A Design Study Approach to Classical Control

Randal W. Beard Timothy W. McLain
Brigham Young University

Updated: July 11, 2016

Homework A.18

For this homework assignment we will use loopshaping to improve the PID controllers developed in HW A.10. Let $C_{pid}(s)$ be the PID controller designed in HW A.10. The final control will be $C(s) = C_{pid}(s)C_l(s)$ where C_l is designed using loopshaping techniques.

(a) Design $C_l(s)$ to meet the following objectives:

  (1) Improve tracking and disturbance rejection by a factor of 10 for reference signals and disturbances below 0.07 rad/sec,

  (2) Improve noise attention by a factor of 10 for frequencies above 1000 radians/sec.

  (3) Phase margin that is approximately $PM = 60$ degrees.

(b) Add zero mean Gaussian noise with standard deviation $\sigma^2 = 0.01$ to the Simulink diagram developed in HW A.10.

(c) Implement the controller $C(s)$ in Simulink using its state space equivalent.

(d) Note that despite having a good phase margin, there is still significant overshoot, due in part to the windup effect in the phase lag filter. This can be mitigated by adding a prefilter, that essentially modifies the hard step input into the system. Add a low pass filter for $F(s)$ as a prefilter to flatten out the closed loop Bode response and implement in Simulink using its discrete time state space equivalent.

Solution

The first two design specifications can be displayed on the Bode magnitude plot as shown in Figure 1. To improve the tracking and disturbance rejection by a factor of 10 for reference signals and disturbances below $\omega_r = 0.07$ rad/sec, the loop gain must be 20 dB above $P(s)C_{PID}(s)$, as shown by the green line in the top left of Figure 1. To improve noise attenuation by a factor of 10 for frequencies above $\omega_n = 1000$ rad/sec, the loop gain must be 20 dB below $P(s)C_{PID}(s)$, as shown by the green line in the bottom right of Figure 1.

