N I N E

Design via Root Locus

SOLUTIONS TO CASE STUDIES CHALLENGES

**Antenna Control: Lag-Lead Compensation**

**a.** Uncompensated: From the Chapter 8 Case Study Challenge, G(s) = = with the dominant poles at - 0.5 ± j6.9. Hence,  = cos (tan-1 ) = 0.0723, or %OS = 79.63% and Ts = = = 8 seconds. Also, Kv = = 36.33.

**b.** Lead-Compensated: Reducing the percent overshoot by a factor of 4 yields, %OS = = 19.91%, or  = 0.457. Reducing the settling time by a factor of 2 yields, Ts = = 4. Improving Kv by 2 yields Kv = 72.66. Using Ts = = 4, n = 1, from which n = 2.188 rad/s. Thus, the design point equals -n + j n= -1 + j1.946. Using the system's original poles and assuming a lead compensator zero at -1.5, the summation of the system's poles and the lead compensator zero to the design point is -123.017o . Thus, the compensator pole must contribute 123.017o-180o = -56.98o. Using the geometry below, = tan 56.98o, or pc = 2.26.



Adding this pole to the system poles and the compensator zero yields 76.39K = 741.88 at -1+j1.946. Hence the lead-compensated open-loop transfer function is GLead-comp(s) =

. Searching the real axis segments of the root locus yields higher-order poles at greater than -150 and at -1.55. The response should be simulated since there may not be pole/zero cancellation. The lead-compensated step response is shown below.



Since the settling time and percent overshoot meet the transient requirements, proceed with the lag compensator. The lead-compensated system has Kv = = 2.487. Since we want Kv = 72.66, an improvement of = 29.22 is required. Select G(s)Lag = to improve the steady-state error by 29.22. A simulation of the lag-lead compensated system,

GLag-lead-comp(s) = is shown below.



**UFSS Vehicle: Lead and Feedback Compensation**

Minor loop: Open-loop transfer function G(s)H(s) = ; Closed-loop transfer

function: . Searching along the 126.87o line ( = 0.6), find the dominant second-order poles at -1.554 ± j2.072 with 0.25K2 = 4.7. Thus K2 = 18.8. Searching the real axis segment of the root locus for a gain of 4.7 yields a 3rd pole at -0.379.

Major loop: The unity feedback, open-loop transfer function found by using the minor-loop closed-loop poles is GML(s) = . Searching along the 120o line ( = 0.5), find the dominant second-order poles at -1.069±j1.85 with 0.25K1 = 4.55. Thus K1 = 18.2. Searching the real axis segment of the root locus for a gain of 4.55 yields a 3rd pole at -0.53 and a 4th pole at -0.815.

Answers to Review QUESTIONS

**1.** Chapter 8: Design via gain adjustment. Chapter 9: Design via cascaded or feedback filters

**2.** A. Permits design for transient responses not on original root locus and unattainable through simple gain adjustments. B. Transient response and steady-state error specifications can be met separately and independently without the need for tradeoffs

**3.** PI or lag compensation

**4.** PD or lead compensation

**5.** PID or lag-lead compensation

**6.** A pole is placed on or near the origin to increase or nearly increase the system type, and the zero is placed near the pole in order not to change the transient response.

**7.** The zero is placed closer to the imaginary axis than the pole. The total contribution of the pole and zero along with the previous poles and zeros must yield 1800 at the design point. Placing the zero closer to the imaginary axis tends to speed up a slow response.

**8.** A PD controller yields a single zero, while a lead network yields a zero and a pole. The zero is closer to the imaginary axis.

**9.** Further out along the same radial line drawn from the origin to the uncompensated poles

**10.** The PI controller places a pole right at the origin, thus increasing the system type and driving the error to zero. A lag network places the pole only close to the origin yielding improvement but no zero error.

**11.** The transient response is approximately the same as the uncompensated system, except after the original settling time has passed. A slow movement toward the new final value is noticed.

**12.** 25 times; the improvement equals the ratio of the zero location to the pole location.

**13.** No; the feedback compensator's zero is not a zero of the closed-loop system.

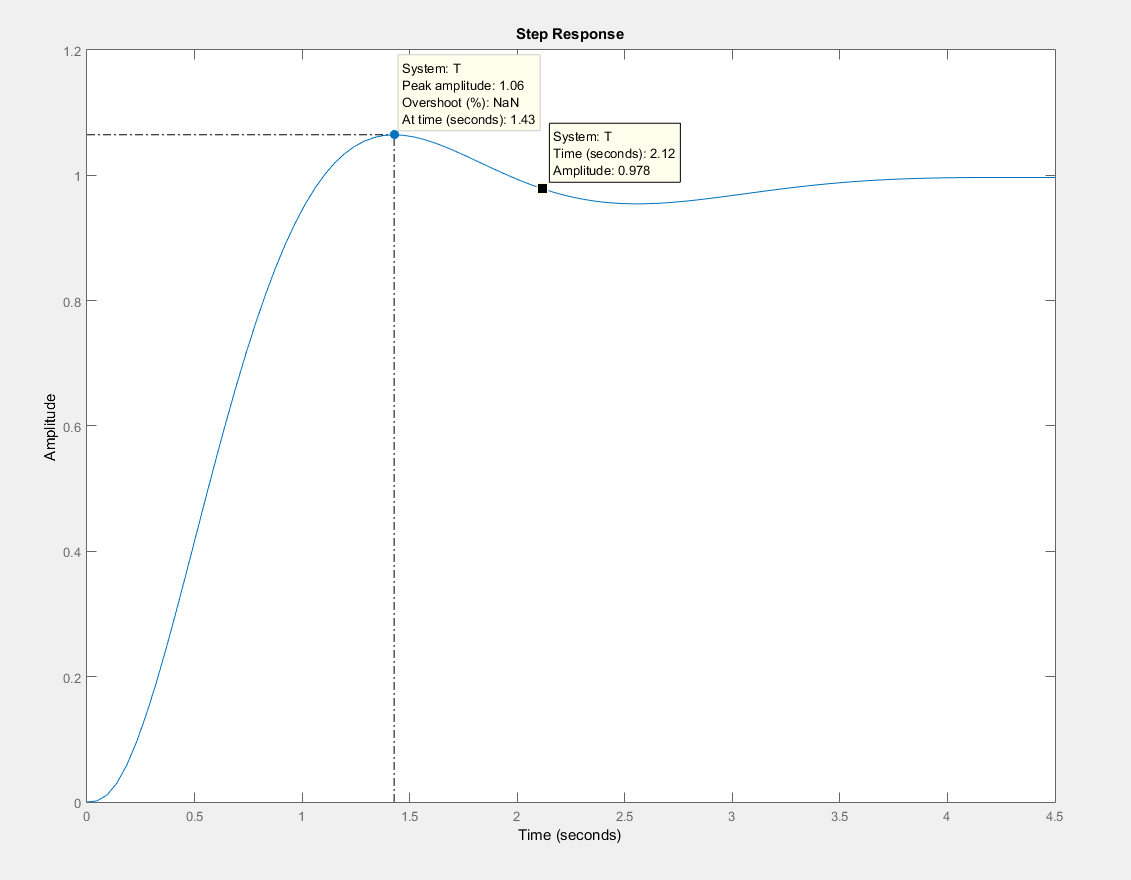
**14.** A. Response of inner loops can be separately designed; B. Faster responses possible; C. Amplification may not be necessary since signal goes from high amplitude to low.

SOLUTIONS To Problems

**1.**

Uncompensated: Search along the line and find the operating point with . A damping factor of corresponds to a %OS=16.3%. sec. . There is a higher order pole at .

Compensated: The PI controller is chosen as Search along the line and find the operating point with . sec.  . The two higher order poles are at -0.84 and -20.4. The system must be simulated to verify performance:



**2.**

**a.** Insert a cascade compensator, such as .

**b.**

**Program:**

K=1

G1=zpk([],[0,-3,-6],K) %G1=1/s(s+3)(s+6)

Gc=zpk([-0.01],[0],1) %Gc=(s+0.01)/s

G=G1\*Gc

rlocus(G)

T=feedback(G,1)

T1=tf(1,[1,0]) %Form 1/s to integrate step input

T2=T\*T1

t=0:0.1:200;

step(T1,T2,t) %Show input ramp and ramp response

**Computer response:**

K =

1

Zero/pole/gain:

1

-------------

s (s+3) (s+6)

Zero/pole/gain:

(s+0.01)

--------

s

Zero/pole/gain:

(s+0.01)

---------------

s^2 (s+3) (s+6)

Zero/pole/gain:

(s+0.01)

-------------------------------------------

(s+6.054) (s+2.889) (s+0.04384) (s+0.01304)

Transfer function:

1

-

s

Zero/pole/gain:

(s+0.01)

---------------------------------------------

s (s+6.054) (s+2.889) (s+0.04384) (s+0.01304)



**3.**

**a.** Searching along the 148.9o line (15% overshoot,  = 0.517), find the operating point at

-1.08 + j1.79 with K = 15.52. Hence, 

**b.** A 6.4433 x improvement will yield Kp = 1. Use a lag compensator, .

**c.**





**4. I-1 Instructor**

**a.** Searching along the 126.16o line (10% overshoot, ** = 0.59), find the operating point at

-1.1207 + j1.5336 with *K* = 27.9948. Hence, *Kv* = .

**b.** A 3.0006 x improvement will yield *Kv* = 4. Use a lag compensator, *Gc(s)* = .

**c.**

**Program:**

K=17.5

G=zpk([],[0,-3,-5],K)

Gc=zpk([-0.3429],[-0.1],1)

Ge=G\*Gc;

T1=feedback(G,1);

T2=feedback(Ge,1);

T3=tf(1,[1,0]); %Form 1/s to integrate step input

T4=T1\*T3;

T5=T2\*T3;

t=0:0.1:20;

step(T3,T4,T5,t) %Show input ramp and ramp responses

**Computer response:**

K =

27.9948

Zero/pole/gain:

27.9948

-------------

s (s+3) (s+7)

Zero/pole/gain:

(s+0.3001)

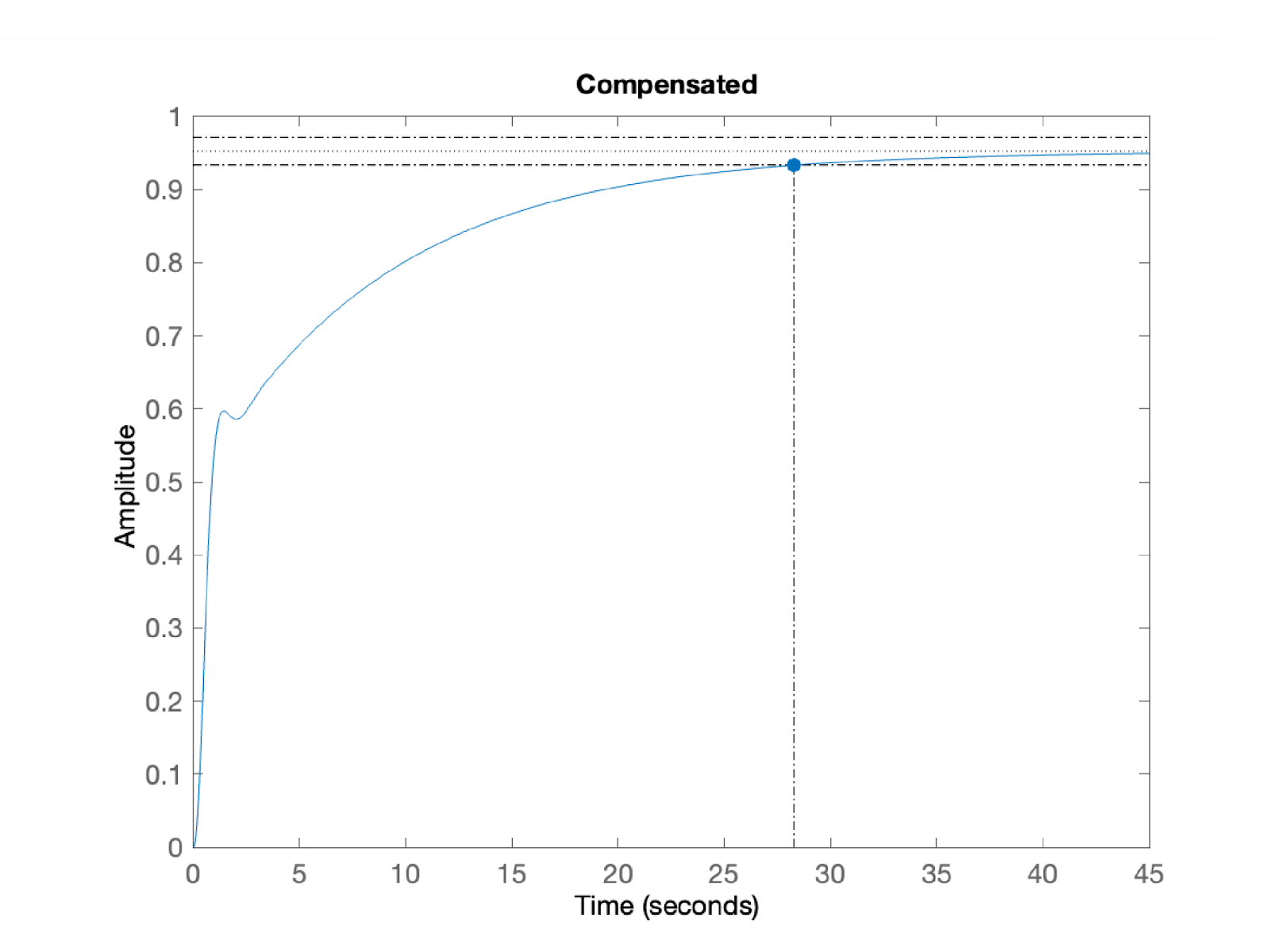
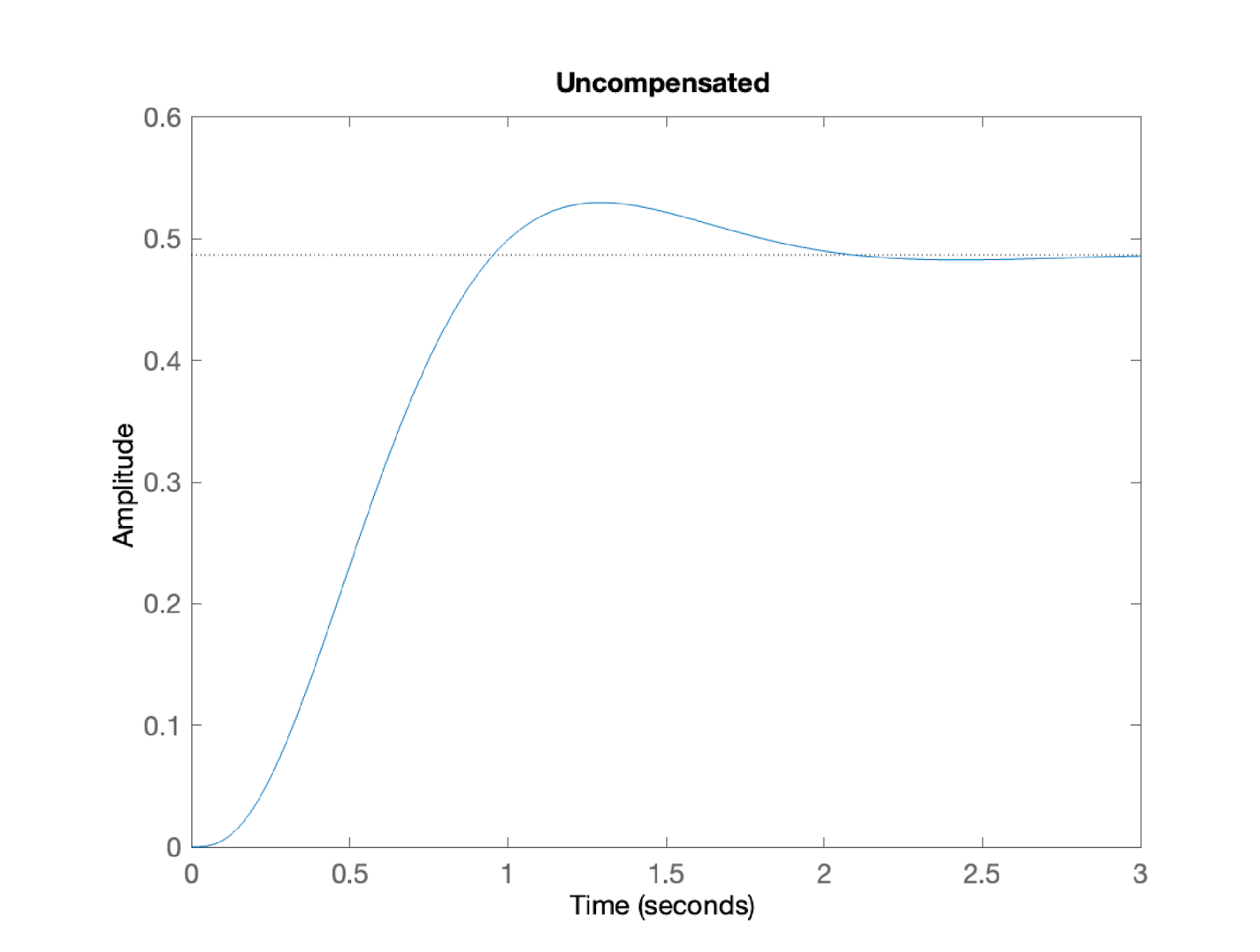
----------

(s+0.1)

**4**

**a.** Uncompensated: Searching along the 126.16o line (10% overshoot,  = 0.59), find the operating point at -2.03 + j2.77 with K = 45.537. Hence, . An improvement of  is required. Let . Compensated: Searching along the 126.16o line (10% overshoot,  = 0.59), find the operating point at – 1.99+j2.7 with   
*K* = 45.518. Hence, .

