CHAPTER 4

Analysis and Design of Digital Control Systems

The analysis and design of sampled-data control systems is similar in principal to the design of continuous-data control systems. The design objective is basically that of determining the controller so that the system will perform in accordance with specifications. In this chapter we shall investigate the important topics of stability, steady-state error, root-locus, and frequency response as applied to the sampled-data system.

Stability Analysis in the z-plane

A linear continuous feedback control system is stable if all poles of the closed-loop transfer function lie in the left half of the s-plane. Consider the closed-loop transfer function $T(s)$, which is transformed into a sampled-data transfer function $T(z)$. The z-plane is related to the s-plane by the transformation

$$\begin{aligned} z = e^{sT} &= e^{(\sigma + j\omega)T} = e^{\sigma T} e^{j\omega T} \\ &= e^{\sigma T}(\cos\omega T + j\sin\omega T) \\ &= e^{\sigma T}\angle\omega T \end{aligned} \tag{4.1}$$

$$|z| = e^{\sigma T} \tag{4.2}$$

From (1.3) it is clear that points on the $j\omega$-axis, i.e., $s = 0 + j\omega$, maps into points $z = 1\angle\omega T$, namely a unit circle in the z-plane as shown in Figure 4.1

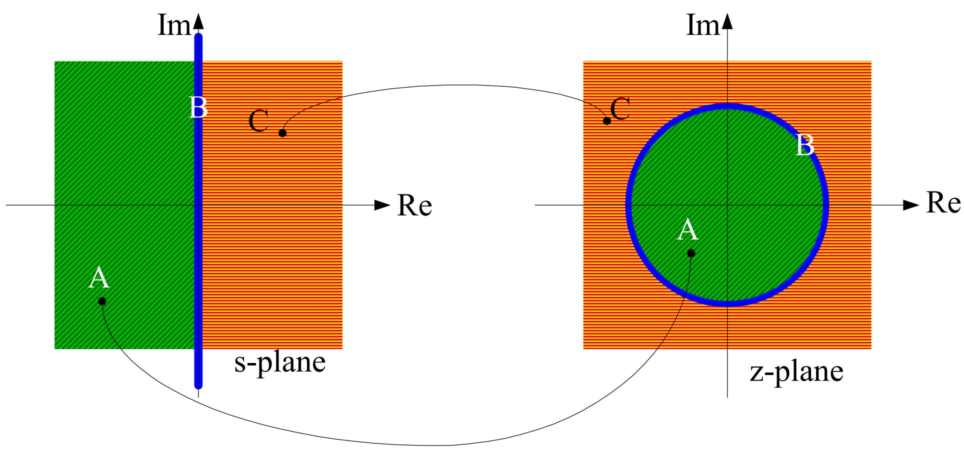


Figure 4.1 Mapping regions of the s-plane onto the z-plane

Points with negative values of σ in the s-plane will map into points $|z|<1$ in the z-plane, i.e., inside the unit circle. Points with positive values of σ will map into points $z>1$, namely outside the unit circuit. Finally point at origin $s=0$ maps into the point $z=1$ in the z-plane. Thus for closed-loop sampled-data system stability we have the following criterion:

- The sample-data system is stable if roots of the z-domain characteristic equation lie inside the unit circle in the z-plane.
- The sample-data system is marginally stable if roots of the z-domain characteristic equation lie on the unit circle in the z-plane.
- The sample-data system is unstable if roots of the z-domain characteristic equation lie outside the unit circle in the z-plane.

The relation between s-domain poles and z-domain poles and the corresponding step response is illustrated in Figure 4.2.

