36 Control of a High-Speed Vehicle

An instability in certain aircraft and in some high-speed marine vehicles is characterized by a pair of complex right-half plane poles, and is due to inadequate aerodynamic stability. It is particularly pronounced in highly maneuverable craft, where open-loop stability in the physical design has been intentionally traded off against agility. In this problem, you will use the Nyquist criterion to bring such a vehicle under control and to achieve specific performance and robustness properties.

1. The transfer function taking the elevator control surface command into the pitch of the vehicle is given as

$$P(s) = \frac{y(s)}{u(s)} = \frac{2(s+2)}{s^2 - 0.2s + 16}$$

   Confirm that this is an unstable system by both stating the poles and by plotting the impulse response.

2. Plot the complex loci for this plant. This means plotting the real versus the imaginary parts of $P(s)$, $s = j\omega$, for ω going from zero to infinity. Then plot this same curve again, negating the imaginary part, and you then have covered the range of ω from $[-\infty, \infty]$. Use logarithmically-spaced frequencies.

3. Applying the Nyquist criterion to this plot, can the vehicle be stabilized with unity feedback gain, i.e., $C(s) = 1$? If so, comment on whether it is a useful feedback system.

4. We need to achieve through feedback a) integral action, b) a phase margin of 30-40 degrees, and c) a gain margin of at least two. To do this, design a controller $C(s)$ with the following properties:

   - a real pole just to the left of the origin, corresponding with an integrator (we want it to be definitely *not* in the RHP);
   - a stable complex zero pair that approximately mirrors the unstable plant poles across the imaginary axis, but possibly with a higher damping ratio;
   - two additional stable, real poles so that the compensator has a high-frequency roll-off behavior.

   State the complete transfer function of your designed compensator, show that it achieves the desired gain and phase margins on a plot of $P(s)C(s)$ loci, and show a closed-loop step response to prove that it is indeed well-behaved.

1. *The plant poles are at $s = 0.1 \pm 4j$. Since they are in the right-half plane, the system is unstable. The open-loop response shows an oscillation of growing amplitude.*

2. *See the figures with loci of $P(s)$ and magnitude $|P(s)|$.*

3. *$|P(s)|$ goes to zero at high frequencies, and takes a small but definitely nonzero value for low frequencies. Thus, sweeping s from negative to positive infinity and around the RHP leads to two counterclockwise encirclements of the critical point $1 + 0j$. This is matched by the two unstable poles in the plant, i.e., we do have $P = CCW$, and the system is stable with unity feedback!*

   It is not a very good design, however, for at least one reason: the magnitude of $P(s)C(s)$ is small at low frequencies, and hence the system will track a reference signal very poorly. A simulation quickly verifies this fact.