**b. *(****Uncompensated and Compensated graphs below were corrected & updated in solutions manual in August 2022)*



**c.** From (b), about \*\*28 seconds.

**\*\***Answer to **c.**corrected from “about 272 seconds” to “28 seconds” - August 2022

**54. •Sp 9-7 (Checked)(PD compensation)) (Chap 9)**

**4. The unity feedback system shown in Figure P.9.1 with G(s) = is operating with a dominant pole damping ratio = 0.707. Design a PD controller so that the settling time is reduced by two. Compare the transient and steady-state performance of the uncompensated and compensated systems. Describe any problems with your design.**

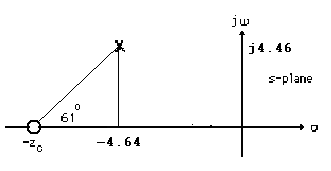
**.**

Uncompensated: Searching along the 135o line ( = 0.707), find the operating point at

-2.32 + j2.32 with K = 4.6045. Hence, Kp = = 0.153; Ts = = 1.724 seconds; Tp = = 1.354 seconds;  = 4.33%;

n = = 3.28 rad/s; higher-order pole at -5.366.

Compensated: To reduce the settling time by a factor of 2, the closed-loop poles should be – 4.64 ± j4.64. The summation of angles to this point is 119o . Hence, the contribution of the compensating zero should be 180o -119o =61o . Using the geometry shown below,   
= tan (61o). Or, zc = 7.21.



After adding the compensator zero, the gain at -4.64+j4.64 is K = 4.77. Hence, .  second;  second;   
 = 4.33%; rad/s; higher-order pole at

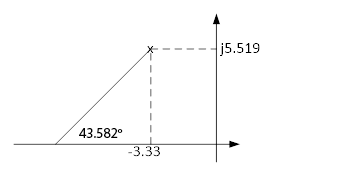
-5.49. The problem with the design is that there is steady-state error, and no effective pole/zero cancellation. The design should be simulated to be sure the transient requirements are met.

**6.**

**a**.  ,  . Thus rad/sec and the testing point is  .

**b**. Summation of angles including the compensating zero is -136.419°, The compensator pole must contribute 136.419°-180°=-43.582°

**c**. Using the geometry shown below . Thus .

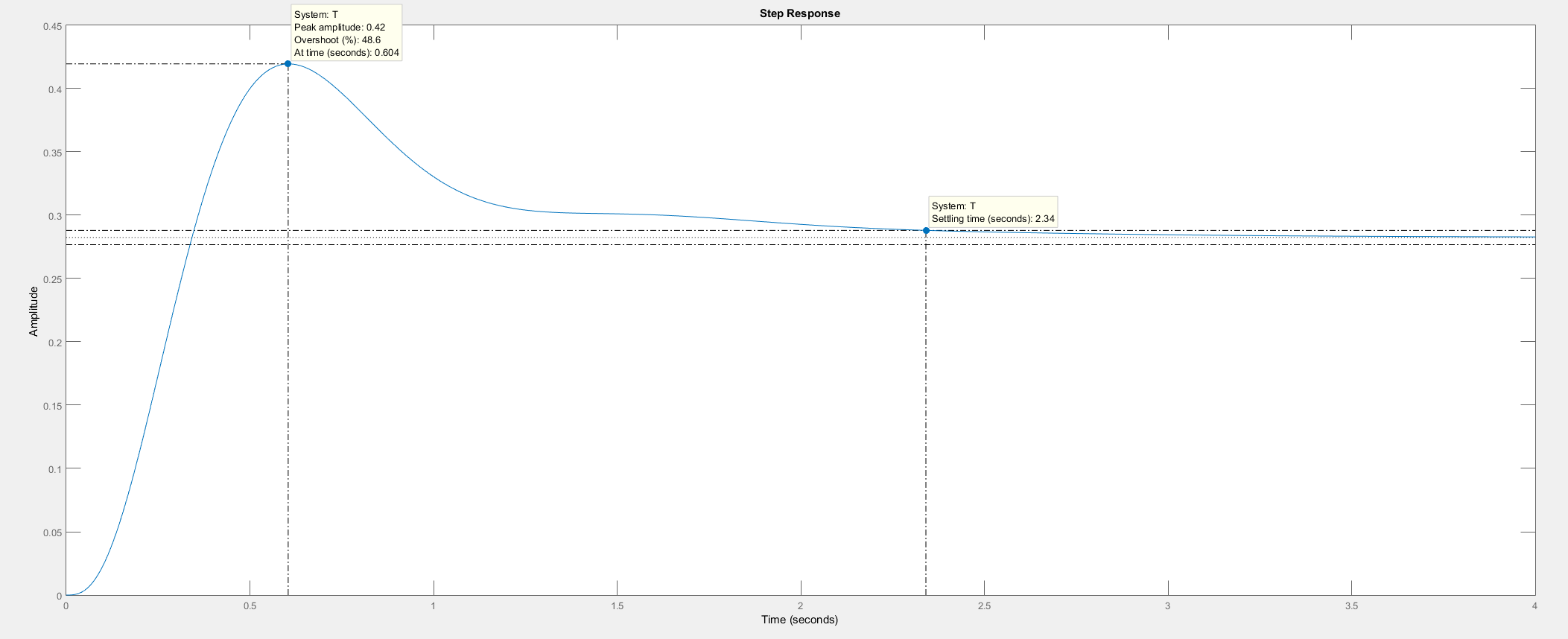


**d**. Adding the compensator pole test for -3.33+j5.519 which gives K=230.

**e**. Searching the real axis segments for K=230 we find higher order poles at -1.52 and -12.9

**f.** The pole at -12.9 is far away to the left. The pole at -1.52step(T)\_ may not cancel the zero at -1. Questionable second order approximation the system should be simulated.

**g.**



A simulation of the system shows a percent overshoot of 48.6% and a settling time of 2.34 sec. This the specifications were not met because pole-zero cancellation was not achieved. A redesign is required.

**7.**

**a.**  n = = 2.4;  = = 0.5. Thus, n = 4.799 rad/s and the operating point is -2.4 ± j4.16.**b.** Summation of angles including the compensating zero is -131.36o. Therefore, the compensator pole must contribute 180o - 131.36o = -48.64o. Using the geometry shown below, =

tan 48.64o. Thus, pc = 6.06.



**c.** Adding the compensator pole and using -2.4 + j4.16 as the test point, K = 29.117.

**d.** Searching the real axis segments for K = 29.117, we find a higher-order pole at -1.263.

**e.** Pole at -1.263 is near the zero at -1. Simulate to ensure accuracy of results.

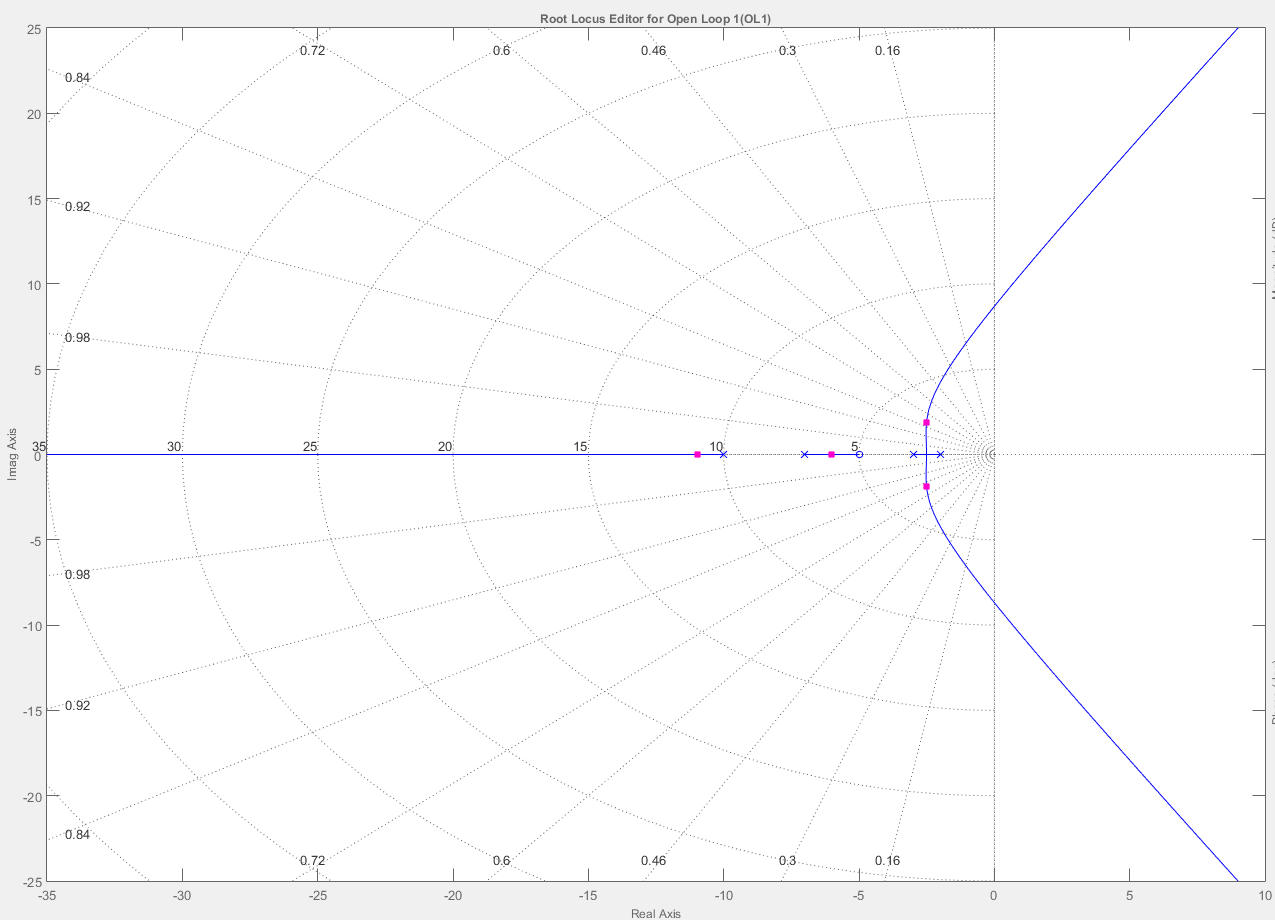
**f.** Ka = = 4.8

**g.**



From the plot, Ts = 1.4 seconds; Tp = 0.68 seconds; %OS = 35%.

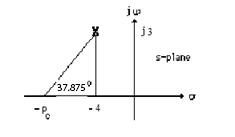
**8.**



**b.** and **c.** Searching along the  = 0.8 line (143.13o), find the operating point at

–2.5 + j1.89 with K =45.8.

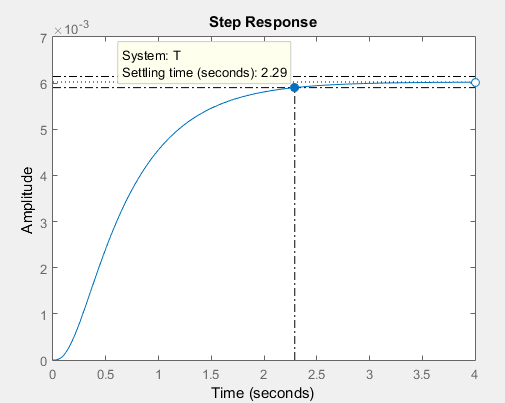
**d.** Since n = , the real part of the compensated dominant pole is -4. The imaginary part is   
4 tan (180o-143.13o) = 3. Using the uncompensated system's poles and zeros along with the compensator zero at - 4, the summation of angles to the design point, -4 + j3 is –142.125o. Thus, the contribution of the compensator pole must be 142.125 - 180o = -37.875. Using the following geometry,  = tan 37.875, or pc = -7.8571.



Adding the compensator pole and using – 4 + j3 as the test point, K = 167.

**e.** Compensated: Searching the real axis segments for K = 67, we find higher-order poles at -13.3, and approximately at –4.3 ± 1.26. Since there is no pole/zero cancellation with the zeros at -5 and –4 and no clear dominant poles, the system should be simulated to check the settling time.

**f.**



The graph shows 0% overshoot and a 2.29 second settling time compared to a desired 1.52% overshoot and a settling time of 1 second.

