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Convert controller to discrete transfer functions for implementation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    C_in_d = c2d(C,P.Ts,'tustin')
    [P.Cin_d_num,P.Cin_d_den] = tfdata(C_in_d,'v');

    C_in = C;
```

For the outer loop design, Figure 5 shows the Bode plot of the plant $P = P_{out}\frac{P_{in}C_{in}}{1+P_{in}C_{in}}$ together with the design specification on reference tracking and noise attenuation.

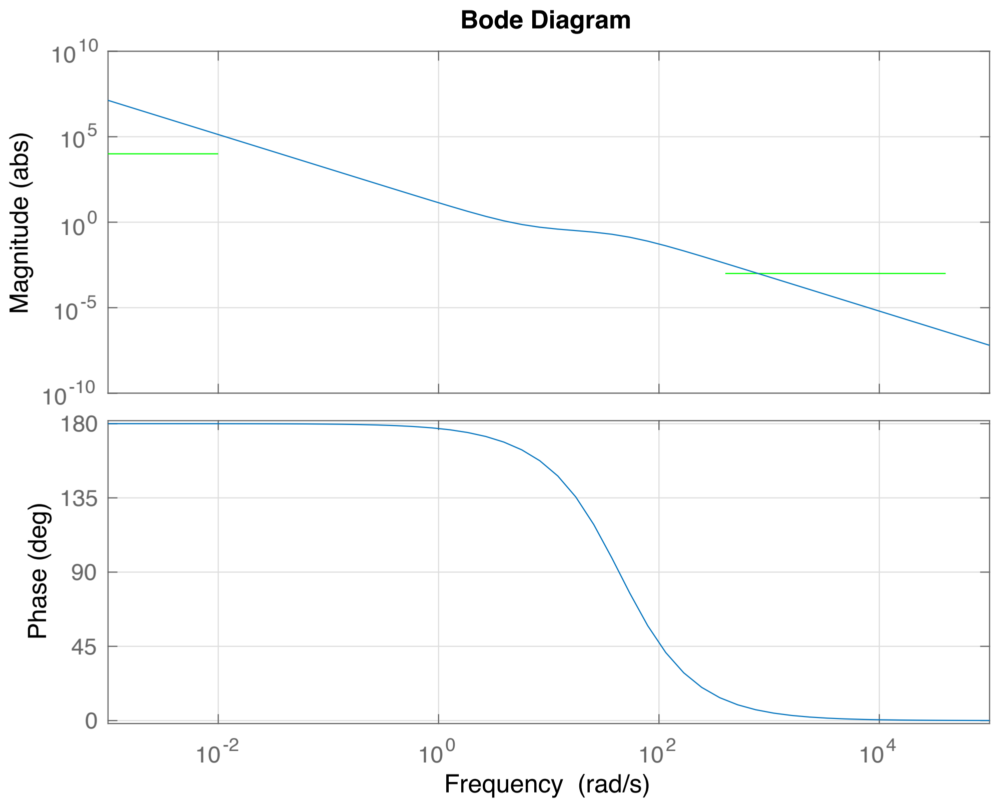

Figure 5: The Bode plot for the outer loop plant in HW B.18, together with the design specification.

We want a rise time of 2 sec for the system, so this leads us to select a crossover frequency of 1.1 rad/s. At 1.1 rad/s, the plant alone has a small negative phase margin (remember that 180 deg and −180 deg are equivalent), so to bring the