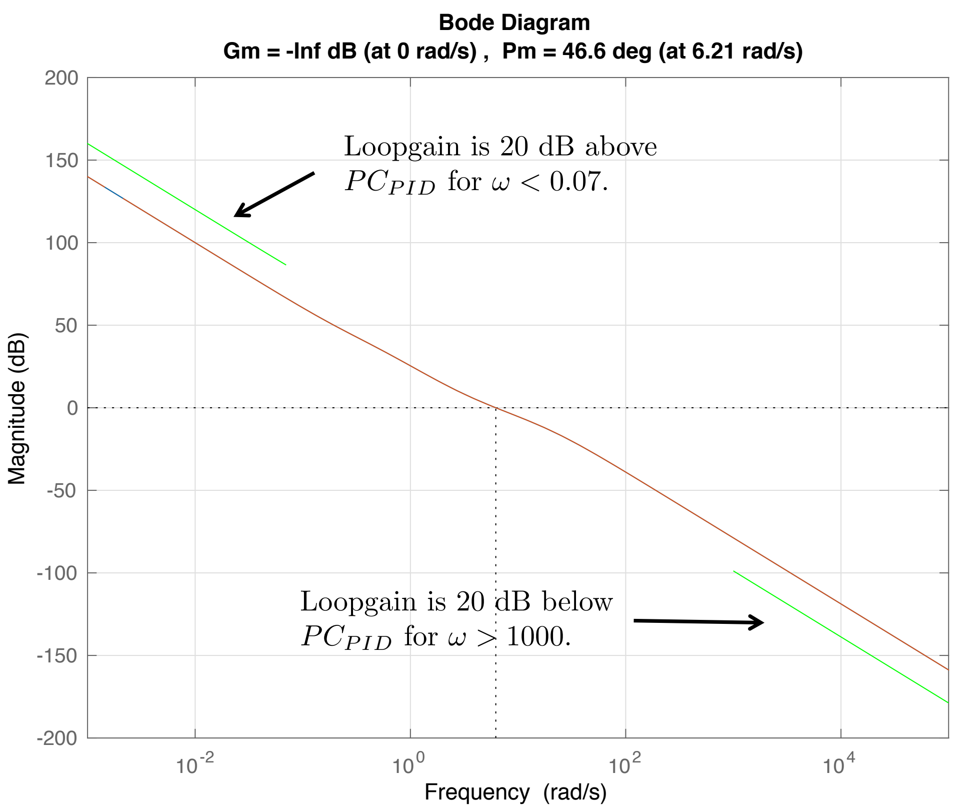

Figure 1: The Bode plot for the open loop plant in HW A.18, together with the design specifications.

To increase the loop gain below $\omega_r = 0.07$ rad/sec, a phase lag filter can be added with zero at $z = 0.7$ with a separation of $M = 10$. Figure 2 shows the loopgain of PC when

$$C(s) = C_{PID}(s)C_{Lag}(s) = \left(\frac{0.1036s^2 + 0.3108s + 0.1}{0.05s^2 + s}\right)\left(\frac{s + 0.7}{s + 0.07}\right).$$

From Figure 2 we see that the closed loop system will satisfy the tracking and input disturbance specifications.

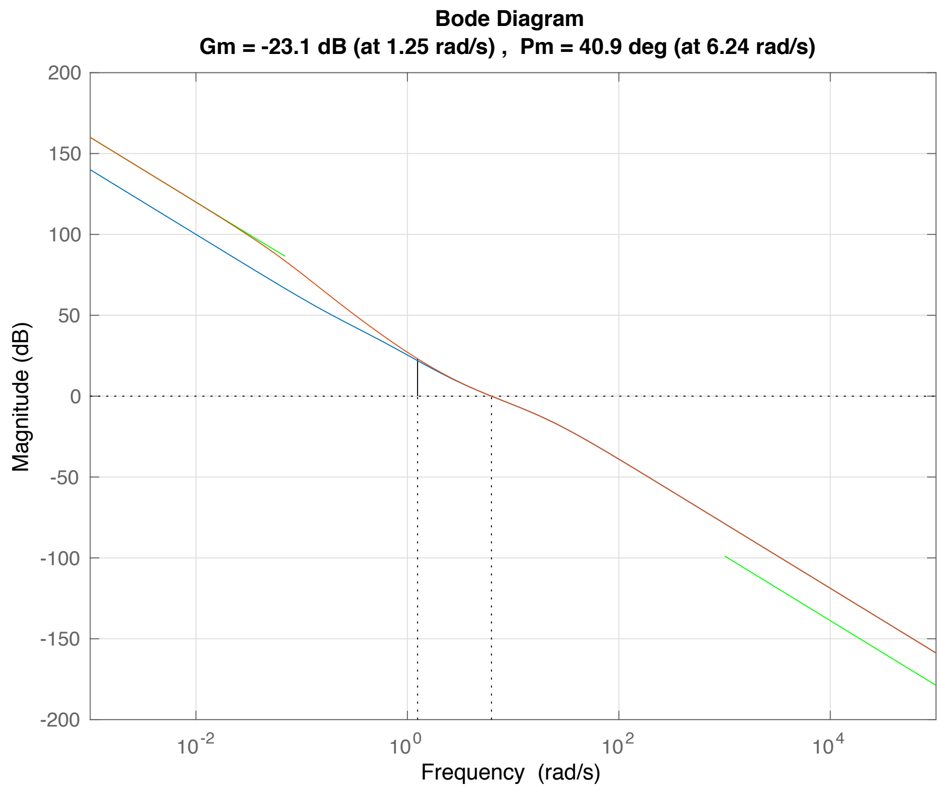

Figure 2: The Bode plot for HW A.18 where $C = C_{PID}C_{lag}$.