**9.**

**Program:**

clf

numg=[1 5];

deng=poly([-2 -3 -7 -10]);

'G(s)'

printsys(numg,deng);

rlocus(numg,deng)

z=0.8;

pos=exp(-pi\*z/sqrt(1-z^2))\*100;

sgrid(z,0)

title(['Uncompensated Root Locus with ' , num2str(z), ' Damping Ratio Line'])

[K,p]=rlocfind(numg,deng); %Allows input by selecting point on graphic

'Closed-loop poles = '

p

i=input('Give pole number that is operating point ');

'Summary of estimated specifications'

operatingpoint=p(i)

gain=K

estimated\_settling\_time=4/abs(real(p(i)))

estimated\_peak\_time=pi/abs(imag(p(i)))

estimated\_percent\_overshoot=pos

estimated\_damping\_ratio=z

estimated\_natural\_frequency=sqrt(real(p(i))^2+imag(p(i))^2)

Kp=K\*numg(max(size(numg)))/deng(max(size(deng)))

[numt,dent]=cloop(K\*numg,deng,-1);

'T(s)'

printsys(numt,dent)

'Press any key to continue and obtain the step response'

pause

step(numt,dent)

title(['Step Response for Uncompensated System with ' , num2str(z), ' Damping Ratio'])

'Press any key to go to Lead compensation'

pause

'Compensated system'

b=4

'Lead Zero at -b '

done=1;

while done>0

a=input('Enter a Test Lead Compensator Pole, (s+a). a = ');

numg=conv([1 5],[1 b]);

deng=conv([1 a],poly([-2 -3 -7 -10]));

wn=4/((1)\*z);

rlocus(numg,deng);

sgrid(z,wn)

title(['Lead Compensated Root Locus with ' , num2str(z), ' Damping Ratio Line, Lead Pole at ', num2str(-a), ', and Required Wn'])

done=input('Are you done? (y=0,n=1) ');

end

[K,p]=rlocfind(numg,deng); %Allows input by selecting point on graphic

printsys(numg,deng)

'Closed-loop poles = '

p

i=input('Give pole number that is operating point ');

'Summary of estimated specifications'

operatingpoint=p(i)

gain=K

estimated\_settling\_time=4/abs(real(p(i)))

estimated\_peak\_time=pi/abs(imag(p(i)))

estimated\_percent\_overshoot=pos

estimated\_damping\_ratio=z

estimated\_natural\_frequency=sqrt(real(p(i))^2+imag(p(i))^2)

K\*numg(max(size(numg)))

deng(max(size(deng)))

Kp=K\*numg(max(size(numg)))/deng(max(size(deng)))

[numt,dent]=cloop(K\*numg,deng,-1);

'T(s)'

printsys(numt,dent)

'Press any key to continue and obtain the step response'

pause

step(numt,dent)

title(['Step Response for Compensated System with ' , num2str(z), ' Damping Ratio'])

**Computer response:**

>> ch9p12

ans =

'G(s)'

num/den =

s + 5

------------------------------------

s^4 + 22 s^3 + 161 s^2 + 452 s + 420

Select a point in the graphics window

selected\_point =

-2.5331 + 1.8532i

ans =

'Closed-loop poles = '

p =

-10.9429 + 0.0000i

-6.0461 + 0.0000i

-2.5055 + 1.8517i

-2.5055 - 1.8517i

Give pole number that is operating point 3

ans =

'Summary of estimated specifications'

operatingpoint =

-2.5055 + 1.8517i

gain =

44.4394

estimated\_settling\_time =

1.5965

estimated\_peak\_time =

1.6966

estimated\_percent\_overshoot =

1.5165

estimated\_damping\_ratio =

0.8000

estimated\_natural\_frequency =

3.1155

Kp =

0.5290

ans =

'T(s)'

num/den =

44.4394 s + 222.1968

----------------------------------------------

s^4 + 22 s^3 + 161 s^2 + 496.4394 s + 642.1968

ans =

'Press any key to continue and obtain the step response'

ans =

'Press any key to go to Lead compensation'

ans =

'Compensated system'

b =

4

ans =

'Lead Zero at -b '

Enter a Test Lead Compensator Pole, (s+a). a = 7.8

Are you done? (y=0,n=1) 0

Select a point in the graphics window

selected\_point =

-4.1694 + 2.7980i

num/den =

s^2 + 9 s + 20

---------------------------------------------------------

s^5 + 29.8 s^4 + 332.6 s^3 + 1707.8 s^2 + 3945.6 s + 3276

ans =

'Closed-loop poles = '

p =

-13.1248 + 0.0000i

-4.0751 + 2.7527i

-4.0751 - 2.7527i

-4.2625 + 1.3814i

-4.2625 - 1.3814i

Give pole number that is operating point 2

ans =

'Summary of estimated specifications'

operatingpoint =

-4.0751 + 2.7527i

gain =

154.8317

estimated\_settling\_time =

0.9816

estimated\_peak\_time =

1.1413

estimated\_percent\_overshoot =

1.5165

estimated\_damping\_ratio =

0.8000

estimated\_natural\_frequency =

4.9177

ans =

3.0966e+03

ans =

3276

Kp =

0.9452

ans =

'T(s)'

num/den =

154.8317 s^2 + 1393.4855 s + 3096.6345

--------------------------------------------------------------------

s^5 + 29.8 s^4 + 332.6 s^3 + 1862.6317 s^2 + 5339.0855 s + 6372.6345

ans =

'Press any key to continue and obtain the step response'

**>>**









**10.** a. Searching along the 117.13o line (%OS = 20%; = 0.456), find the operating point at -6.39 + j12.47 with K = 9273. Searching along the real axis for K = 9273, we find a higher-order pole at –47.22. Thus,  second.**b.** For the settling time to decrease by a factor of 2, Re = -n = -6.39 x 2 = -12.78. The imaginary part is Im = -12.78 tan 117.13o = 24.94. Hence, the compensated closed-loop poles are

-12.78 ± j24.94. A settling time of 0.313 second would result. **c.** Assume a compensator zero at -20. Using the uncompensated system's poles along with the compensator zero, the summation of angles to the design point, -12.78 ± j24.94 is –159.63o. Thus, the contribution of the compensator pole must be 159.63o-180o = -20.37o. Using the following geometry, = tan 20.37o, or pc = 79.95.



Adding the compensator pole and using --12.78 ± j24.94 as the test point, K = 74130.

**d.**



**11.**

**a.** Searching along the 110.97o line (*%OS* = 30%; **= 0.358), find the operating point at

-2.065 + j5.388 with *K* = 366.8. Searching along the real axis for *K* = 366.8, we find a higher-order pole at –16.87. Thus,  seconds. For the settling time to decrease by a factor of 2, Re = -*n* = -2.065 x 2 = - 4.13. The imaginary part is – 4.13 tan 110.970 = 10.77. Hence, the compensated dominant poles are – 4.13 ± *j*10.77. The compensator zero is at -7. Using the uncompensated system's poles along with the compensator zero, the summation of angles to the design point, – 4.13 ± *j*10.77 is –162.06o. Thus, the contribution of the compensator pole must be –162.06o - 180o = -17.94o. Using the following geometry, = tan 17.94o, or *pc* = 37.4.



Adding the compensator pole and using – 4.13 ± *j*10.77 as the test point, *K* = 5443.**b.** Searching the real axis segments for *K* = 5443 yields higher-order poles at approximately –8.12 and –42.02. The pole at –42.02 can be neglected since it is more than five times further from the imaginary axis than the dominant pair. The pole at –8.12 may not be canceling the zero at -7. Hence, simulate to be sure the requirements are met.

**c.**

**Program:**

'Uncompensated System G1(s)'

numg1=1;

deng1=poly([-15 (-3+2\*j) (-3-2\*j)]);

G1=tf(numg1,deng1)

G1zpk=zpk(G1)

K1=366.8

'T1(s)'

T1=feedback(K1\*G1,1);

T1zpk=zpk(T1)

'Compensator Gc(s)'

numc=[1 7];

denc=[1 37.4];

Gc=tf(numc,denc)

'Compensated System G2(s) = G1(s)Gc(s)'

K2=5443

G2=G1\*Gc;

G2zpk=zpk(G2)

'T2(s)'

T2=feedback(K2\*G2,1);

T2zpk=zpk(T2)

step(T1,T2)

title(['Uncompensated and Lead Compensated Systems'])

**Computer response:**

ans =

Uncompensated System G1(s)

Transfer function:

1

--------------------------

s^3 + 21 s^2 + 103 s + 195

Zero/pole/gain:

1

-----------------------

(s+15) (s^2 + 6s + 13)

K1 =

366.8000

ans =

T1(s)

Zero/pole/gain:

366.8

---------------------------------

(s+16.87) (s^2 + 4.132s + 33.31)

ans =

Compensator Gc(s)

Transfer function:

s + 7

--------

s + 37.4

ans =

Compensated System G2(s) = G1(s)Gc(s)

K2 =

5443

Zero/pole/gain:

(s+7)

--------------------------------

(s+37.4) (s+15) (s^2 + 6s + 13)

ans =

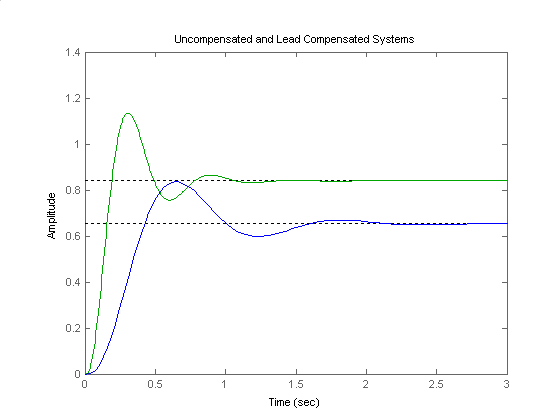
T2(s)

Zero/pole/gain:

5443 (s+7)

-------------------------------------------

(s+42.02) (s+8.118) (s^2 + 8.261s + 133.1)



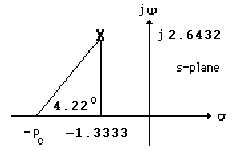
**12.**

**a.** From 20.5% overshoot evaluate . Also, since , . The compensated dominant poles are located at -n ± jn= - 1.3333 ± j2.6432. Assuming

the compensator zero at -0.02, the contribution of open-loop poles and the compensator zero to the design point, - 1.3333 ± j2.6432 is -175.78o. Hence, the compensator pole must contribute

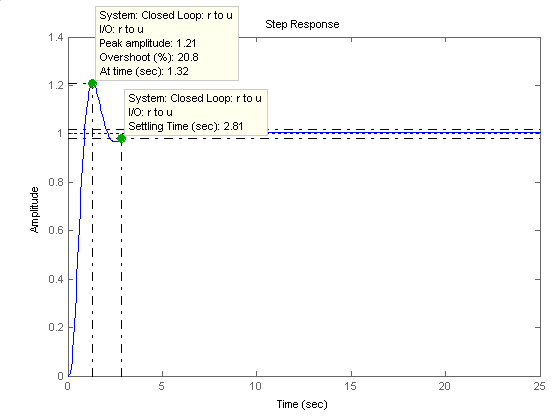
175.78o - 180o = -4.22o. Using the following geometry, , or pc = 37.16

Adding the pole to the system, *K* = 4401.52 at the design point..



**b.** Searching along the real axis segments of the root locus for *K* = 4401.52, we find higher-order poles at -0.0202, -13.46, and -37.02. There is pole/zero cancellation at -0.02. Also, the poles at ,  
 -13.46, and -37.02 are at least 5 times the design point’s real part. Thus, the second-order approximation is valid.

**c.**



From the plot, Ts = 2.81 seconds, and %OS = 20.8%. Thus, the requirements are met.

**13.**

**a.** n = = = 8. Since  = 0.4, n = 20. Therefore the compensated closed-loop poles are located at - n ± jn= -8 ± j18.33.

**b.** Using the system's poles along with the compensator's pole at -15, the sum of angles to the test point –8 ± j18.33 is -293.4o . Therefore, the compensator's zero must contribute 293.4o - 180o = 113.4o . Using the following geometry, = tan 66.6o, or zc = 0.0679.



**c.** Adding the compensator zero and using –8 ± j18.33 as the test point, K = 7297.

**d.** Making a second-order assumption, the predicted performance is as follows:

**Uncompensated:** Searching along the 133.58o line ( = 0.4), find the uncompensated closed-loop pole at -5.43 + j12.45 with K = 3353. Hence, Ts = = 0.74 seconds; = 25.38%; Kp = = 1.66. Checking the second-order assumption by searching the real axis segments of the root locus for K = 3353, we find a higher-order pole at -29.13. Since this pole is more than five times further from the imaginary axis than the dominant pair, the second order assumption is reasonable.

**Compensated:** Using the compensated dominant pole location, - 8 ± j18.33, Ts = = 0.5 seconds; = 25.38%; Kp = = 0.016. Checking the second-order assumption by searching the real axis segments of the root locus for K = 7297, we find higher-order poles at -2.086 and -36.91. The poles are not five times further from the imaginary axis nor do they yield pole/zero cancellation. The second-order assumption is not valid.

**e.**



The uncompensated system exhibits a steady-state error of 0.38, a percent overshoot of 22.5%, and a settling time of 0.78 seconds.



Since there is no pole/zero cancellation the closed-loop zero near the origin produces a large steady-state error. The student should be asked to find a viable design solution to this problem by choosing the compensator zero further from the origin. For example, placing the compensator zero at -20 yields a compensator pole at -90.75 and a gain of 28730. This design yields a valid second-order approximation.

**14.**

**a.** Since %OS = 1.5%,  = = 0.8. Since Ts = = second,

n = 7.49 rad/s. Hence, the location of the closed-loop poles must be -6±j4.49. The summation of angles from open-loop poles to -6±j4.49 is -226.3o. Therefore, the design point is not on the root locus.

**b.** A compensator whose angular contribution is 226.3o-180o = 46.3o is required. Assume a compensator zero at -5 canceling the pole. Thus, the breakaway from the real axis will be at the required -6 if the compensator pole is at -9 as shown below.



Adding the compensator pole and zero to the system poles, the gain at the design point is found to be 29.16. Summarizing the results: Gc(s) = with K = 29.16.

**15.**

**Lead compensator design:** Searching along the 120o line ( = 0.5), find the operating point at

-1.531 + j2.652 with K = 354.5. Thus, Ts = = = 2.61 seconds. For the settling time to decrease by 0.5 second, Ts = 2.11 seconds, or Re = -n = - = -1.9. The imaginary part is

-1.9 tan 60o = 3.29. Hence, the compensated dominant poles are -1.9 ± j3.29. The compensator zero is at -5. Using the uncompensated system's poles along with the compensator zero, the summation of angles to the design point, -1.9 ± j3.29 is -166.09o. Thus, the contribution of the compensator pole must be 166.09o - 180o = -13.91o. Using the following geometry, = tan 13.91o, or pc = 15.18.



Adding the compensator pole and using -1.9 ± j3.29 as the test point, K = 1417.Computer simulations yield the following: Uncompensated: Ts = 3 seconds, %OS = 14.6%. Compensated: Ts = 2.3 seconds, %OS = 15.3%.**Lag compensator design:** The lead compensated open-loop transfer function is   
GLC(s) = . The uncompensated

Kp = 354.5/(2 x 4 x 6 x 8) = 0.923. Hence, the uncompensated steady-state error is = 0.52. Since we want 30 times improvement, the lag-lead compensated system must have a steady-state error of 0.52/30 = 0.017. The lead compensated Kp = 1417\*5/(2\*4\*6\*8\*15.18) = 1.215. Hence, the lead-compensated error is = 0.451. Thus, the lag compensator must improve the lead-compensated error by 0.451/0.017 = 26.529 times. Thus 0.451/ ( ) = 26.529, where Kpllc = 57.823 is the lead-lag compensated system's position constant. Thus, the improvement in Kp from the lead to the lag-lead compensated system is 57.823/1.215 = 47.59. Use a lag compensator, whose zero is 47.59 times farther than its pole, or Glag = . Thus, the lead-lag compensated open-loop transfer function is GLLC(s) = .

**16.**

**a.**  For the settling time to be 2.86 seconds with 4.32% overshoot, the real part of the compensated dominant poles must be = = 1.4. Hence the compensated dominant poles are -1.4 ± j1.4. Assume the compensator zero to be at -1 canceling the system pole at -1. The summation of angles to the design point at -1.4 ± j1.4 is -176.19o. Thus the contribution of the compensator pole is

176.19o - 180o = 3.81o. Using the geometry below, = tan 3.81o, or pc = 22.42.



Adding the compensator pole and using -1.4 ± j1.4 as the test point, K = 88.68.

**b.** **Uncompensated:** Search the 135o line (4.32% overshoot) and find the uncompensated dominant pole at - 0.419 + j0.419 with K = 1.11. Thus Kv = = 0.37. Hence, Ts = = = 9.55 seconds and %OS = 4.32%. Compensated: Kv = = 1.32 (Note: steady-state error improvement is greater than 2). Ts = = = 2.86 seconds and %OS = 4.32%.

**c.** **Uncompensated:** K = 1.11; Compensated: K = 88.68.**d.** **Uncompensated:** Searching the real axis segments for K = 1.11 yields a higher-order pole at -3.16 which is more than five times the real part of the uncompensated dominant poles. Thus the second-order approximation for the uncompensated system is valid.**Compensated:** Searching the real axis segments for K = 88.68 yields a higher-order pole at -22.62 which is more than five times the real part of the compensated dominant poles' real part. Thus the second order approximation is valid.