phase margin up above 60 deg, we will use a lead compensator centered at 1.1 rad/s with a lead ratio of 32 to add 70 deg of phase. (A lead ratio of 32 is large. An alternative would be to add two leads that each contribute 35 deg of phase margin.) As shown in Figure 6, with the lead applied, the crossover frequency is pushed out beyond 100 rad/s. By applying a proportional gain of 0.0154, we set the crossover at 1.1 rad/s.

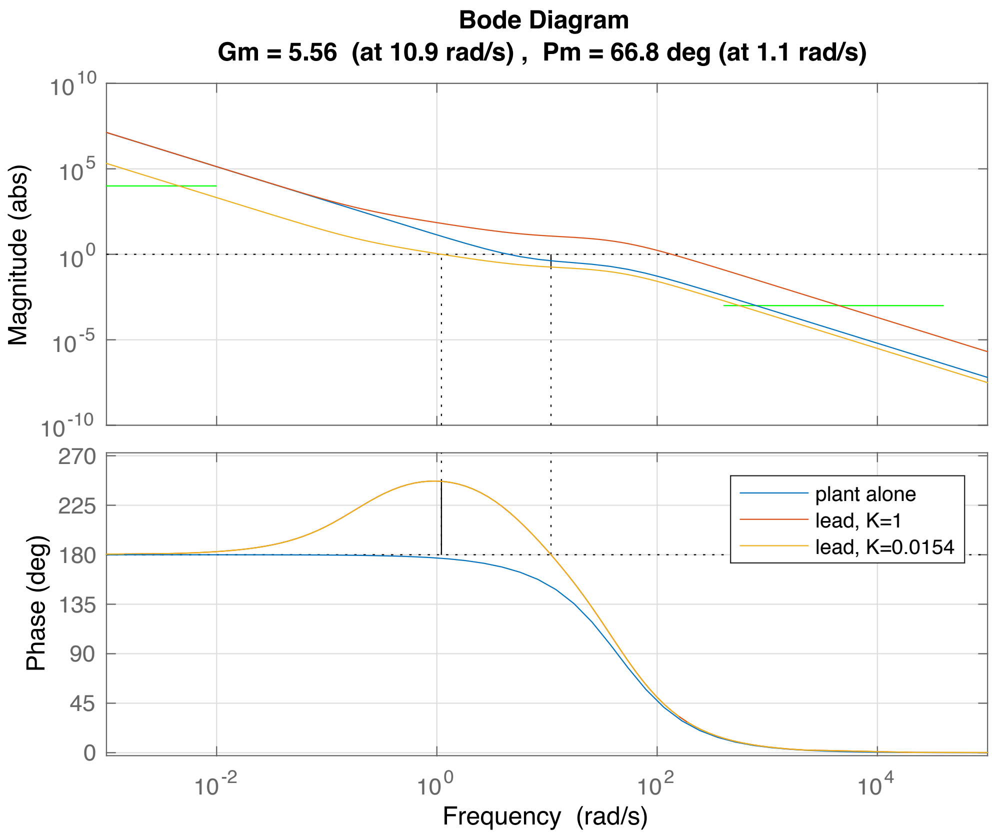

Figure 6: The Bode plot for the outer loop system in HW B.18, with lead compensation and proportional gain.

As can be seen from Figure 6, with the lead applied and crossover set at 1.1 rad/s, neither the tracking constraint nor the noise attenuation constraint are satisfied. We can satisfy the tracking constraint by applying lag compensation to boost the low-frequency gain near the constraint. Checking