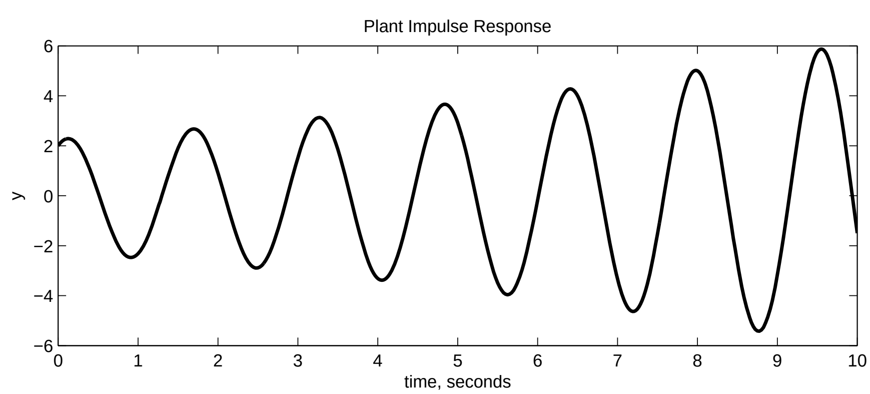
4. *A feasible compensator is:*

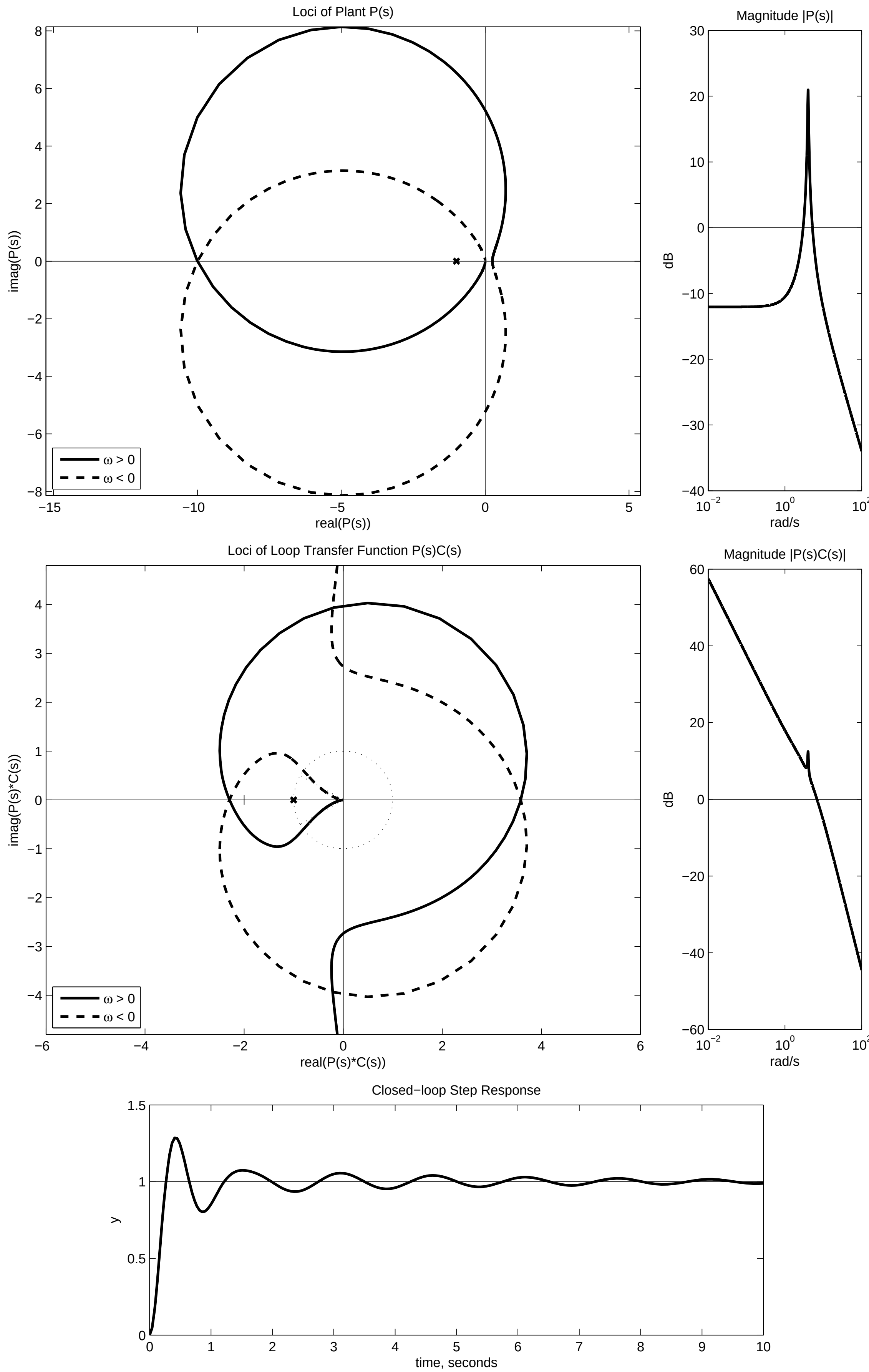
$$C(s) = \frac{u(s)}{e(s)} = \frac{30(s^2 + 0.4s + 16.04)}{(s + 0.0001)(s + 4)^2}.$$

   As suggested in the problem statement, this solution has a pole very close to the origin but definitely stable, two stable zeros in a mirror configuration plus a little more damping, and two real poles. The mirror configuration zeros provide a clever way to deal with nasty unstable plant poles; the pair roughly cancels the $|P(s)|$ peak. This compensator has quite high gain below about four radians per second, but rolls off to zero at high frequencies.

   Looking at the loci of the loop transfer function $|P(s)C(s)|$, we see immediately two CCW encirclements of the critical point, with finite magnitudes. The transition through zero frequency (from small negative to small positive), however, occurs at very large magnitude of $P(s)C(s)$ and is approached at ± 90 degrees. We have to ask which way do these lines connect, in the right- or left-half plane? Since the path of s encloses the entire RHP, this path must include a positive, real value - putting a positive real value for s into the loop transfer function shows that the completing portion of the $P(s)C(s)$ loci occurs in the RHP. We are then able to conclude that this loop does not encircle the critical point. The remaining two CCW encirclements match the unstable poles of the plant, $P = CCW$ as before, and the system is stable.

   The figures show that this design satisfies the gain and phase margin requirements, and that the closed-loop response is acceptable in terms of steady-state error and ringing.




Loci of Plant P(s)
imag(P(s))
real(P(s))
ω > 0
ω < 0
Magnitude |P(s)|
dB
rad/s
Loci of Loop Transfer Function P(s)C(s)
imag(P(s)*C(s))
real(P(s)*C(s))
ω > 0
ω < 0
Magnitude |P(s)C(s)|
dB
rad/s
Closed−loop Step Response
y
time, seconds

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Design of a flight controller using Nyquist plot.