**e.**

****

****

****

****

**17.**

**a.** Searching the 30% overshoot line ( = 0.358; 110.97o) for 180o yields -1.464 + j3.818 with a gain, K = 218.6.**b.** Tp = = = 0.823 second. Kv = = 3.975.**c.** **Lead design:** From the requirements, the percent overshoot is 15% and the peak time is 0.4115 second. Thus,  = = 0.517; d = = 7.634 = n. Hence, n = 8.919. The design point is located at -n + jn= -4.61 + j7.634. Assume a lead compensator zero at -5. Summing the angles of the uncompensated system's poles as well as the compensator zero at -5 yields –171.2o. Therefore, the compensator pole must contribute (171.2o - 180o) = -8.8o. Using the geometry below,



= tan (8.8o) . Hence, pc = 53.92. The compensated open-loop transfer function is . Evaluating the gain for this function at the point, -4.61 + j7.634 yields

K = 4430.

**Lag design:** The uncompensated . The required Kv is 30\*3.975 = 119.25. The lead compensated Kv = = 7.469. Thus, we need an improvement over the lead compensated system of 119.25/7.469 = 15.97. Thus, use a lag compensator

Glag(s) = . The final open-loop function is .

**18.**

**a.** Uncompensated: Search the 135o line (4.32% overshoot) for 180o and find the dominant pole at

–3 + j3 with K = 10.

Lag Compensated: Search the 135o line (4.32% overshoot) for 180o and find the dominant pole at -2.88 + j2.88 with K = 9.95.

**b.** Uncompensated: Kp = = 1.25

Lag compensated: Kp = = 6.22

**c.** %OS = 4.32% both cases;

Uncompensated Ts = = 1.33 seconds; Compensated Ts = = 1.39 seconds

**d.** Uncompensated: Exact second-order system; approximation OK

Compensated: Search real axis segments of the root locus and find a higher-order pole at -0.3. System should be simulated to see if there is effective pole/zero cancellation with zero at - 0.5.

**e.**



The compensated system's response takes a while to approach the final value.

**f.** We will design a lead compensator to speed up the system by a factor of 5. The lead-compensated dominant poles will thus be placed at –15 ± j15. Assume a compensator zero at - 4 that cancels the open-loop pole at - 4. Using the system's poles and the compensator's zero, the sum of angles to the design point, -15±j15 is 131.69o. Thus, the angular contribution of the compensator pole must be 131.69o - 180o = - 48.31o. Using the geometry below, pc = 28.36.



Using the compensated open-loop transfer function, Ge(s) = and using the design point –15 ± j15, K = 404.1.The time response of the lag-lead compensated system is shown below.



**19.**

Since Tp = 1.122, the imaginary part of the compensated closed-loop poles will be:

= 2.8. Since = tan (cos-1), the magnitude of the real part is = 2.8.

Hence, the design point is – 2.8 + j2.8. Assume a PI controller with a transfer function:

Gc(s) = , to reduce the steady-state error to zero.

Using the system's poles and the pole and zero of the ideal integral compensator, the summation of angles to the design point is - 209.7o. Hence, the ideal derivative controller must contribute 209.7o -180o = 29.7o. Using the geometry below, zc = 7.71.

s-plane

j





j2.8

-2.8

-zc

29.7o

X

The PID controller is thus . Using all poles and zeros of the system and PID controller, the gain at the design point is K = 1.683. Searching the real axis segment, a higher-order pole is found at - 0.0828. A simulation of the system shows the requirements are met (See the root locus and step-response graphs shown below.)



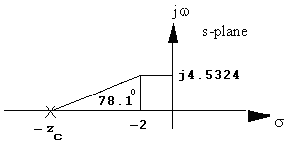


**20.**

**a.** The desired operating point is found from the desired specifications.  and

. Thus, . Hence the design point is –2 +j4.5324. Now, add a pole at the origin to increase system type and drive error to zero for step inputs.

Now design a PD controller. The angular contribution to the design point of the system poles and pole at the origin is 101.90. Thus, the compensator zero must contribute 1800 – 101.90 =78.10. Using the geometry below,



 . Hence, *zc* = 2.955. The compensated open-loop transfer function with PD compensation is . Adding the compensator zero to the system and evaluating the gain for this at the point –2 + j4.5324 yields K = 294.51 with a higher-order pole at   
-2.66 and -13.34.

**PI design:** Use . Hence, the equivalent open-loop transfer function is

 with K = 294.75.

**b.**

**Program (Step Response):**

numg=[-2.995 -0.01];

deng=[0 0 -4 -6 -10];

K=294.75;

G=zpk(numg,deng,K)

T=feedback(G,1);

step(T)

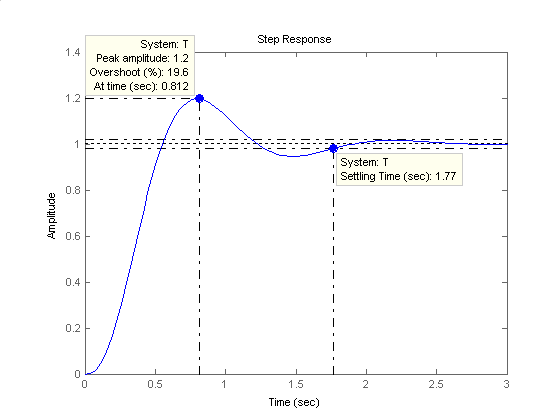
**Computer response:**

Zero/pole/gain:

294.75 (s+2.995) (s+0.01)

-------------------------

s^2 (s+4) (s+6) (s+10)



**Program (Ramp Response):**

numg=[-2.995 -0.01];

deng=[0 0 -4 -6 -10];

K=294.75;

G=zpk(numg,deng,K)

T=feedback(G,1);

Ta=tf([1],[1 0]);

step(T\*Ta)

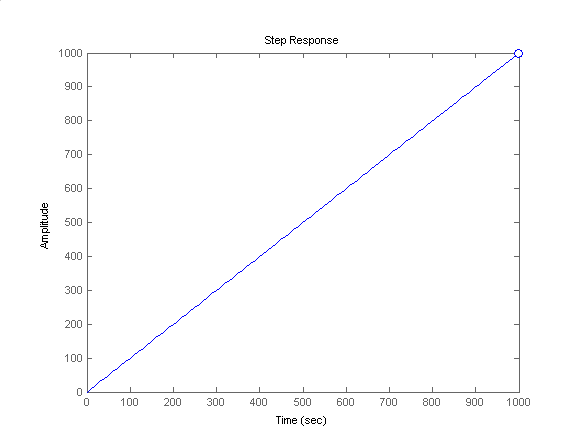
**Computer response:**

Zero/pole/gain:

294.75 (s+2.995) (s+0.01)

-------------------------

s^2 (s+4) (s+6) (s+10)



**21.**

**Program:**

numg=[]

deng=[-4 -6 -10]

'G(s)'

G=zpk(numg,deng,1)

pos=input('Type desired percent overshoot ');

z=-log(pos/100)/sqrt(pi^2+[log(pos/100)]^2);

Ts=input('Type desired settling time ');

zci=input(...

'Type desired position of integral controller zero (absolute value) ');

wn=4/(Ts\*z);

desired\_pole=(-z\*wn)+(wn\*sqrt(1-z^2)\*i)

angle\_at\_desired\_pole=(180/pi)\*angle(evalfr(G,desired\_pole))

PD\_angle=180-angle\_at\_desired\_pole;

zcpd=((imag(desired\_pole)/tan(PD\_angle\*pi/180))-real(desired\_pole));

'PD Compensator'

numcpd=[1 zcpd];

dencpd=[0 1];

'Gcpd(s)'

Gcpd=tf(numcpd,dencpd)

Gcpi=zpk([-zci],[0],1)

Ge=G\*Gcpd\*Gcpi

rlocus(Ge)

sgrid(z,0)

title(['PID Compensated Root Locus with ' ,...

num2str(pos), '% Damping Ratio Line'])

[K,p]=rlocfind(Ge);

'Closed-loop poles = '

p

f=input('Give pole number that is operating point ');

'Summary of estimated specifications for selected point'

'on PID compensated root locus'

operatingpoint=p(f)

gain=K

estimated\_settling\_time=4/abs(real(p(f)))

estimated\_peak\_time=pi/abs(imag(p(f)))

estimated\_percent\_overshoot=pos

estimated\_damping\_ratio=z

estimated\_natural\_frequency=sqrt(real(p(f))^2+imag(p(f))^2)

T=feedback(K\*Ge,1);

step(T)

title(['Step Response for PID Compensated System with ' ,...

num2str(pos),'% Damping Ratio Line'])

pause

one\_over\_s=tf(1,[1 0]);

Tr=T\*one\_over\_s;

t=0:0.01:10;

step(one\_over\_s,Tr)

title('Ramp Response for PID Compensated System')

**Computer response:**

numg =

[]

deng =

0 -4 -6 -10

ans =

G(s)

Zero/pole/gain:

1

--------------------

s (s+4) (s+6) (s+10)

Type desired percent overshoot 25

Type desired settling time 2

Type desired position of integral controller zero (absolute value) 0.01

desired\_pole =

-2.0000 + 4.5324i

angle\_at\_desired\_pole =

101.8963

ans =

PD Compensator

ans =

Gcpd(s)

Transfer function:

s + 2.955

Zero/pole/gain:

(s+0.01)

--------

s

Zero/pole/gain:

(s+2.955) (s+0.01)

----------------------

s^2 (s+4) (s+6) (s+10)

Select a point in the graphics window

selected\_point =

-1.9931 + 4.5383i

ans =

Closed-loop poles =

p =

-13.3485

-1.9920 + 4.5377i

-1.9920 - 4.5377i

-2.6575

-0.0100

Give pole number that is operating point 2

ans =

Summary of estimated specifications for selected point

ans =

on PID compensated root locus

operatingpoint =

-1.9920 + 4.5377i

gain =

295.6542

estimated\_settling\_time =

2.0081

estimated\_peak\_time =

0.6923

estimated\_percent\_overshoot =

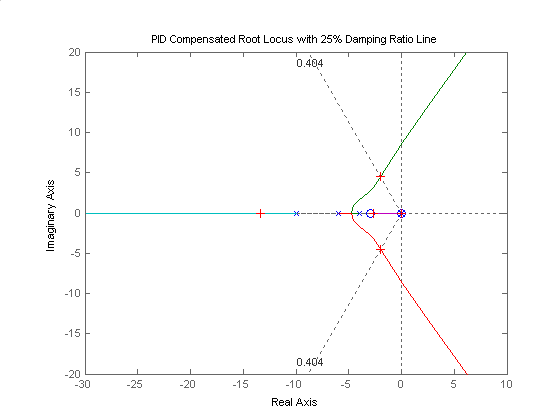
25

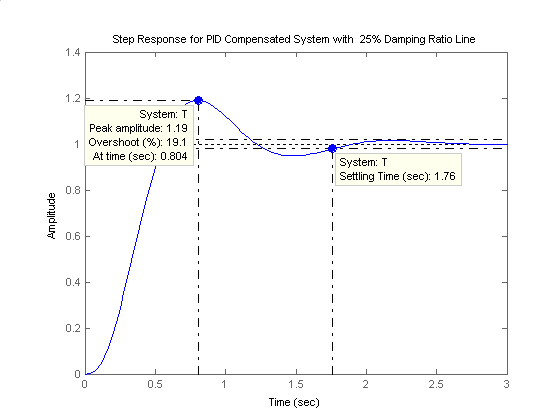
estimated\_damping\_ratio =

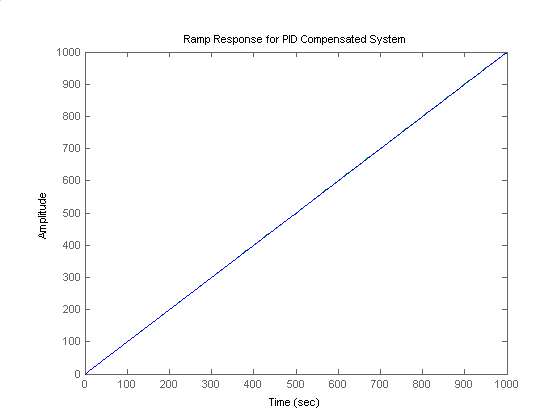
0.4037

estimated\_natural\_frequency =

4.9557







**22.**

Open-loop poles are at -2, - 0.2, and -1.8. An open-loop zero is at -3. Searching the 117.13o line ( = 0.456), the closed-loop dominant poles are found at -0.758 + j1.48 with K = 2.06 (see zoomed-in root-locus below). Searching the real axis segments (of the full locus) locates a higher-order pole at -2.54.

Since the open-loop zero is a zero of H(s), it is not a closed-loop zero. Thus, there are no closed-loop zeros.



**23.**

**a.** The damping ratio for 15% overshoot is 0.517. The desired operating point is found from the desired specifications.  and . Thus, . Hence the design point is –1.333 + j2.207. The angular contribution of the system poles and compensator zero at the design point is 100.80. Thus, the compensator zero must contribute 1800 – 100.80 = 79.20. Using the geometry below,



= tan (79.2o) . Hence, *zc* = 1.754. The compensated open-loop transfer function with PD compensation is . Evaluating the gain for this function at the point

–1.333 + j2.207 yields K = 47.28 with higher-order poles at –1.617 and –7.718. Following

Figure 9.49(c) in the text, . Therefore, . Also, using the notation of Figure 9.49(c), , from which .

**b.**

**Program:**

K1=82.93;

numg=K1;

deng=poly([0 -2 -4 -6]);

'G(s)'

G=tf(numg,deng);

Gzpk=zpk(G)

Kf=0.5701

numh=Kf\*[1 1.754];

denh=1

'H(s)'

H=tf(numh,denh);

Hzpk=zpk(H)

'T(s)'

T=feedback(G,H);

T=minreal(T)

step(T)

title('Step Response for Feedback Compensated System')

**Computer response:**

ans =

G(s)

Zero/pole/gain:

82.93

-------------------

s (s+6) (s+4) (s+2)

Kf =

0.5701

denh =

1

ans =

H(s)

Zero/pole/gain:

0.5701 (s+1.754)

ans =

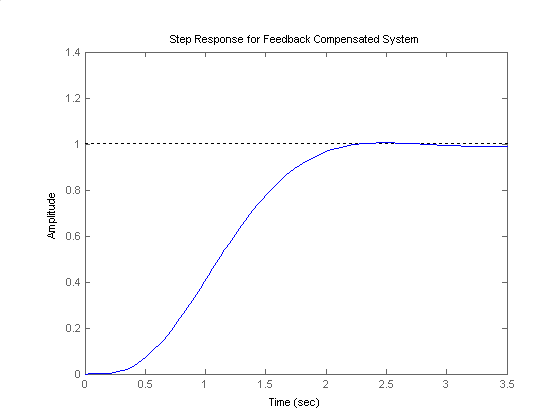
T(s)

Transfer function:

82.93

---------------------------------------

s^4 + 12 s^3 + 44 s^2 + 95.28 s + 82.93

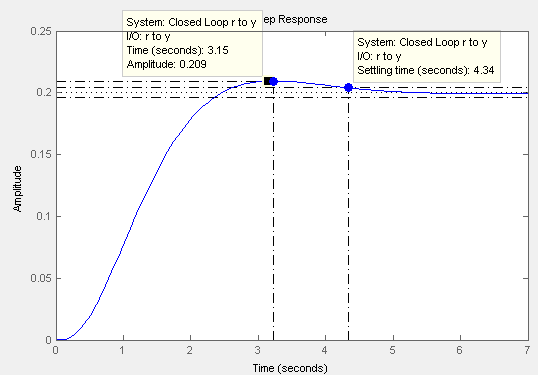


**24.**

**a**. The real part of the design point  ; %OS=5% corresponds to  from which the imaginary part is . Thus the design point is . The sum of angles from the minor-loop’s open-loop poles to the design point is . The zero must provide  of phase lead. The position of the zero is obtained from the geometry below calculating  . So .



Adding the zero and calculating the gain at the design point yields . Therefore the minor-loop transfer function is . The equivalent minor-loop closed-loop transfer function is  . A simulation of the step response of the minor-loop is shown below.



1. The major-loop open-loop transfer function is . Drawing the root locus using  and searching along the 10% overshoot line ( ) for  yields the point  with a gain .