the gain of the lead compensated system at the tracking constraint frequency (ω_r = 0.01 rad/s), we can see that the gain is too low by a factor of 4.8. Applying a lag compensator with a lag ratio of 8 ensures that this tracking constraint is satisfied as shown in Figure 7.

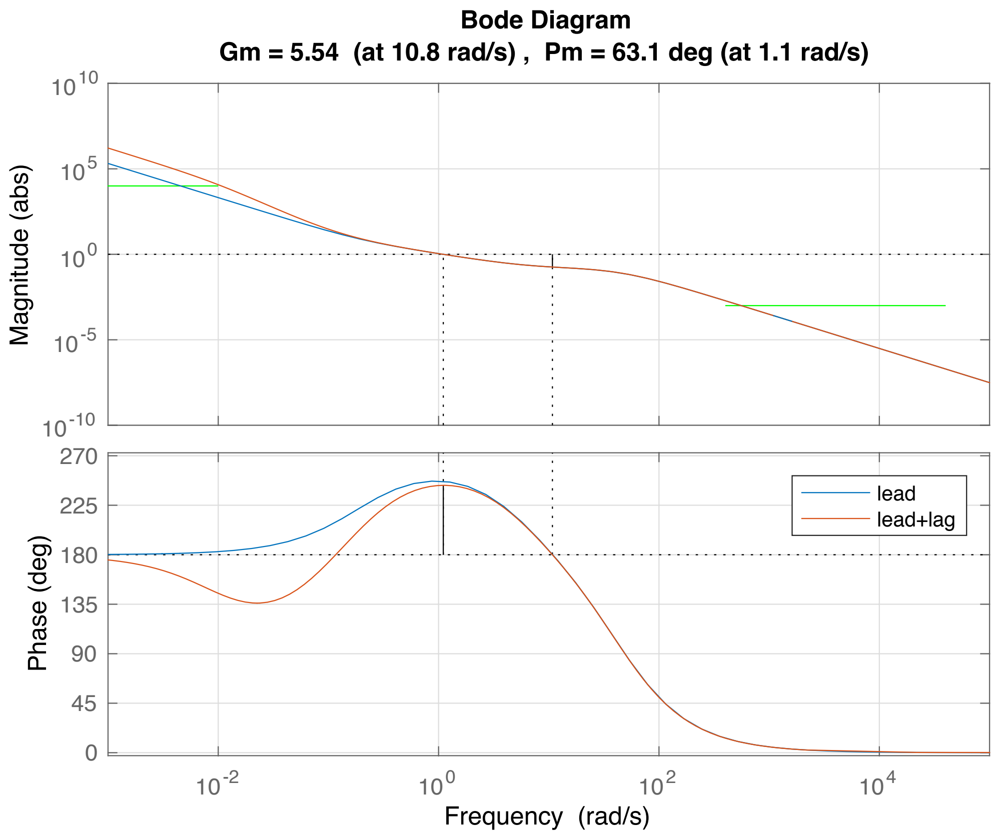

Figure 7: The Bode plot for the outer loop system in HW B.18, with lead, and lag control.

Finally, we can apply a low-pass filter to reduce the effects of sensor noise in the system and satisfy the noise attenuation constraint. To satisfy the constraint, we need to drop the gain of the open-loop system by a factor of 2 at $\omega_{no} = 400$ rad/s. We can accomplish this with a low-pass filter with a cut-off frequency of 230 rad/s. This is shown in Figure 8 below.

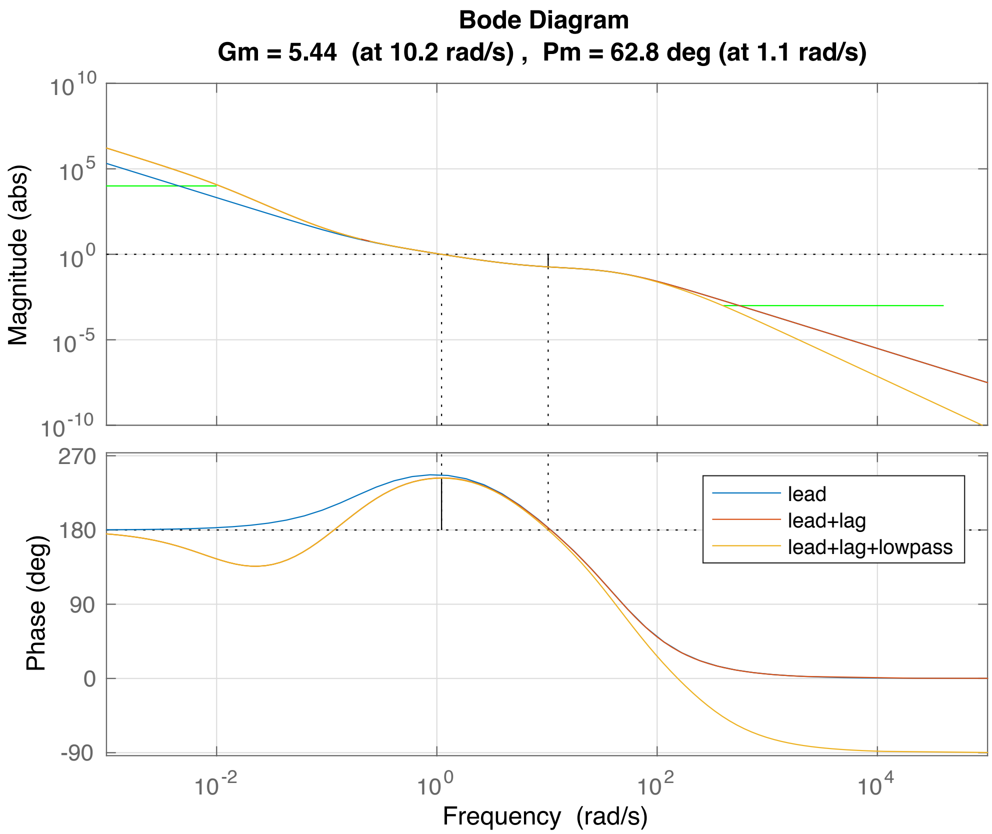