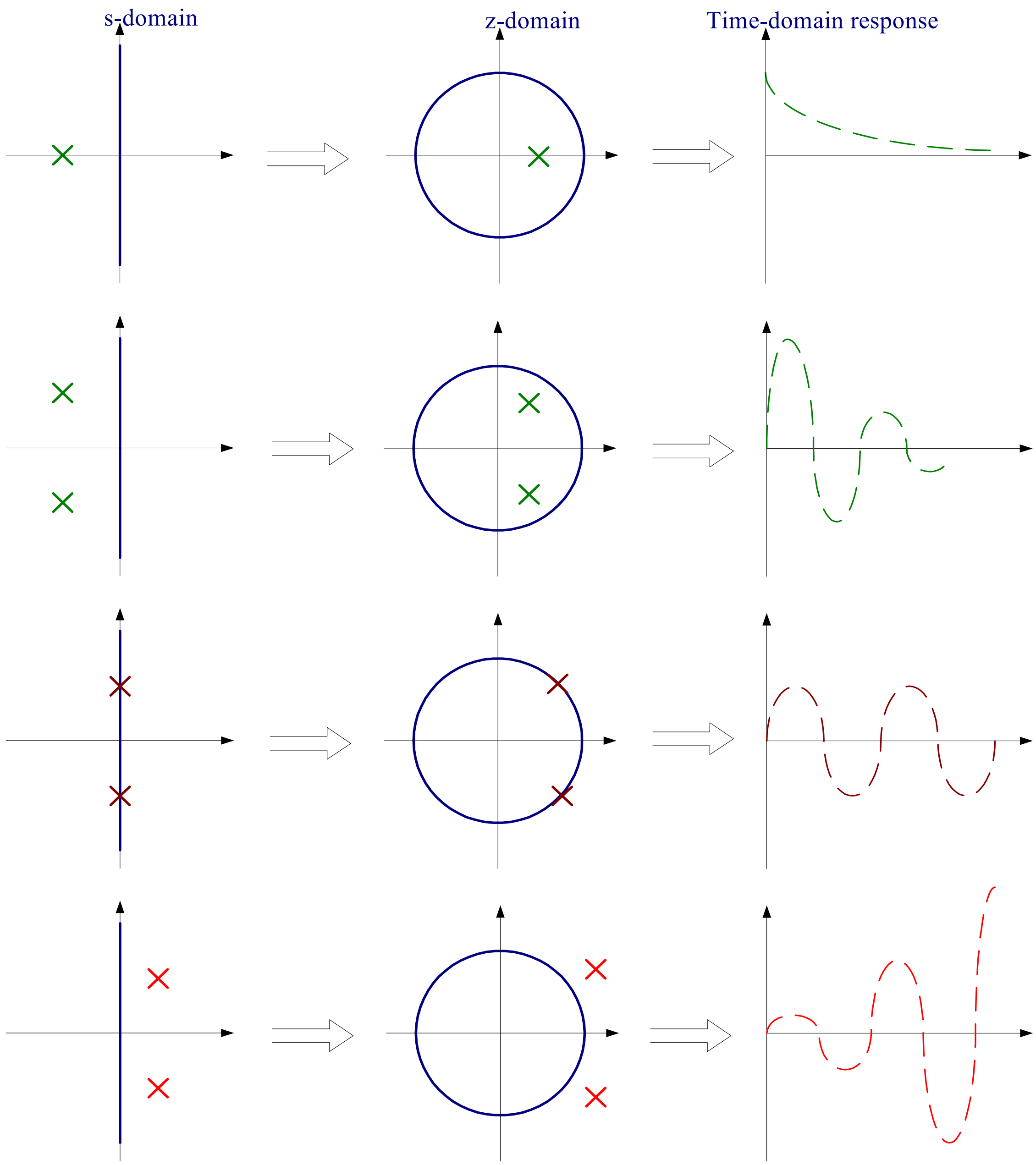


Figure 4.2 Examples relating pole locations to time response.

The important factor determining the stability in sampled-data system is the effect of sampling rate on the transient response. Changes in sampling rate not only change the nature of the response from overdamped to underdamped but also can turn a stable system into an unstable one.

Example 4.1

For the sampled data system shown determine the system stability for $K = 10$ and $K = 80$ $a = 25$ and sampling rate is $T = 0.1$ second.