**Program:**

numg=2.3842;

deng=[1 7 12.11 10.55];

'G(s)'

ans =

G(s)

G=tf(numg,deng)

G =

2.384

-----------------------------

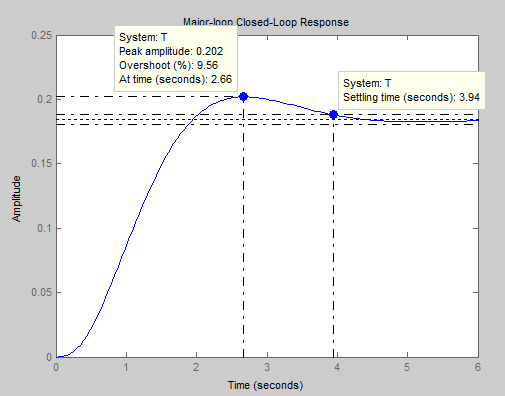
s^3 + 7 s^2 + 12.11 s + 10.55

Continuous-time transfer function.

T=feedback(G,1);

step(T)

title('Major-loop Closed-Loop Response')



1. Adding the PI compensator 

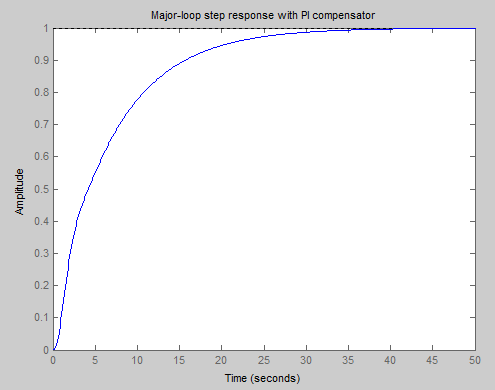
s=tf('s');

Ge= 2.27\*(s+0.7)/(s^4 + 7\*s^3 + 12.11\*s^2 + 10.55\*s);

T=feedback(Ge,1);

step(T)

title('Major-loop step response with PI compensator')



**25.**

Uncompensated System: Search along the line for ξ=0.4. The operating point is  with . The corresponding %OS=25.4%;  sec.  .

Compensated a: A pole is added at the origin and a zero at -0.1 to form the PI controller. Search along the ξ=0.4 line. The operating point is  with . The estimated %OS=25.4% as with the uncompensated.  sec,  . Higher order poles are located at -0.0793 and -11.2.

Compensated b: A pole is added at the origin and a zero at -0.7 to form the PI controller. Search along the ξ=0.4 line. The operating point is  with . The estimated %OS=25.4% as with the uncompensated.  sec, . Higher order poles are located at -0.641 and -11.

Step response simulations for the 3 systems are shown:



Compensated system a is unacceptable because the generated closed loop pole is too close to the origin creating a slow system. The Compensated b system has a more desirable response.

**26.**

The matrix **A** was found to be:



1. The signal-flow diagram for that open-loop system is shown below.

1

1

- 0.9401

0.4167

1

y

u

*θ*

*x*1

*x*2

-1.1574

u

u

*x*2

u

u

*x*2



29.865

*x*4

1

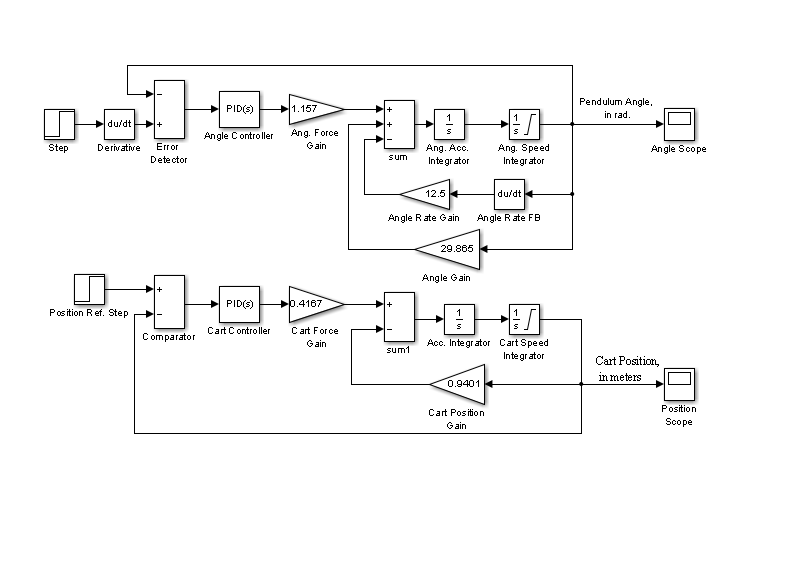
1

*x*3

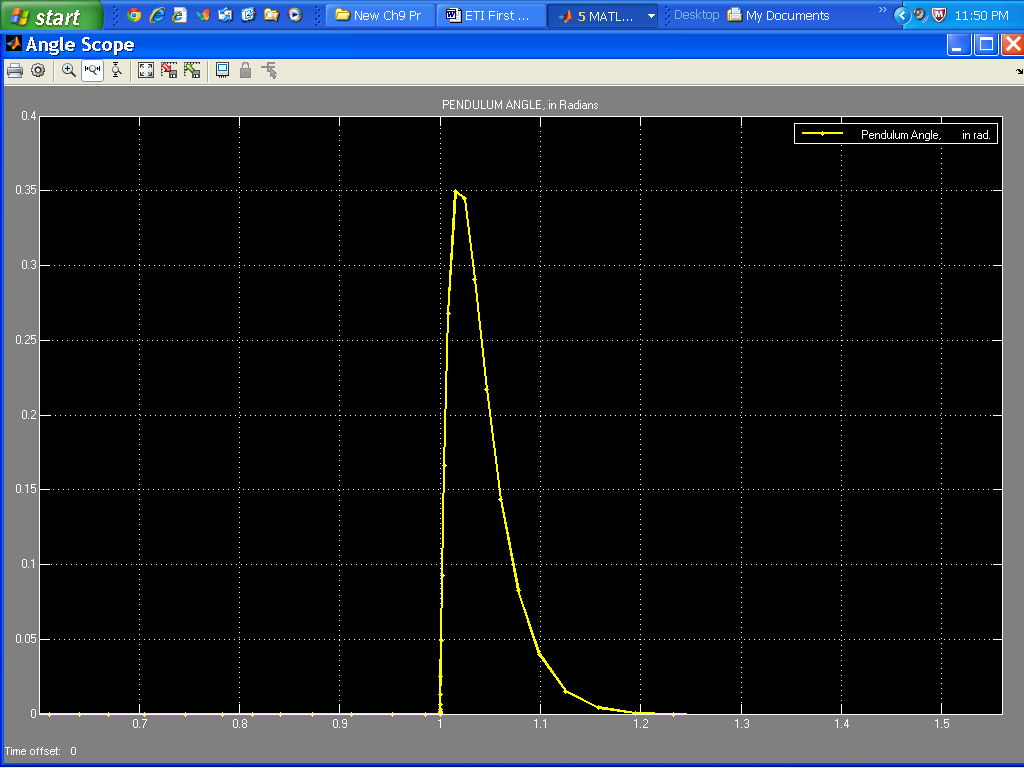
*x*

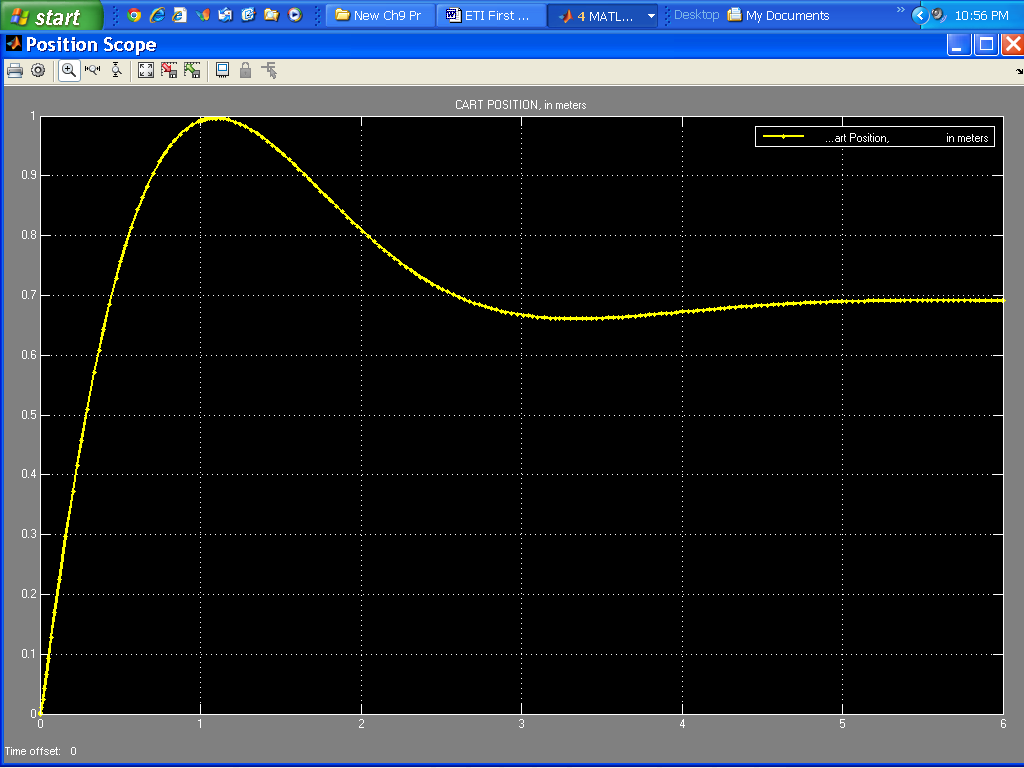


That diagram was then used to develop the following Simulink model. As could be seen, rate feedback with a gain of 12.5 to stabilize the pendulum angle control system was added. The PID blocks were configured as PD controllers with the recommended settings.



Using scopes, the following graphs were captured for the two responses: the impulse response of the pendulum angle in radians and the step response of the cart position in meters. These graphs indicate that not only was the system stabilized, the two responses are remarkably fast with the pendulum angle going down to zero in 1.2 seconds and the cart position having *TP* ≈ 1.1 seconds. The steady-state error in cart position, however, is 31%, which is unacceptable.





To have Tp = 1.2 sec, the imaginary part of the compensated closed-loop poles will be:

= 2.618. For a 20.5% OS, *ζ* = 0.456. Since = tan (cos-1), the magnitude of the real part is = 1.341. Hence, the design point is – 1.341 + j2.618.

Assume a PI controller with a transfer function: Gc(s) = , to reduce the steady-state error to zero. Using the system's poles (± j 0.9696) and the pole and zero of the ideal integral controller, the summation of angles to the design point is - 241.4o. Hence, the ideal derivative controller must contribute 241.4o -180o = 61.4o.

Using the geometry below, zc = 2.77.

s-plane

j





j2.618

-1.341

-zc

61.4o

X

Thus, the recommended settings for the PID controller may be:

*KP* = 7, *KD =* 2.44, and *KI* = 0.676.

As could be seen from the graph shown below, the cart position step response obtained with these PID controller settings satisfies all requirements.



**27.**

**a.** **PI controller:** Using Table 9.10, = , R2C = 100. Let C = 25 F. Therefore, R2 = 4 M. For unity gain, R1 = 4 M. Compensate elsewhere in the loop for the compensator negative sign.

**b**. **PD controller:** Using Table 9.10, R2C(s+) = s+2. Hence, R1C = 0.5. Let C = 1 F. Therefore, R1 = 500 K. For unity gain, R2C = 1, or R2 = 1 M. Compensate elsewhere in the loop for the compensator negative sign.

**28.**

**a.** **Lag compensator:** See Table 9.11.  = . Thus, R2C = 10, and

(R1 + R2)C = 100. Letting C = 10  F, we find R2 = 1 M. Also R1C = 100 - R2C = 90, which yields R1 = 9 M. The loop gain also must be multiplied by .**b.** **Lead compensator:** See Table 9.11. = . Thus, R1C = 0.5, and

+ = 5. Letting C = 1 F, R2 = 333 K, and R1 = 500 K. **c.** Lag-lead compensation: See Table 9.11.  
= . Thus, R1C1 = 1, and   
R2C2 = 10. Also, + + = 1 + 0.1 + = 10.01, or R2C1 = 0.112. Letting C1 = 10 F, we find R1 = 10 M , R2 = 1.12 M, and C2 = 8.9 F.

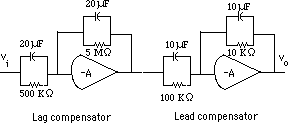
**29.**

**a.** **Lag compensator:** See Table 9.10 and Figure 9.58. = . Therefore,

R1C1 = 10; R2C2  = 100. Letting C1 = C2 = 20 F, we find R1 = 500 K and R2 = 5 M. Compensate elsewhere in the loop for the compensator negative sign.

**b.** **Lead compensator:** See Table 9.10 and Figure 9.58. = . Therefore,

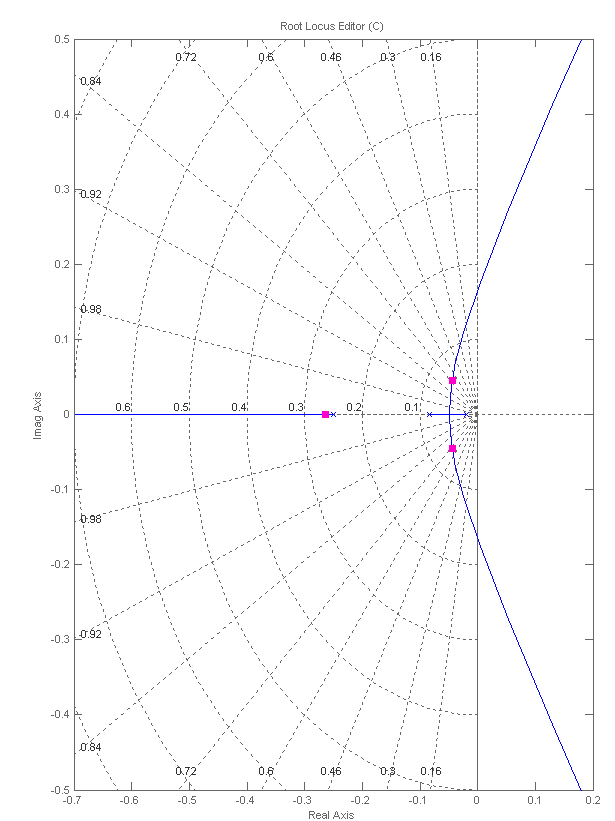
R1C1 = 0.5 and R2C2 = 0.2. Letting C1 = C2 = 20 F, we find R1 = 25 K and R2 = 10 M. Compensate elsewhere in the loop for the compensator negative sign.**c. Lag-lead compensator:** See Table 9.10 and Figure 9.58. For lag portion, use (a). For lead:   
= . Therefore, R1C1 = 1 and R2C2 = 0.1. Letting C1 = C2 = 10 F, we find   
R1 = 100 K and R2 = 10 K. The following circuit can be used to implement the design.