Figure 8: The Bode plot for the outer loop system in HW B.18, with lead, lag, and low-pass compensation applied.

The resulting compensator is

$$C_{out}(s) = 0.50 \left(\frac{s + 0.194}{s + 6.24} \right) \left(\frac{s + 0.08}{s + 0.01} \right) \left(\frac{230}{s + 230} \right).$$

The closed-loop response for both the inner and outer loop systems, as well as the unit-step response for the output and control signal of the outer loops are all show in Figure 9.

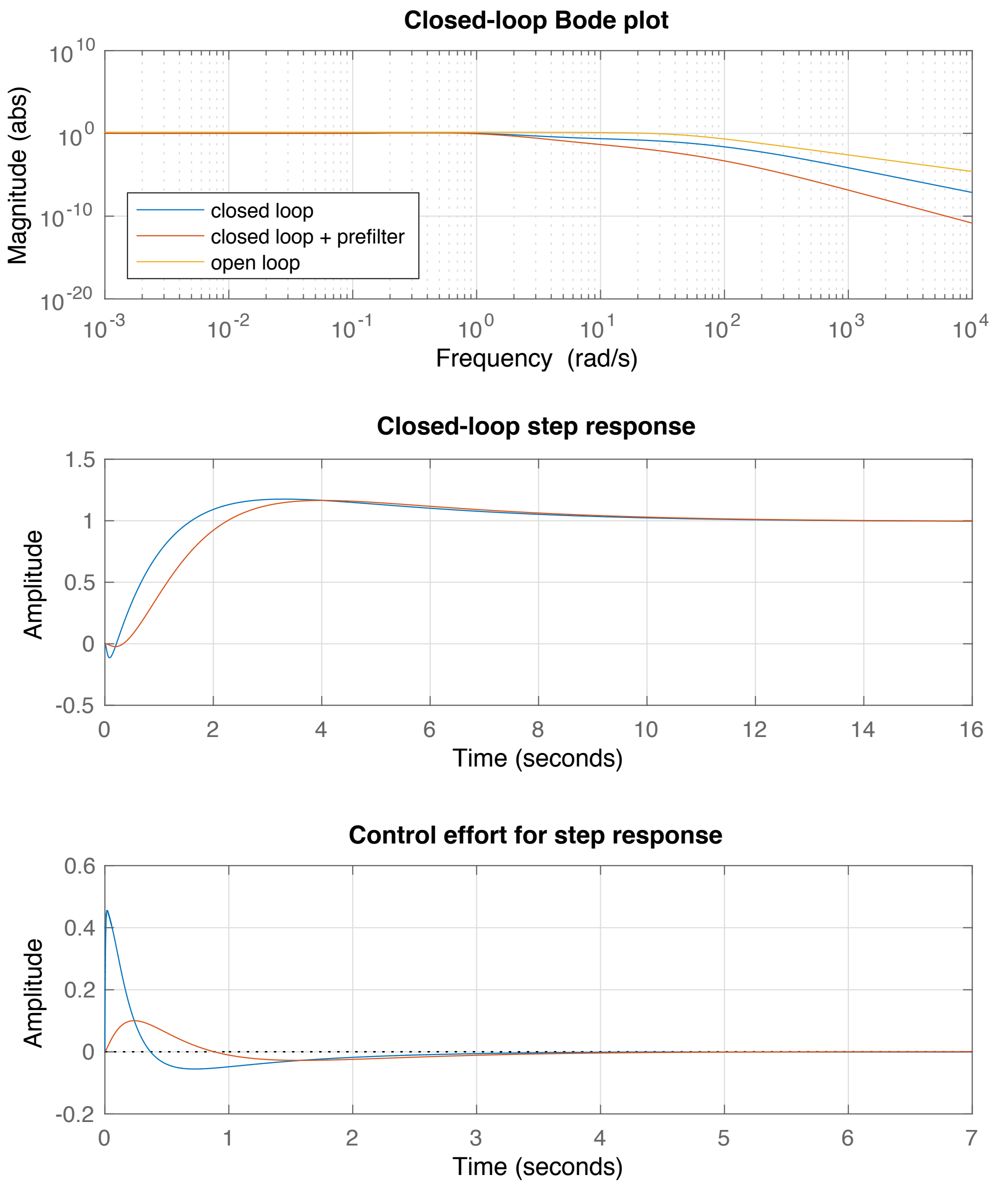

Figure 9: The closed-loop bode response, the unit-step response for the output, and the unit step response for the input of the inner loop design in HW B.18.

A prefilter

$$F(s) = \frac{2}{s+2}$$

has been added to the system. Note the significant decrease in the control effort required when prefiltering is used with the step input.

The Matlab code used to design the outer loop is shown below.