The phase margin after adding the lag filter is $PM = 40.9$ degrees. To increase the phase margin, a phase lead filter is added with center frequency $\omega_c = 30$ rad/sec and a separation of $M = 10$. The center frequency for the lead filter is selected to be above the cross over frequency so as not to change the location of the cross over frequency, thereby keeping the closed loop bandwidth, and therefore the control effort, roughly the same as with the PID controller. After adding the lead filter, the controller is

$$
\begin{aligned}
C(s) &= C_{PID}(s)C_{Lag}(s)C_{Lead}(s) \\
&= \left(\frac{0.1036s^2 + 0.3108s + 0.1}{0.05s^2 + s}\right)\left(\frac{s+0.7}{s+0.07}\right)\left(\frac{10s+94.87}{s+94.87}\right),
\end{aligned}
$$

and the corresponding loop gain is shown in Figure 3.

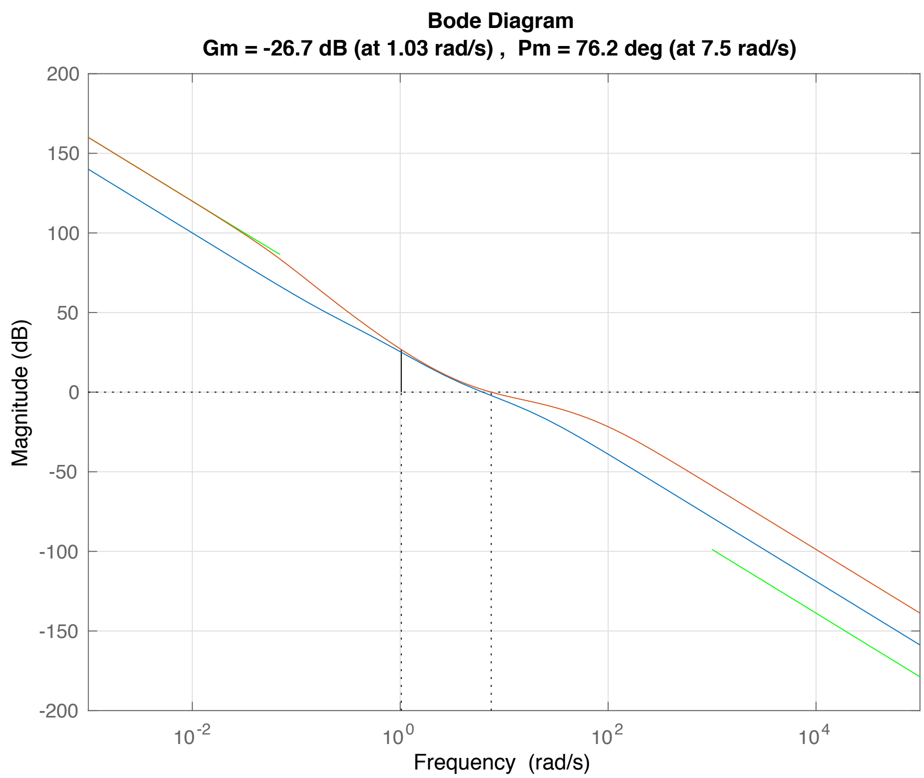

Figure 3: The Bode plot for HW A.18 where $C = C_{PID}C_{lag}C_{lead}$.

In order to satisfy the noise attenuation specification, we start by adding a low pass filter at $p = 50$. The corresponding loop gain is shown in Figure 4, from which we see that the noise specification is not yet satisfied. Therefore, we add an additional low pass filter at $p = 150$ to obtain the loopgain in Figure 5. The phase margin is $PM = 64$ degrees. The corresponding controller is

$$
\begin{aligned}
C(s) &= C_{PID}(s)C_{Lag}(s)C_{Lead}(s) \\
&= \left(\frac{0.1036s^2 + 0.3108s + 0.1}{0.05s^2 + s}\right)\left(\frac{s+0.7}{s+0.07}\right)\left(\frac{10s+94.87}{s+94.87}\right)\left(\frac{50}{s+50}\right)\left(\frac{150}{s+150}\right).
\end{aligned}
$$

Note that we have selected the filter values to leave the cross over frequency the same as with the original PID controller.