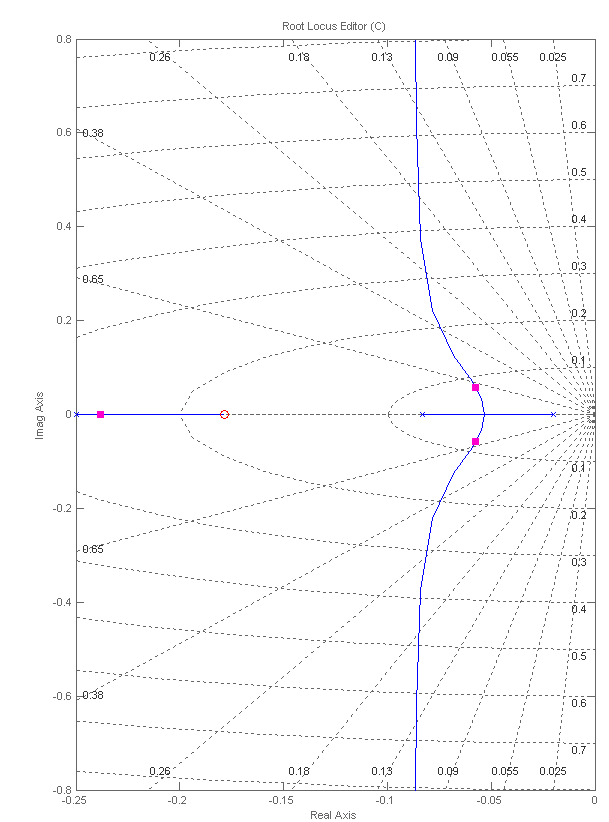
**SOLUTIONS TO DESIGN PROBLEMS**

**30.**

1. The system’s root locus shows that is achieved when , with a corresponding . At this value of gain the system has dominant poles at  and a third pole at -0.265 justifying the dominant pole approximation.



1. We calculate the desired closed loop positions. The desired so the real part of the poles is . Since we want to maintain , the imaginary part is found from resulting in . So the desired closed loop poles are . The compensator has the form . A search using MATLAB’s sisotool gives with . The resulting root locus is





>> syms s

>> s=tf('s');

>> Gv = 0.02/(4\*s+1);

>> G1 = 70/(50\*s+1);

>> H = 1/(12\*s+1);

>> G = Gv\*G1’

>> T=1.11\*G/(1+1.11\*G\*H);

>> C=11.424\*(s+0.178);

>> T2=C\*G/(1+C\*G\*H);

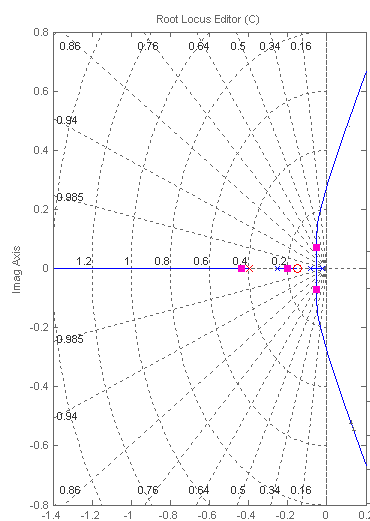
>> step(T,T2)



Note that the steady state error is also varied through ideal PD design because this is a type 0 system.

**31.**

1. This part is identical to the one in Problem 39.
2. As is problem III.b the desired closed loop pole positions are: . The lead compensator has the form . We arbitrarily choose , and perform a numerical search until the 180o angle condition is satisfied on the open loop transmission giving . Then the gain is adjusted in sisotool until the poles are placed at their desired positions. The resulting compensator is: . The resulting root locus is:



**c.**

>> syms s

>> s=tf('s');

>> Gv=0.02/(4\*s+1);

>> G1=70/(50\*s+1);

>> H=1/(12\*s+1);

>> G=Gv\*G1;

>> T=1.11\*G/(1+1.11\*G\*H);

>> C=6.62\*(s+0.15)/(s+0.396);

>> T2=C\*G/(1+C\*G\*H);

>> step(T,T2)



**32**.

**a.** =

Kb = = = 0.005; Jeq = 5 (x )2 = 0.05; Deq = 1 ()2 = 0.01;

= = = 0.1. Therefore, = .

**b.** The block diagram of the system is shown below.



Forming an equivalent unity feedback system,



Now, T(s) = . Thus, n = ; 2n = 0.21 + 0.2Kt. Since  = 0.5, Kt = 157.06.

**c. Uncompensated:** Kt = 0; T(s) = ; n = 31.62 rad/s;  = 3.32 x 10-3;   
 = 98.96%; Ts = = 38.09 seconds;

Tp = = 9.93 x 10-2 second; Kv = = 4761.9.

**Compensated:** Kt = 157.06; T(s) = ; n = 31.62 rad/s;  = 0.5;   
= 16.3%; Ts = = 0.253 second; Tp = = 0.115 second;   
Kv = = 31.63.

**33.**

**a.** The transfer functions of the subsystems are as follows:

Pot: Gp(s) = = ; Amplifier: Ga(s) = ; Motor and load: Since the time to rise to 63% of the final value is 0.5 second, the pole is at -2. Thus, the motor transfer function is of the form, Gm(s) = . But, from the problem statement, = , or K = 20. The block diagram of the system is shown below.



Using the equivalent system, search along the 117.126o line (20% overshoot) and find the dominant second-order pole at - 0.89 + j1.74 with K = 10K1 = 77.4. Hence, K1 = 7.74.

**b.** Kv = = 1.935. Therefore, e(∞) = = 0.517.

**c.** %OS = 20%;  = = 0.456; n = = 1.95 rad/s;   
Ts = = 4.49 seconds; Tp = = 1.81 seconds.

**d.** The block diagram of the minor loop is shown below.



The transfer function of the minor loop is GML(s) = . Hence, the block diagram of the equivalent system is



where a = 2 + 20Kf. The design point is now found. Since %OS = 20%,  = = 0.456. Also, since Ts = = 2 seconds, n = 4.386 rad/s. Hence, the design point is –2 + j3.9. Using just the open-loop poles at the origin and at -20, the summation of angles to the design point is -129.37o. The pole at -a must then be contributing 129.37o - 180o = -50.63o. Using the geometry below, a = 5.2, or Kf = 0.16.



Adding the pole at -5.2 and using the design point, we find 10K1 = 407.23, or K1 = 40.723.

Summarizing the compensated transient characteristics:  = 0.456; n = 4.386; %OS = 20%; Ts = = 2 seconds; Tp = = 0.81 seconds; Kv = = 3.92.

**34.**

Block diagram

**Preamplifier/Power amplifier:** ; Pots: = 2.

**Torque-speed curve:**



where 1432.35 x x 2= 150 rad/sec; 477.45 x x 2 = 50 rad/sec. The slope of the line is - = - 0.5. Thus, its equation is y = -0.5x + b. Substituting one of the points, find b = 100. Thus Tstall = 100, and no load = 200. = = = 2; Kb = = = 0.25.

**Motor:** = = , where J = 100, D = 50.

**Gears:** 0.1

Drawing block diagram:





**b. Compensator design - Lead**

10% overshoot and Ts = 1 sec yield a design point of - 4 + j5.458. Sum of angles of uncompensated system poles to this point is -257.491o. If we place the lead compensator zero over the uncompensated system pole at -0.505, the angle at the design point is -134.858o. Thus, the lead compensator pole must contribute 134.858o - 180o = -45.142o. Using the geometry below

= tan(45.142o), or pc = 9.431.



Using the uncompensated poles and the lead compensator, the gain at the design point is

0.004K1 = 1897.125.

**Compensator design - Lag**

With lead compensation, Kv = = 5.0295.029. Since we want Kv = 1000, = = 198.85. Use plag = 0.001. Hence zlag = 0.1988. The lag compensated

Ge(s) = .

**c. Compensator schematic**

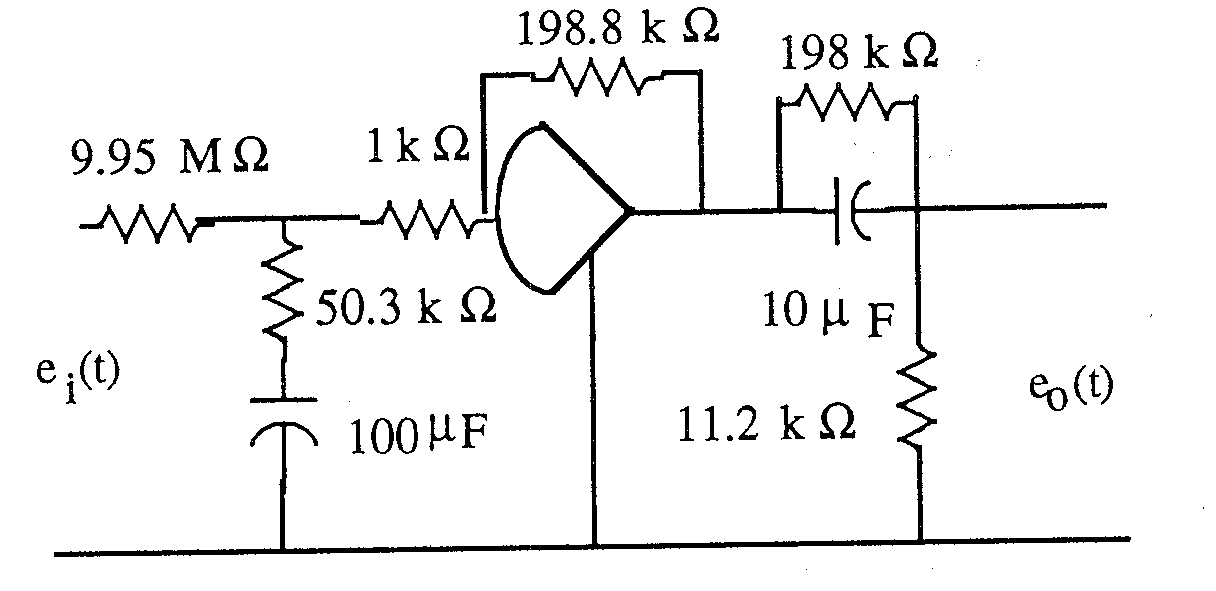
lag: = 0.1988. Let C = 100 F. Then R2 = 50.3 kNow, = 0.001.

Thus, R1 = 9.95 MBuffer gain = reciprocal of lag compensator's . Hence buffer

gain = = 198.8.

lead: = 0.505. Let C = 10 F. Then R1 = 198 kNow + = 9.431.

Thus, R2 = 11.2 k.



**d.**

**Program:**

numg= 1897.125\*[1 0.1988];

deng=poly([0 -40 -9.431 -.001]);

'G(s)'

G=tf(numg,deng);

Gzpk=zpk(G)

rlocus(G)

pos=10

z=-log(pos/100)/sqrt(pi^2+[log(pos/100)]^2)

sgrid(z,0)

title(['Root Locus with ' , num2str(pos), ' Percent Overshoot Line'])

[K,p]=rlocfind(G) %Allows input by selecting point on graphic

pause

T=feedback(K\*G,1);

step(T)

title(['Step Response for Design of ' , num2str(pos), ' Percent'])

**Computer response:**

ans =

G(s)

Zero/pole/gain:

1897.125 (s+0.1988)

----------------------------

s (s+40) (s+9.431) (s+0.001)

pos =

10

z =

0.5912

Select a point in the graphics window

selected\_point =

-3.3649 + 4.8447i

K =

0.9090

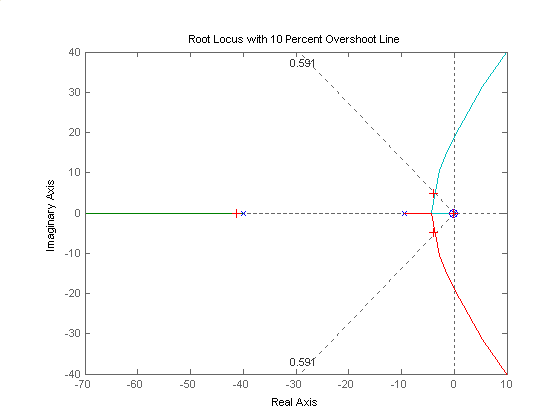
p =

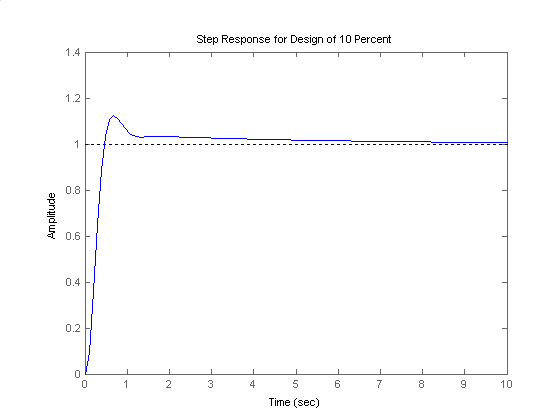
-41.3037

-3.9602 + 4.9225i

-3.9602 - 4.9225i

-0.2080



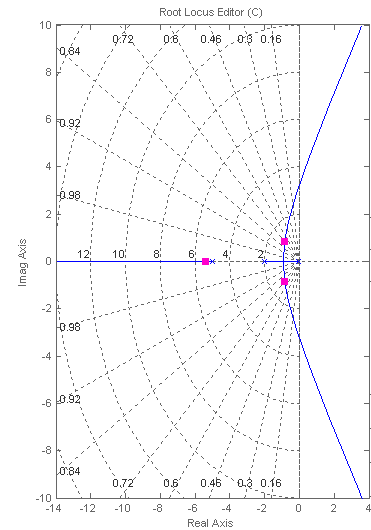


**35.**

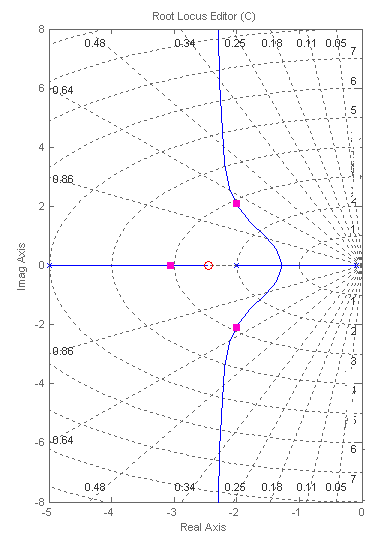
Consider only the minor loop. Searching along the 143.13o line ( = 0.8), locate the minor-loop dominant poles at -3.36 ± j2.52 with Kf = 8.53. Searching the real axis segments for Kf = 8.53 locates a higher-order pole at - 0.28. Using the minor-loop poles as the open-loop poles for the entire system, search along the 120o line ( = 0.5) and find the dominant second-order poles at -1.39 + j2.41 with K = 27.79. Searching the real axis segment locates a higher-order pole at - 4.2.

**36.**

1. The open loop transfer function is  . The root locus for the system is shown below. With  the system has a damping factor for the dominant poles which are located at . The third pole is located at -5.4 so the 2nd order approximation applies. The resulting  with a %OS=4.2%



1. The desired  sec, so the real part of the desired closed loop poles is . The imaginary part is obtained from . So the desired dominant closed loop positions are . We start by doing PD design with a compensator of the form  . A numerical search results in , and  gives the desired pole positions. So . The root locus is shown next where it can be seen that the third pole is at approximately -3.06, so the dominant pole approximation is not as accurate as in part a)



For the PI part we arbitrarily choose a zero close to the origin. After adjusting the gain to obtain the desired damping factor, the resulting PID compensator is , the third pole moving to -3.16.

**c.**

>>syms s

>>s=tf('s');

>>Gc=9.717\*(s+0.1)\*(s+2.456)/s;

>>G=1/(s+0.08)/(s+2)/(s+5);

>>T=6.95\*G/(1+6.95\*G); % No compensation

>>T2=Gc\*G/(1+Gc\*G); % PID Compensation

>> step(T,T2,6) % Simulate up to 6 sec

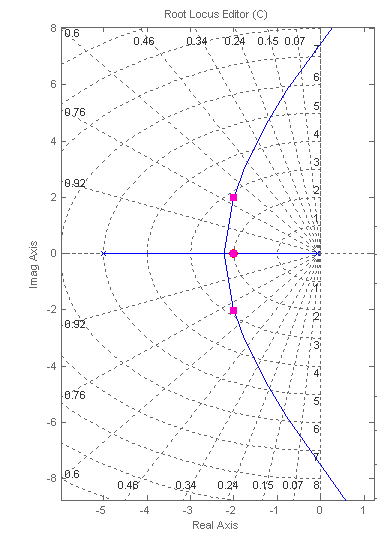


Note that the compensated response results in the desired Settling time, however the resulting %OS≈8% is larger than expected due to the third pole being close to the dominant poles.