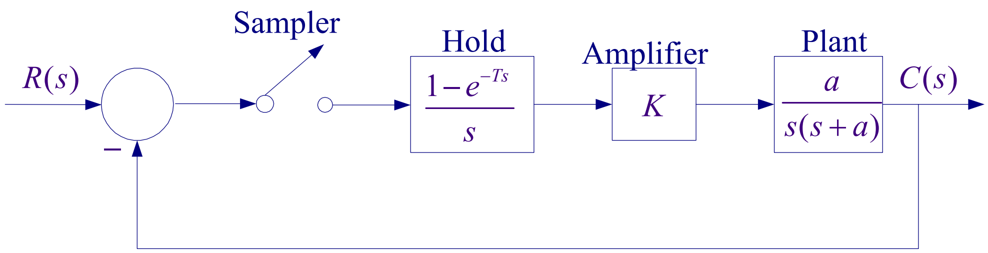


Figure 4.3 A closed-loop sampled system

$$G(s) = \frac{1-e^{-Ts}}{s}\frac{K}{s(s+a)}$$

$$G(z) = (1-z^{-1})\mathcal{Z}\left\{\frac{Ka}{s^2(s+a)}\right\} = (1-z^{-1})\mathcal{Z}\{G_2(s)\} = \frac{z-1}{z}\mathcal{Z}\{G_2(s)\}$$

Performing partial fraction expansion

$$K\frac{a}{s^2(s+a)} = K\left[\frac{1}{s^2}+\frac{-1/a}{s}+\frac{1/a}{s+a}\right]$$

$$\mathcal{Z}\{G_2(s)\} = K\left[\frac{Tz}{(z-1)^2}-\frac{z/a}{z-1}+\frac{z/a}{z-e^{-aT}}\right]$$

$$= K\left[\frac{Tz}{(z-1)^2}-\frac{(1-e^{-aT})z/a}{z-1(z-e^{-aT})}\right]$$

Therefore

$$G(z) = K\left[\frac{T}{(z-1)}-\frac{(1-e^{-aT})/a}{(z-e^{-aT})}\right] = K\left[\frac{T(z-e^{-aT})-(z-1)(1-e^{-aT})/a}{(z-1)(z-e^{-aT})}\right]$$

Substituting for a and T, the result is

$$G(z) = \frac{K(0.0633z+0.0285)}{(z-1)(z-0.082)} = \frac{K(0.0633z+0.0285)}{z^2-1.08z+0.082}$$

The closed loop response is

$$T(z) = \frac{G(z)}{1+G(z)} = \frac{K(0.0633z+0.0285)}{z^2+(0.0633K-1.082)z+(0.0285K+0.082)}$$

(a) For $K = 10$, the characteristic equation is $z^2 - 0.45z + 0.367$ and the roots are $0.2246 \pm j0.5628$, therefore $|z| = 0.606$, i.e., the roots are inside the unit circle and the system is stable.
(b) For $K = 80$, the characteristic equation is $z^2 + 3.98z + 2.367$, and the roots are -3.25, -0.0726. Since $|z| > 1$, we have a root outside the unit circle and system is unstable.

Example 4.2 (chd4ex2.m)

For the system in Example 4.1, use MATLAB to find the value of K for marginal stability.

We use the following commands:

```
num = 25;
den = [1  25  0];
Gp = tf(num, den)                              % Plant transfer function
T = 0.1;                                       %  Sampling time
[numz, denz] = c2dm(num, den, T, 'zoh');       % open-loop sampled-data num and den
Gz = tf(numz, denz, 0.1)                       % Open-loop sampled-data transfer function

for K = 1: 0.01: 40
   Tz = feedback(K*Gz, 1);                     % Closed-loop sampled data transfer function
    [numT, denT] = tfdata(Tz, 'v');  % Returns the num and den of the closed-loop SDTF
    r = roots(denT);                           % Roots of the sampled-data char. Eq.
    rmag = max(abs(r));                        % Absolute value of the largest roots
    if rmag >= 1;              % If magnitude of the largest root >= 1 beark the loop
      break
    end
end

K                                              %  Value of K for marginal stability
disp(['      Roots        Magnitude'])
disp([r  abs(r)])
```