```
Plant = minreal(P_out*(P_in*C_in/(1+P_in*C_in)));
figure(2), clf
    hold on
    grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Define Design Specifications
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        %--- general tracking specification ---
        omega_r = 0.01;  % track signals below this frequency
        gamma_r = 0.0001;  % tracking error below this value
        w = logspace(log10(omega_r)-2,log10(omega_r));
        plot(w,(1/gamma_r)*ones(size(w)),'g')

        %--- noise specification ---
        omega_n = 400;  % attenuate noise above this frequency
        gamma_n = 0.001;   % attenuate noise by this amount
        w = logspace(log10(omega_n),2+log10(omega_n));
        plot(w,gamma_n*ones(size(w)),'g')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Control Design
  C = tf([1],[1]);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% phase lead: increase PM (stability)
% At desired crossover frequency, PM = -3
% Add 70 deg of PM with lead
    w_max = 1.1; % location of maximum frequency bump (desired crossover)
    phi_max = 70*pi/180;
    M = (1+sin(phi_max))/(1-sin(phi_max)) % lead ratio
    z = w_max/sqrt(M)
    p = w_max*sqrt(M)
    Lead = tf([1/z 1],[1/p 1]);
    C = C*Lead;

% find gain to set crossover at w_max = 1.1 rad/s
    [m,p] = bode(Plant*C,1.1);
    K = 1/m;
```

```
    C = K*C;

% Tracking constraint not satisfied -- add lag compensation to boost low-frequenc

% Find gain increase needed at omega_r
    [m,p] = bode(Plant*C,omega_r);
    gain_increase_needed = 1/gamma_r/m
    % Minimum gain increase at low frequencies is 4.8. Let lag ratio be 8.
    M = 8;
    p = omega_r;     % Set pole at omega_r
    z = M*p;         % Set zero at M*omega_r
    Lag = tf(M*[1/z 1],[1/p 1]);
    C = C*Lag;

% Noise attenuation constraint not quite satisfied
% Can be satisfied by reducing gain at 400 rad/s by a factor of 2
% Use a low-pass filter
        m = 0.5;     % attenuation factor
        a = m*omega_n*sqrt(1/(1-m^2));
        lpf = tf(a,[1 a]);
        C = lpf*C;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Prefilter Design
    F = tf([1],[1]);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% low pass filter
    p = 2;  % frequency to start the LPF
    LPF = tf(p,[1 p]);
    F = F*LPF

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Convert controller to state space equations for implementation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    C=minreal(C);
    [num,den] = tfdata(C,'v');
    [P.Aout_C,P.Bout_C,P.Cout_C,P.Dout_C]=tf2ss(num,den);

    [num,den] = tfdata(F,'v');
    [P.Aout_F, P.Bout_F, P.Cout_F, P.Dout_F] = tf2ss(num,den);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Convert controller to discrete transfer functions for implementation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
    C_out_d = c2d(C,P.Ts,'tustin')
    [P.Cout_d_num,P.Cout_d_den] = tfdata(C_out_d,'v');

    F_d = c2d(F,P.Ts,'tustin');
    [P.F_d_num,P.F_d_den] = tfdata(F_d,'v');

    C_out = C;
```

See http://controlbook.byu.edu for the complete solution.