**37.**

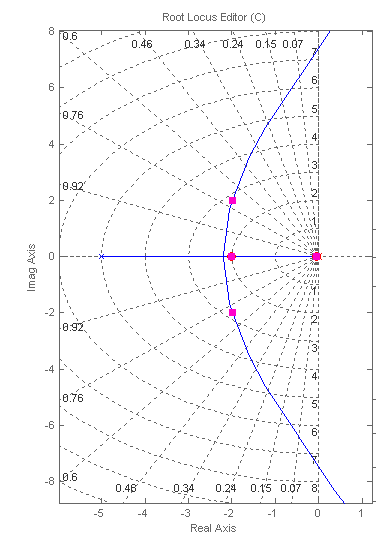
**a.**  This part is identical to 47(a)

**b.**  We start by designing the lead compensator. The desired closed loop pole positions are as in problem IV. We arbitrarily choose the position of the compensator zero at -2 (cancelling a plant pole), and do a numerical search to get the pole position as -11.8, then adjust the gain until the desired pole positions are obtained. At this stage we have . The resulting roots locus is:



Now we do the lag compensator design by first choosing arbitrarily the lag compensators pole position as -0.01 so the systems open loop transmission is:

 . The steady state requirements demand , so . The lead lag compensator is: . The roots locus is:



**c.**

>> syms s

>> s=tf('s');

>> G=1/(s+0.08)/(s+2)/(s+5);

>> Gc = 95.7\*(s+0.0463)\*(s+2)/(s+0.01)/(s+11.2);

>> T=6.95\*G/(1+6.95\*G); %Uncompensated System

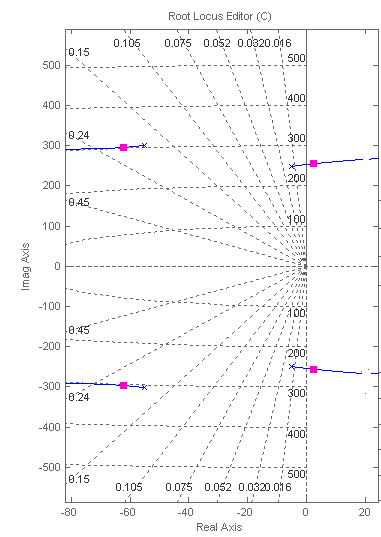
>> T2=Gc\*G/(1+Gc\*G); %Compensated System

>> step(T,T2,6) %Simulate for up to 6 sec

**38.**

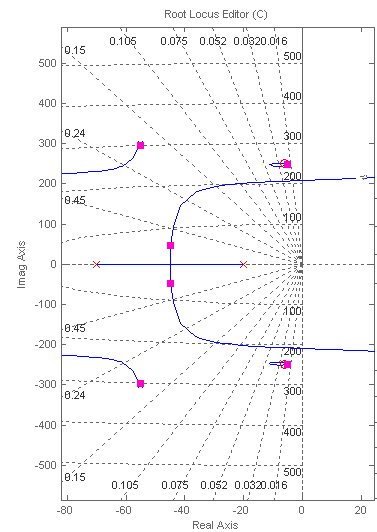
**a. , b., c.**

The root locus for the uncompensated system is shown next. The range for closed loop stability is . The fastest system will occur for vary small values of  with . The dominant poles have a damping factor 



**d**., **e.**

The notch filter is designed by arbitrarily placing two complex conjugate zeros close to the dominant complex pole plant pair, and then arbitrarily adding two ‘far away’ real poles in between the two pairs of plant poles. The gain is adjusted to obtain . The resulting  The resulting root locus is:



The time domain simulation results in: %OS=5.22%, 

>> syms s

>> s=tf('s');

>> P=0.63/(1+2\*0.18\*s/305.4+s^2/305.4^2)/(1+2\*0.02\*s/248.2+s^2/248.2^2);

>> G=0.0679\*(s^2+12\*s+6.25e4)/(s+20)/(s+70);

>> T=G\*P/(1+G\*P);

>> step(T)



**f., g.**

The PI compensator is designed by adding a pole at the origin and arbitrarily placing a zero at the same point as that of the rightmost compensator pole. The resulting  where the gain was adjusted to maintain the damping factor. The resulting simulation gives %OS=4.3% and 

>> syms s

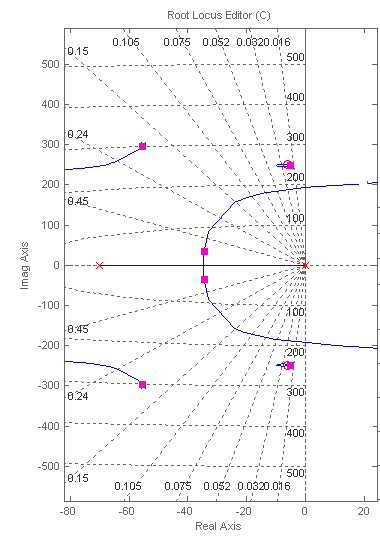
>> s=tf('s');

>> P=0.63/(1+2\*0.18\*s/305.4+s^2/305.4^2)/(1+2\*0.02\*s/248.2+s^2/248.2^2);

>> G=0.0586\*(s^2+12\*s+6.25e4)/s/(s+70);

>> T=G\*P/(1+G\*P);

>> step(T,0.15)





**39.**

1. As can be seen from the entries in the first column of Table P9.54 below, we need to improve the steady-state error of this system as well as its transient response.
   1. We design first the PD controller to meet transient response specifications. This includes the controller’s zero location and gain: .

We start by finding the desired operating point (dominant pole), corresponding to transient response specifications:

 corresponds to an overshoot of 4.321%, which ≤4.4%.

The peak time of the PID-controlled system, *Tp*, should be less than that of the uncompensated system by ~20%; e.g.; *Tp* **≈** 0.8 x 0.0207 = 0.01656



The natural frequency should be: .

Hence, the desired operating point is: .



**Design point:**

*-* **189.7 ± *j* 189.7**

***- Zc***

Next we design the PD controller. Using the geometry of the triangle shown in red in Figure above we calculate the compensating zero’s location.

To use the root locus utility, we find the open-loop poles and zeros of the plant, e.g., the roots of the numerator and denominator of *Gp*(s), which are: – 12.5, – 606.3 ± *j* 2155.8, and – 612.5 ± *j* 1467.9, respectively. As the coordinates of the desired dominant poles were entered, the sum of angles from the uncompensated system’s poles and zeros to the desired compensated dominant pole was found to be – 217.84o. Thus, the contribution required from the PD compensator zero is – 180 o + 217.84o = 37.84o. Hence:  and the poles of the systems are determined by the equation  where:



is the open-loop transfer function.

1. The following MATLAB M-file was written to simulate the system and plot its step response (shown below) to check whether all requirements may be met using a PD controller.

numg = 125\*[1 1225 2.53E6];

deng = [1 1225 503E4 625E5];

Gp = tf(numg, deng); %Gp is the transfer function (TF) of the plant

numh = 200;

denh = [1 200];

H = tf(numh, denh); %H is the TF of the feedback low-pass filter

Zc = 433.9;

numgc = [1 Zc];

dengc = 1;

Gc = tf(numgc, dengc); %Gc is the TF of the PD Controller

rlocus(Gp\*Gc\*H);

axis ([-250, 0, -250, 250]);

z=0.707;

sgrid(z,0)

title('Root Locus Zoomed-in around Dominant Poles with a 0.707 Damping Line')

[K1,p]=rlocfind(Gp\*Gc\*H);

pause

K = K1;

T=feedback((K\*Gc\*Gp),H); %T is the closed-loop TF of the system

step(750\*T);

axis ([0, 0.04, 0, 1300]);

grid

title(['Step Response of PD-cont. Bus Voltage for Zc = - ', num2str(Zc)]);



1. As could be seen from the above graph, the settling time, *Ts*, has been reduced significantly, but the overshoot is extremely high. Therefore, other values of the PD controller’s zero were tested to get a satisfactory transient response or, at least, the best possible one, with the idea that the transient may be improved further by adjusting the zero that will be added later in the following step (associated with the PI controller), since reducing the steady-state error to zero, requires the addition of an integral mode.

The two graphs obtained for the PD controller’s zero at *Zc* = *-* 800 and at *–* 2000 are shown below. It could be seen from these graphs that the response of the PD-controlled system with *Zc* = *-* 800 is faster and its peak time, *Tp* = 0.0142 sec, satisfies that requirement.

Therefore, the results obtained for *Zc* = *-* 800 were added to Table P9.54 below as entries, which characterize the PD-compensated system.





1. Adding the PI controller will introduce a pole at the origin, which will have a negative effect on the transient response. To minimize that effect we place a zero(), *ZPI*= *K*2/*K*1, close to that pole. The open-loop transfer function of the system now becomes:

.

The following MATLAB M-file was written to add the integral mode. It was run a few times with various values of the PI controller’s zero, *ZPI*, (between zero and the closest open-loop pole at -12.5) to check whether all requirements are met using a PID controller.

numg = 125\*[1 1225 2.53E6];

deng = [1 1225 503E4 625E5];

Gp = tf(numg, deng); %Gp is the transfer function (TF) of the plant

numh = 200;

denh = [1 200];

H = tf(numh, denh); %H is the TF of the feedback low-pass filter

Zc = 800;

Zi = 4;

numgc = poly ([-Zi, -Zc]);

dengc = [1 0];

Gc = tf(numgc, dengc); %Gc is the TF of the PID Controller

rlocus(Gp\*Gc\*H);

axis ([-200, 0, -200, 200]);

z=0.707;

sgrid (z,0)

title('Root Locus Zoomed-in around Dominant Poles with a 0.707 Damping Line')

[K1,p]=rlocfind(Gp\*Gc\*H);

pause

K = K1;

T=feedback((K\*Gc\*Gp),H); %T is the closed-loop TF of the system

step(750\*T);

axis ([0, 0.4, 0, 850]);

grid

title(['Step Response of PID-cont. of DC Bus for Zc = -',num2str(Zc),' and Zi = - ',num2str(Zi),]);



* 1. As a result, the following transfer function of the PID controller was found:

.

Thus, the gains, *K*1, *K*2, and *K*3 are: *K*3 = 0.003; *K*1 = 2.41; and *K*2 = 9.6.

1. The system simulation showed that all steady-state and transient response requirements have been met (see the figure below and Table P9.54).



1. All requirements have been met. No need for any redesign.

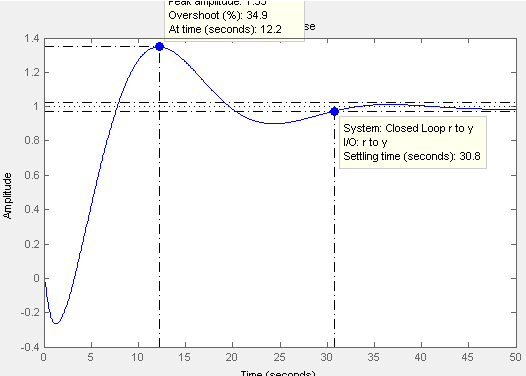
Table P9.54

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Uncompensated** | **PD-compensated** | **PID-compensated** |
| Plant & Compensator TF, *Gc*(s)*Gp*(s)*H*(s) |  |  |  |
| Dominant Poles | *-* 108.5 ± *j* 108.5 | *-* 130.6 ± *j* 130.6 | *-* 125 ± *j* 125 |
| Gain, *K* | 1.675 | 1.498 | 2.41 |
| Damping Coefficient, ζ | 0.707 | 0.707 | 0.707 |
| Natural Frequency, *ωn*; | 153.47 rad**/**sec | 184 rad**/**sec | 177 rad**/**sec |
| Percent Overshoot, *%OS* | 7.31% | 10.2% | 4.33% |
| Peak Time, *Tp* | 0.0207 sec | 0.0142 sec | 0.0139 sec |
| Settling Time, *Ts* | 0.0338 sec | 0.0279 sec | 0.387 sec |
| Static Error Constant, *Kp* | 8.494 | 12.549 | ∞ |
| Steady-state Error, *eV*step(∞) | 10.53% | 7.38% | 0% |
| Other Poles | *-* 604.0 ± *j* 2160.2 | *-* 623.6 ± *j* 2166.0 | *-* 622.7 ± *j* 2165.5, *–* 3.8 |
| Zeros | *-* 612.5 ± *j* 1467.9 | *-* 200, *-* 612.5 ± *j* 1467.9, *-* 800 | *-* 4, *-* 200, *-* 612.5 ± *j* 1467.9, *-* 800 |
| Comments | Second-order Approximation OK | Zeros at *–* 200 & *-* 800 not cancelled | Zeros at *-* 4, *–* 200, and *-* 800 not cancelled |

**40.**

Uncompensated: Searching along the line for  the operating point is found at  when  . The expected  , the settling time  sec.  .

Compensated: The PI compensator is chosen as  . Searching along the line for  the operating point is found at  when  . The expected %OS is unchanged, the settling time sec.  . The system has a new closed loop pole at -0.00297 and several far away poles. A step response simulation for the compensated system is shown below:



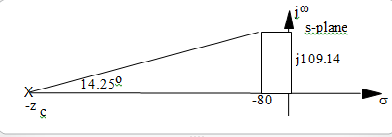
**41.**

For an overshoot of less than 10%, the damping ratio *ζ* should be > 0.5912.

Given that , the design point is: – 80 + j 109.1.

Assume a PI controller with Gc(s) = , to reduce the steady-state error to zero.

The total angular contribution of the system poles and zero plus the PI controller pole and zero at the design point is –194.25o. Hence, the ideal derivative controller must contribute 194.25o -180o = 14.25o. Using the geometry below, zc = 509.7.



The PID controller is, thus, given by the transfer function:  and the open-loop transfer function in a pole zero form with a unity gain is:

.

The MATLAB M-file is:

numgp = [1 2200];

numgc = poly([-509.7 -0.1]);

dengc = poly(0);

Gc = tf(numgc,dengc);

numg = conv(numgp, numgc);

dengp1 = [1 219.9];

dengp2 = [1 114.4 17653251.48];

dengp = conv(dengp1, dengp2);

Gp = tf (numgp, dengp);

deng = conv(dengp, dengc);

G = tf(numg, deng);

rlocus(G)

title('Full Root Locus for PID-Controlled Grid Connected VSC')

pause

pos=10;

z=-log(pos/100)/sqrt(pi^2+log(pos/100)^2);

Gzpk=zpk(G);

axis ([-3600, -3200, 4400, 5000]);

sgrid(z,0)

title('Root Locus Zoomed-in for Dominant Poles at a 0.5912 Damping Line')

[K1,p]=rlocfind(G);

pause

K = K1;

T=feedback(K\*G,1); %T is the closed-loop TF of the system

T=minreal(T);

step(208\*T);

grid

ylabel('Output Voltage of the VSC, Va(t), in volts')

title('Output Response for a Grid Connected VSC')

p

Kv = K1\*509.7\*0.1\*2200/(219.9\*17653251.48)

The full root-locus obtained is shown below. The close-up of that locus (from -3600 to -3200 on the real axis and from 4400 to 5000 on the imaginary axis) follows. From that close-up, *K* and the coordinates of all poles corresponding to *ζ* = 0.5912 were found to be:

selected\_point = -3.4412e+03 + 4.6953e+03i

p = 1.0e+03 \*

-3.4439 + 4.7064i

-3.4439 - 4.7064i

-0.3415 + 0.0000i

-0.0001 + 0.0000i

K = 6.8950e+03

Kv = 0.1992

The corresponding closed-loop transfer function is:

T =

6895 s^3 + 1.868e07 s^2 + 7.733e09 s + 7.732e08

-----------------------------------------------------

s^4 + 7229 s^3 + 3.636e07 s^2 + 1.162e10 s + 7.732e08

Continuous-time transfer function.

The figure, shown below the close-up, illustrates the step-response obtainedat that value of the gain (with some important characteristics marked on it) when a step input, *r(t)=* 208 *u*(t), volts, was applied at t = 0.