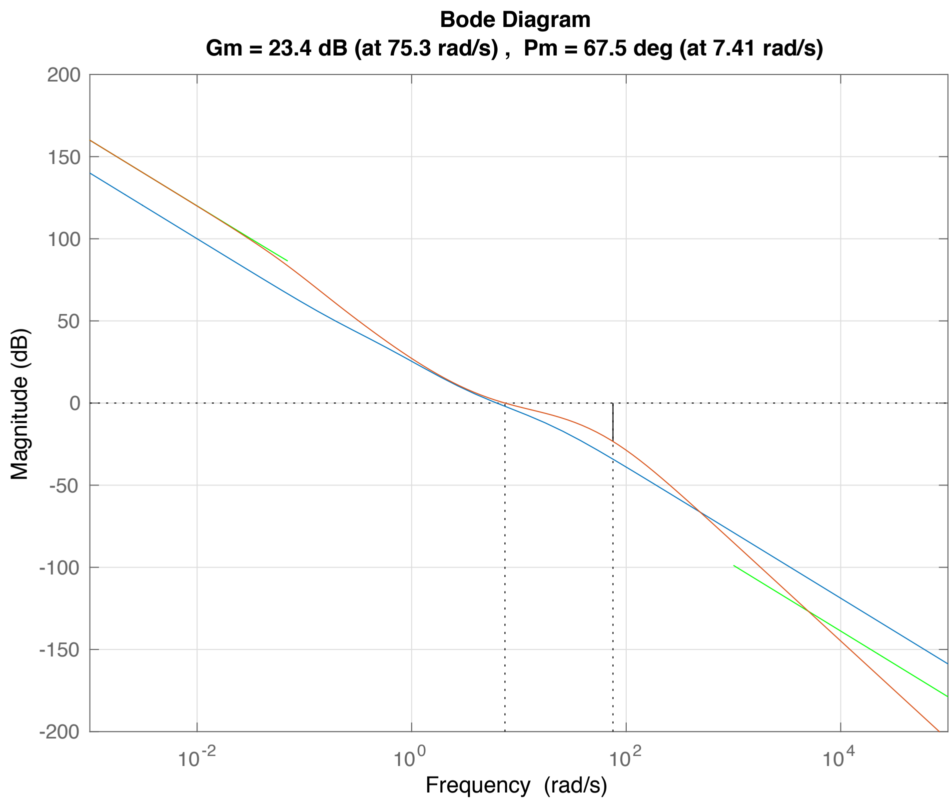

Figure 4: The Bode plot for HW A.18 where $C = C_{PID}C_{lag}C_{lead}C_{lpf}$.