% FSH MIT Mechanical Engineering April 2008

clear all;

zP = [-2] ; % zeros in plant
pP = [0.1+4*sqrt(-1), 0.1-4*sqrt(-1)] ; % poles in plant
kP = 2 ; % gain of plant

zC = [-0.2+4*sqrt(-1), -0.2-4*sqrt(-1)] ; % zeros in controller
pC = [-4 -4 -0.0001] ; % poles in controller
kC = 30 ; % gain of controller

w = logspace(-2,2,3000); % (a very dense) frequency vector

% compute and plot the plant impulse response
sysP = zpk(zP,pP,kP);
[y,t] = impulse(sysP,10);
        % (could also solve the ode explicitly, with a tall
        % rectangle or triangle approximation for the impulse)
figure(1);clf;hold off;
subplot('Position',[.1 .2 .8 .4]);
plot(t,y,'k','LineWidth',2);
xlabel('time, seconds');
ylabel('y');
title('Plant Impulse Response');
print -deps flightControl1.eps ;

% make up the loci of P(s)
% (could also do this in one line with freqresp() !)
for i = 1:length(w),
    P(i) = kP ;
    s = sqrt(-1)*w(i) ;
    for j = 1:length(zP),
        P(i) = P(i) * (s - zP(j)) ;
    end;
    for j = 1:length(pP),
        P(i) = P(i) / (s - pP(j)) ;
    end;
end;

% plot the loci of P(s)
```

```
figure(2);clf;hold off;
plot(real(P),imag(P),'k',real(P),-imag(P),'k--','LineWidth',2);
axis('equal');
a = axis ;
hold on;
title('Loci of Plant P(s)');
legend('\omega > 0','\omega < 0',3);
plot([0 0],[a(3) a(4)],'k',[a(1) a(2)],[0 0],'k');
plot(-1,0,'kx','LineWidth',2);
xlabel('real(P(s))');
ylabel('imag(P(s))');
print -deps flightControl2.eps ;

% make up the loci of C(s)
for i = 1:length(w),
    C(i) = kC ;
    s = sqrt(-1)*w(i) ;
    for j = 1:length(zC),
        C(i) = C(i) * (s - zC(j)) ;
    end;
    for j = 1:length(pC),
        C(i) = C(i) / (s - pC(j)) ;
    end;
end;

% plot the loci of P(s)*C(s)
figure(3); clf;hold off;
plot(real(P.*C),imag(P.*C),'k-',real(P.*C),-imag(P.*C),'k--',...
    -1,0,'bx','LineWidth',2);
hold on;
legend('\omega > 0','\omega < 0',3);
axis([-5 5 -4 4]*1.2);
a = axis ;
plot(sin(.0:.01:2*pi),cos(0:.01:2*pi),'k:');
plot([-2,-2],[-.1 .1],'k-');
plot([0 0],[a(3) a(4)],'k',[a(1) a(2)],[0 0],'k');
plot([0 -sqrt(3)/2]*1.,[0 1/2]*1.,'k:',...
        [0 -sqrt(3)/2]*1.,[0 -1/2]*1.,'k:');
xlabel('real(P(s)*C(s))');
ylabel('imag(P(s)*C(s))');
title('Loci of Loop Transfer Function P(s)C(s)');
print -deps flightControl3.eps ;

% plot the magnitude of PC
```

```
figure(4);clf;hold off;
subplot('Position',[.2 .1 .2 .8]);
semilogx(w,20*log10(abs(P.*C)),'k','LineWidth',2);
hold on;
semilogx([min(w),max(w)],[0 0],'k-');
xlabel('rad/s');
ylabel('dB');
title('Magnitude |P(s)C(s)|');
print -deps flightControl4.eps ;

% compute and plot the closed-loop step response
sysC = zpk(zC,pC,kC);
[y,t] = step(feedback(sysP*sysC,1),10);
    % could also do an ode to solve this

figure(5);clf;hold off;
subplot('Position',[.1 .2 .8 .4]);
plot(t,y,'k','LineWidth',2);
hold on;
plot([0 10],[1 1],'k-');
xlabel('time, seconds');
ylabel('y');
title('Closed-loop Step Response');
a = axis ; axis([0 10 a(3:4)]);
print -deps flightControl5.eps ;

% plot the magnitude of P - just for good measure
figure(6);clf;hold off;
subplot('Position',[.2 .1 .2 .8]);
semilogx(w,20*log10(abs(P)),'k','LineWidth',2);
hold on;
semilogx([min(w),max(w)],[0 0],'k-');
xlabel('rad/s');
ylabel('dB');
title('Magnitude |P(s)|');
print -deps flightControl6.eps ;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```