Looking at the above values of the poles and the time response obtained, we note the following:

* Although a damping ratio *ζ* = 0.5912 was used to design the PID controller, the real poles turned out to be more dominant;
* The output voltage rises sharply to ~150 volts and then grows gradually (without any noticeable oscillations) to reach 204 volts (e.g. within 2% of the final value) in a settling time of ~ 42 seconds.



****

****

**42.**

The transfer function, *Gp* (s) is given by:



A 15% overshoot corresponds to a damping ratio:

.

From: , we get:  and 

Thus, the design point is: – 20 + *j* 33.1.

The poles and zeros of *Gp* (s) are: *p1* = – 0.5; *p2, 3* = – 3.8 ± *j* 249 and *Z*1, 2 = – 0.6 ± *j* 111.8.

Assume a PI controller with Gc(s) = , to reduce the steady-state position error to zero.

The total angular contribution of the system poles and zeros plus the PI controller pole and zero at the design point is –125.85o.

The ideal derivative controller must contribute, therefore, 180o – 125.85o = 54.15o. Using the geometry below:

.

The PID controller is, thus, given by the transfer function:

.

s-plane



*j*33.1

– 20

– z

c

54.15

o

X

*j*

The following MATLAB M-file was written to plot system’s full root locus (shown below); zoom it in to find the controller gain that best satisfies the requirements; and plot the feedback system’s output response. That figure is shown below with all of the important characteristics of the output response marked on it.

clc;

clf;

num1=250;

num2=[1 1.2 12500];

numgp = conv(num1, num2);

dengp=[1 8.1 62003 31250];

Gp=tf(numgp, dengp);

numgc = poly([-43.9 -0.1]);

dengc = poly(0);

Gc = tf(numgc,dengc);

numg = conv(numgp, numgc);

deng = conv(dengp, dengc);

G = tf(numg, deng);

rlocus(G)

title('Full Root Locus for PID-Controlled Coupled Drive')

pause

pos=15;

z=-log(pos/100)/sqrt(pi^2+log(pos/100)^2);

Gzpk=zpk(G);

axis ([-50, 0, -10, 150]);

sgrid(z,0)

title('Root Locus Zoomed-in for Dominant Poles at a 0.517 Damping Line')

[K1,p]=rlocfind(G);

pause

K = K1;

T=feedback(K\*G,1); %T is the closed-loop TF of the system

T=minreal(T);

step(260\*T);

grid

ylabel('Shaft Angular Speed, Omega\_L, in rad/sec')

title('Output Response for the Coupled Drive')

K

p

Kv = K1\*6.04\*0.1\*250\*12500/31250





The selected\_point = - 43.1872 + 0.1863i, corresponds to: K = 1.0311 and the following closed-loop poles:

p =

1.0e+02 \*

-0.0090 + 1.1254i

-0.0090 - 1.1254i

-0.4316 + 0.0000i

-0.0010 + 0.0000i

The velocity error constant was found to be Kv = 62.2769. Thus the velocity error is:

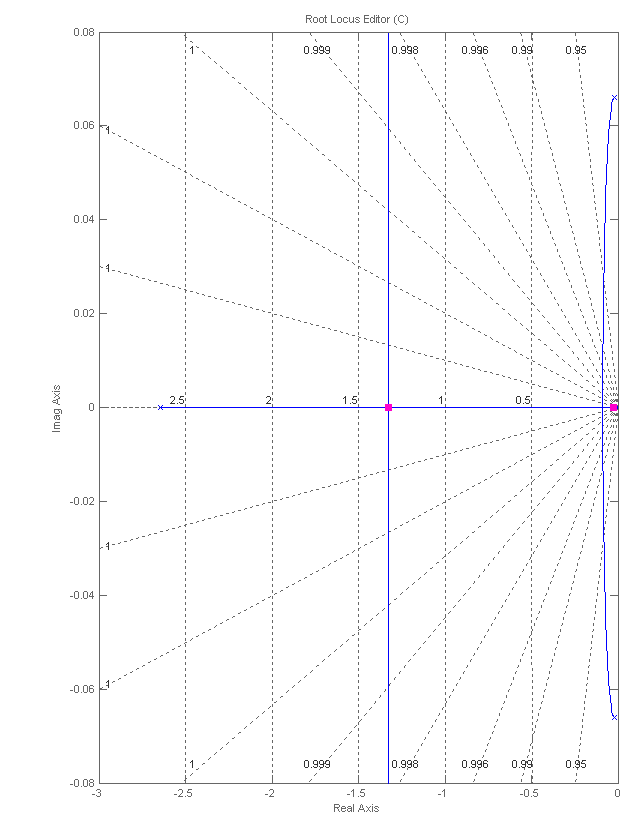


As could be seen from the results given above and the graph of *L*(t) all of the design requirements are met or exceeded with the exception of the settling time, which is merely acceptable.

****

**43.**

a. The root locus for the uncompensated system was obtained in Chapter 8 . It was shown there that the system is closed loop stable for all K>0 . All the poles are real when .



The step response simulation:

>> syms s

>> s=tf('s');

>> P=(520\*s+10.3844)/(s^3+2.6817\*s^2+0.11\*s+0.0126);

>> Gc = 0.00331;

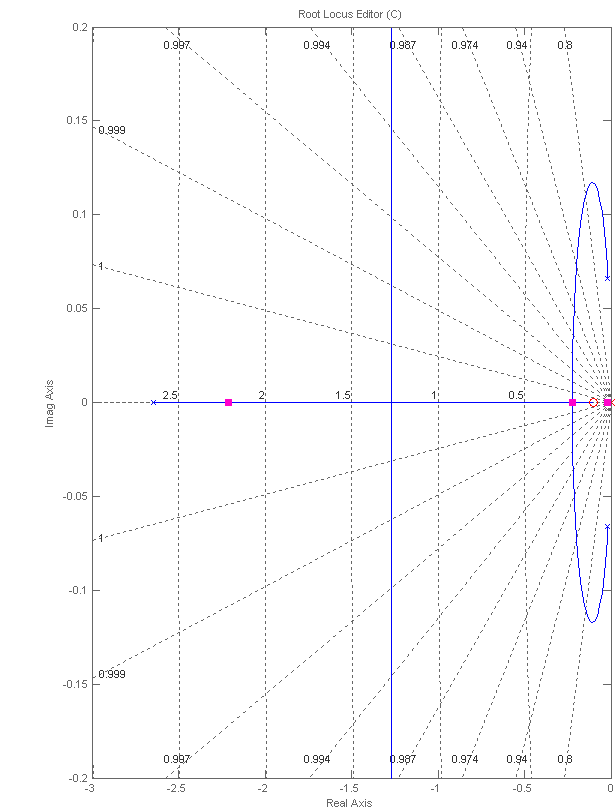
>> T=Gc\*P/(1+Gc\*P);

>> step(T)



The measured %OS=31% and 

**b.** A pole is placed at the origin and a zero is arbitrarily placed at -0.1, since the system has a pole very close to the origin (z=-0.02) , it may erve as the compensator’s zero. s The resulting compensator is  . The root locus is shown next:



The step response simulation:

>> syms s

>> s=tf('s');

>> P=(520\*s+10.3844)/(s^3+2.6817\*s^2+0.11\*s+0.0126);

>> Gc=0.00189\*(s+0.1)/s;

>> T=Gc\*P/(1+Gc\*P);

>> step(T,100)



%Os=7% and 

**44.**

1. To determine the improvements required in the transient and steady-state responses of the uncompensated system, its performance characteristics, obtained in part (a) of problem 8.72 are entered into Table P9.57 which is given at the end of part (b) below. To plot *c(t)* for *r(t)* = 4 *u(t)*, volts, the MATLAB M-file developed for part (a) of problem 8.72 was modified as shown below:

numg = [1 0.6];

deng = poly ([-0.0163 -0.5858]);

G = tf(numg, deng);

rlocus(G);

axis ([-0.8, 0.1, -0.1, 0.1]);

pos=(0);

z=-log(pos/100)/sqrt(pi^2+(log(pos/100))^2);

sgrid(z,0)

title(['Root Locus with ', num2str(pos) , ' Percent Overshoot Line'])

[K1,p]=rlocfind(G);

pause

T=feedback(K1\*G,1); %Tu is the closed-loop TF of the uncompensated system

T=minreal(T);

step(4\*T);

xlabel ('Time')

ylabel ('Speed Sensor Output, c(t) in volts')

title (['Step Response of Uncompensated System at K = ', num2str(K1), ' to a 4 volt Input Step'])

grid

The step response, c (t), of the uncompensated system to *r(t)* = 4 *u(t)* is shown below. As could be seen from the plot and the entries made into Table P9.57 below, the uncompensated system does not meet any of the steady-state requirements – the need to reduce *e*step[[1]](#footnote-2) (∞) from 2% to the required 0% and to have a finite value for the steady-state error due a ramp input, *e*ramp (∞) = 2%, indicate that the system should become type-1, e.g. an integral mode should be added to the controller. The speed of response also is not adequate – the rise time is 2.69 seconds and the settling time is 4.69 seconds whereas the compensated system is required to have a settling time equal to or less than four seconds.

****

1. We now design the PI-controller to obtain a steady-state error, *e*step (∞) = 0, for a step input, *r(t)* = 4 *u(t)* and a steady-state error for a unit-ramp input, *e*ramp(∞) ≤ 2%.

To achieve that and at the same time either improve or have a minor negative effect on the transient response of the system, we need to place the PI-controller’s zero on-top or close to the dominant pole of the uncompensated system, located at – 0.0163. Hence, we’ll start by assuming that the transfer function of the PI-controller is: .

Thus, we obtain a type-1 system with a transfer function[[2]](#footnote-3) of the “Plant & Compensator” given by: , which has a “position” error constant  and a “velocity” error constant, .

Hence, the requirement to have a zero steady-state error for a step input, *r(t)* = *A* *u(t)* = 4 *u(t)* is satisfied, since: .

The requirement that *e*ramp(∞) ≤ 2%, for a unit-ramp input will be satisfied if , e.g., if . At this value of *K1*, the open-loop transfer function of the PI-compensated system is: .

The following MATLAB file was written to check whether all requirements have been met:

numg = [48.83 29.3];

deng = poly ([0 -0.5858]);

G = tf(numg, deng);

T = feedback(G,1); %T is the closed-loop TF of the PI\_compensated system

T = minreal(T);

step(4\*T);

axis ([0, 0.2, 0, 5]);

grid

xlabel ('Time')

ylabel ('Speed Sensor Output, c(t) in volts')

title ('PI-compensated Systems Response to a 4 volt Input Step')

pause

numr = 1;

denr = poly (0);

R = tf(numr, denr);

T2 = R\*T;

E= R-T2; %E = Ramp error of PI\_compensated system

step(T2,'b', R,'g', E,'r');

axis ([0, 5, 0, 5]);

grid

xlabel ('Time')

ylabel ('Output Rate, dc(t)/dt, V/s (Blue); Unit Ramp (Green); Error (Red)');

title ('Response of PI-compensated System to a Unit-ramp Input')

The step response, c(t), and the unit-ramp response, dc(t)/dt, of the PI-compensated system are shown below. As could be seen from the plots and the entries made into Table P.9.Z\*, the PI-compensated system satisfies all performance requirements. Its steady-state error for a step input, *r(t)* = 4 *u(t)*, is zero and the value of its steady-state error due a unit-ramp input, .

It has also an extremely fast speed of response – the rise time is 0.045 seconds and the settling time is 0.08 seconds. Since the PI-compensated system satisfies all performance requirements, there is no need to add a derivative mode. Therefore, Table P.9.Z\*, shown below has been limited to entries for the uncompensated and PI-compensated systems only.

It should be noted, however, that the rise and settling times obtained seem to be unrealistic for the reasons noted above. Therefore, the Simulink model (which was developed originally for problem 5.81) was modified as shown above and will be run in step (c).





**Table P9.57 \* Characteristics of Uncompensated and PI-compensated Systems**

Table P9.57

|  |  |  |
| --- | --- | --- |
|  | **Uncompensated** | **PI-compensated** |
| Plant & Compensator, G(s) |  |  |
| Closed-loop Transfer Function, T (s) |  |  |
| Dominant Pole(s) | – 0.691, – 0.691 | – 48.82 |
| Proportional Gain, *K* | *K* = *K*1 = 0.78 | *K*1 = 48.83 |
| Damping Coefficient, ζ | 1 | 1 |
| Rise Time, *Tr* | 2.69 sec | 0.045 sec |
| Settling Time, *Ts* | 4.69 sec | 0.08 sec |
| *e*step (∞) | 2% | 0 |
| *e*ramp (∞) | **∞** | 2% |
| Other Poles | none | – 0.6002 |
| Zeros | – 0.6 | – 0.6 |
| Comments | Second-order Critically-damped System | Second-order[[3]](#footnote-4) Over-damped System |

* 1. We now run the following Simulink model, which includes a saturation element placed at the output of the motor armature, which was set to an upper limit of 250 A.

****

As a result of that simulation, the following figures were obtained:

* 1. Change in Motor Armature Current in Response to a Speed Change Command;
  2. Change in Car Acceleration in Response to a Speed Change Command;
  3. Change in Car Speed in Response to a 4-V Reference Input Step.

Based on these graphs, the following observations may be made:

1. The armature current rises very fast to 250 amps; sustains at that level for 0.7 seconds,

****

****

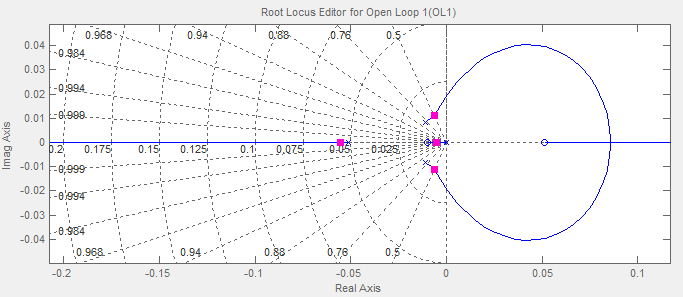
then drops down exponentially (in about 4 seconds) to 7.0 amps ± 5%.

1. Car acceleration also rises very fast to 3.833 m/s2, drops down slightly (to 3.75 m/s2) in the first 0.7 seconds, then drops down exponentially (in about 4.3 seconds) to 0.
2. With a saturation element (set to a limit of 250 A) placed at the output of the motor armature, the step response obtained for the change in car speed exhibits realistic values of the rise and settling times (1.75 and 2.8 seconds, respectively), whereas the steady-state error in car speed remains equal to zero (the final car speed = 5.556 m/sec) and the steady-state error in car acceleration remains below 2% ().

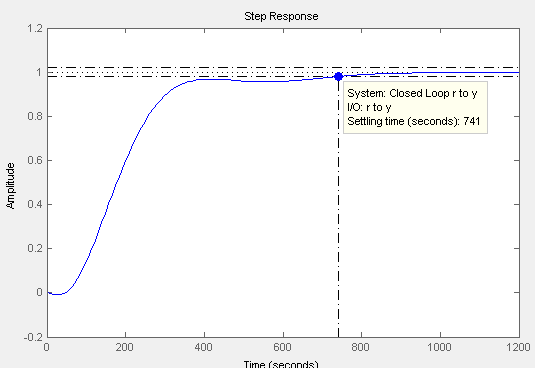
****

**45.**

1. Using the results of Problem 81 in Chapter 8, we search for the intersection with the  line, found when K=0.83. The open loop transfer function is , Substituting we calculate. For a step input  .
2. A PI compensator design is performed by arbitrarily adding a pole at the origin, and a zero to its left. We try  . We use this compensator, and look for the value of K that results in  as shown below:



The resulting gain is K=0.7. The unit step response of the system is shown next:



1.  [↑](#footnote-ref-2)
2. Enter this transfer function and the results obtained for the PI-compensated system into Table P9.57. [↑](#footnote-ref-3)
3. With the second closed-loop pole very small (compared to the dominant pole) and almost equal to the system’s zero, this system acts almost as a first-order lag. [↑](#footnote-ref-4)