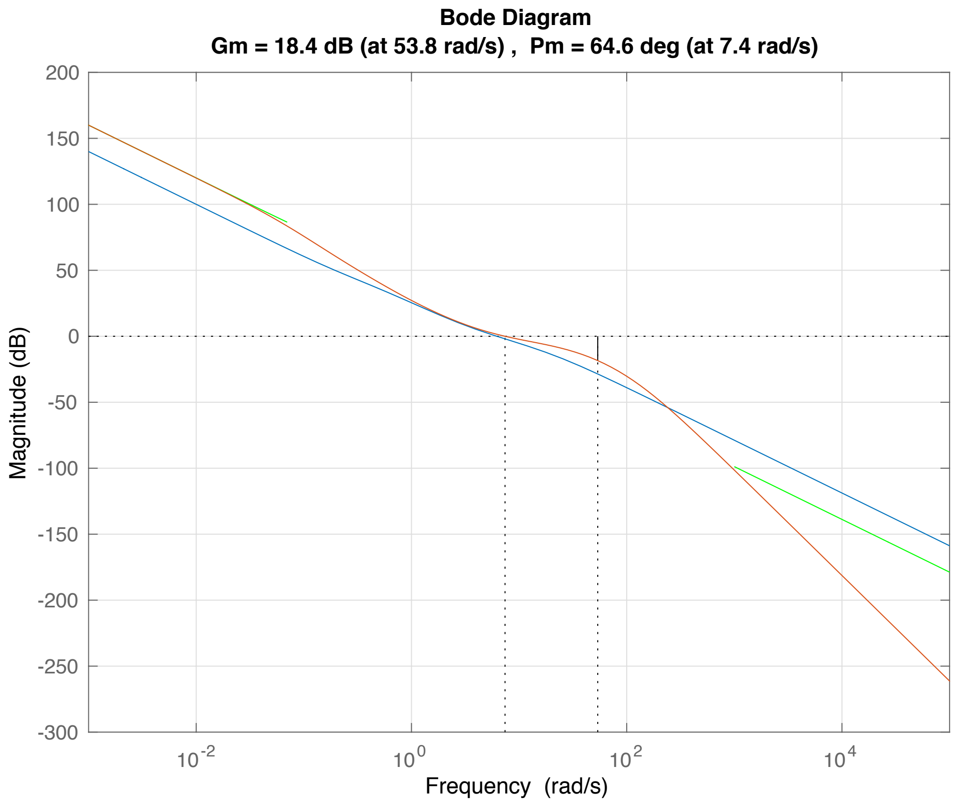

Figure 5: The Bode plot for HW A.18 where $C = C_{PID}C_{lag}C_{lead}C_{lpf}C_{lpf2}$.

The closed loop frequency response $PC/(1+PC)$ and the corresponding step response and control effort are shown by the blue lines in Figure 6. The overshoot in the step response is caused by the small amount of peaking on the closed loop bode plot. The peaking can be reduced by adding a prefilter, which in this case is a low pass filter with pole $p = 3$. The prefiltered closed loop frequency response $FPC/(1+PC)$ and the corresponding step response and control effort are shown by the red lines in Figure 6. As shown by Figure 6, the prefilter reduces the overshoot and lowers the control effort.

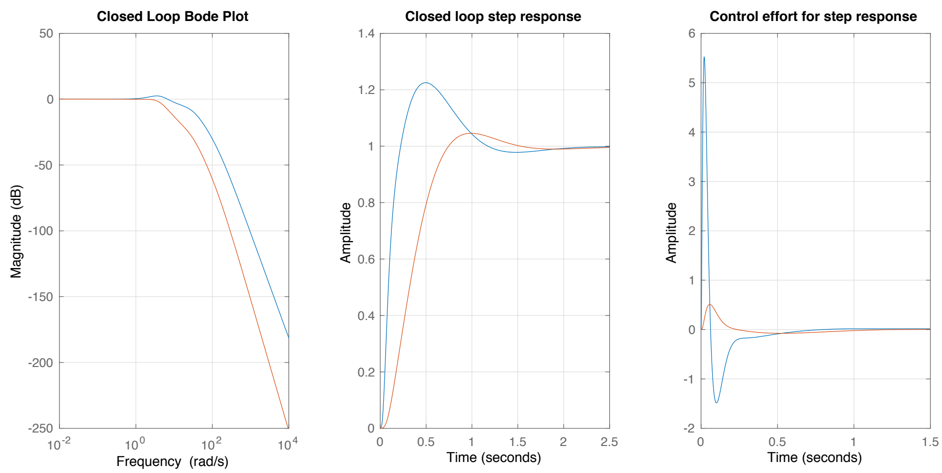

Figure 6: On the left is the closed loop frequency response in blue, and the prefiltered closed loop frequency response in red. In the middle are the associated closed-loop step response. On the right is the associated control effort.

Matlab code for the design of the controller is shown below.

```
param

% start with pid control designed in problem 10
 figure(1), clf
    bodemag(Plant*C_pid,logspace(-3,5))
    hold on
    grid on

% add constraints

% increase tracking by factor of 10 below omega_r=0.007
  omega_r = 0.07;  % reject input dist. below this frequency
  gamma_r = 0.1;    % amountn of input disturbance in output
  w = logspace(log10(omega_r)-2,log10(omega_r));
  Pmag=bode(Plant*C_pid,w);
  for i=1:size(Pmag,3), Pmag_(i)=Pmag(1,1,i); end
  plot(w,20*log10(1/gamma_r)*ones(1,length(Pmag_))...
         +20*log10(Pmag_),'g')

% attenuate noise by factor of gamma_n above omega_n
```

```
  omega_n = 1000;  % attenuate noise above this frequency
  gamma_n = 0.1;    % amount of noise attenuation in output
  w = logspace(log10(omega_n),log10(omega_n)+2);
  Pmag=bode(Plant*C_pid,w);
  for i=1:size(Pmag,3), Pmag_(i)=Pmag(1,1,i); end
  plot(w,20*log10(gamma_n)*ones(1,length(Pmag_))...
         +20*log10(Pmag_),'g')
  %figure(1), margin(Plant*C_pid)
  %pause % hw_arm_compensator_design_1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Control Design
     C = C_pid;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% phase lag (|p|<|z|): add gain at low frequency
%                       (tracking, dist rejection)
     % low frequency gain = K*z/p
     % high frequency gain = K
     z = .7;
     p = z/10;
     Lag = tf([1,z],[1,p]);
     C = C*Lag;
     %figure(1), margin(Plant*C)
     %pause % hw_arm_compensator_design_2

% phase lead (|p|>|z|): increase PM (stability)
     % low frequency gain = K*z/p
     % high frequency gain = K
     wmax = 30; % location of maximum frequency bump
     M   = 10; % separation between zero and pole
     Lead =tf(M*[1,wmax/sqrt(M)],[1,wmax*sqrt(M)]);
     C = C*Lead;
     %figure(1), margin(Plant*C)
     %pause % hw_arm_compensator_design_3

% low pass filter: decrease gain at high frequency (noise)
     p = 50;
     LPF = tf(p,[1,p]);
     C = C*LPF;
     %figure(1), margin(Plant*C)
     %pause % hw_arm_compensator_design_4

% low pass filter: decrease gain at high frequency (noise)
     p = 150;
```

```
     LPF = tf(p,[1,p]);
     C = C*LPF;
     %figure(1), margin(Plant*C)
     %pause % hw_arm_compensator_design_5

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% add a prefilter to eliminate the overshoot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
F = 1;
% low pass filter
     p = 3;
     LPF = tf(p,[1,p]);
     F = F*LPF;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Create plots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Open-loop tranfer function
OPEN = Plant*C;
% closed loop transfer function from R to Y
CLOSED_R_to_Y = (Plant*C/(1+Plant*C));
% closed loop transfer function from R to U
CLOSED_R_to_U = (C/(1+C*Plant));

figure(2), clf
    subplot(1,3,1),
        bodemag(CLOSED_R_to_Y), hold on
        bodemag(CLOSED_R_to_Y*F)
        title('Closed Loop Bode Plot'), grid on
    subplot(1,3,2),
        step(CLOSED_R_to_Y), hold on
        step(CLOSED_R_to_Y*F)
        title('Closed loop step response'), grid on
    subplot(1,3,3),
        step(CLOSED_R_to_U), hold on
        step(CLOSED_R_to_U*F)
        title('Control effort for step response'), grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Convert controller to state space equations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[num,den] = tfdata(C,'v');
[P.A_C,P.B_C,P.C_C,P.D_C]=tf2ss(num,den);

[num,den] = tfdata(F,'v');
```

```
[P.A_F, P.B_F, P.C_F, P.D_F] = tf2ss(num,den);
```