The result is

```
Transfer function:
          25
        ----------
        s^2 + 25 s


Transfer function:
        0.06328 z + 0.02851
        -----------------------
        z^2 - 1.082 z + 0.08208
```

Sampling time: 0.1

K =

32.2000

Roots	Magnitude
-0.4778 + 0.8785i	1.0000
-0.4778 - 0.8785i	1.0000

The value of K which causes the poles of the closed-loop sampled-data transfer function to lie on the unit circle is 32.2. This results in a system with marginal stability. You should try to change the sampling time to see its effect on the system stability.

For continuous-time system we used Routh array to find the number of roots in the right-half s-plane or establish the range of K for marginal stability. The Routh array cannot be applied directly to the characteristic equation in z-domain. Stability testing for a discrete-time system involves determining whether all the poles of the system's z-transfer function are within the unit circle on the z-plane. A method analogous to Routh-Hurwitz array is the Jury' Test.

Bilinear transformation

A discrete-time system is stable if and only if all poles of its z-transfer function are within the unit circuit on the complex plane. The bilinear transformation

$$z = \frac{2}{T}\left(\frac{1+W}{1-W}\right) \tag{4.3}$$

maps the unit circle to the left half-plane on the complex plane by making a change of the variable from z to W, and so is very useful in relating discrete-time situation to equivalent continuous-time ones. If T is assumed to be 1, the bilinear transformation becomes

The Routh-Hurwitz array can now be applied to the resulting characteristic polynomial expressed in terms of W to determine stability.

$$z = \frac{1+W}{1-W} \tag{4.3b}$$

Example 4.3

For the sampled-data transfer function given in Example 4.2 make the bilinear transformation and find the characteristic equation in terms of W. Apply Routh array and determine range of K for stability.

The open-loop sampled data transfer function is

$$G(z) = \frac{K(0.06328z + 0.0285)}{z^2 - 1.0821z + 0.0821}$$

Substituting for z from (4.3b), we get

$$G(W) = \frac{K(-0.0348W^2 - 0.0570W + 0.0918)}{2.1642W^2 + 1.8358W}$$

The characteristic equation is

$$1 + G(W) = 0 \quad \Rightarrow \quad 1 + \frac{K(-0.0348W^2 - 0.0570W + 0.0918)}{2.1642W^2 + 1.8358W} = 0$$

results in the polynomial

$$(2.1642 - 0.0348K)W^2 + (1.8358 - 0.057K)W + 0.918K = 0$$

Forming the Routh array, we have

$$\begin{array}{l|lllll}
W^2 & 2.1642 - 0.0348K & 0.918k & \Rightarrow & K < 2.1642/.0348 & \Rightarrow K < 62.2 \\
W^1 & 1.8358 - 0.057K & & \Rightarrow & K < 1.8358/0.057 & \Rightarrow K < 32.2 \\
W^0 & 0.0918K & & \Rightarrow & K > 0 &
\end{array}$$

Therefore, the system is stable if $K < 32.2$. This is the same result obtained in Example 4.2 sweeping K in a MATLAB loop. You must bear in mind that the range of K for stability is a function of the sampling time T. (See Example 13.9 in the textbook for another example of bilinear transformation and application of Routh array).

Range of T for stability

In Example 4.2 as the sampling time T is decreased, the value of K for marginal stability will increase. For example if sampling time is changed to $T = 0.01$, the gain for marginal stability becomes $K = 208.7$ In fact as T becomes infinitesimally small the value of K for stability approaches infinity. This is evident from the characteristic equation of the continuous-time system in Example 2.1, $s^2 + 25s + 25K = 0$ where system is stable or all $K < \infty$. Sapling time plays an important role in stability of the discrete-time systems.

Example 4.4

Determine the range of sampling interval T, for which the sampled-data system shown in Figure 4.4 is stable.

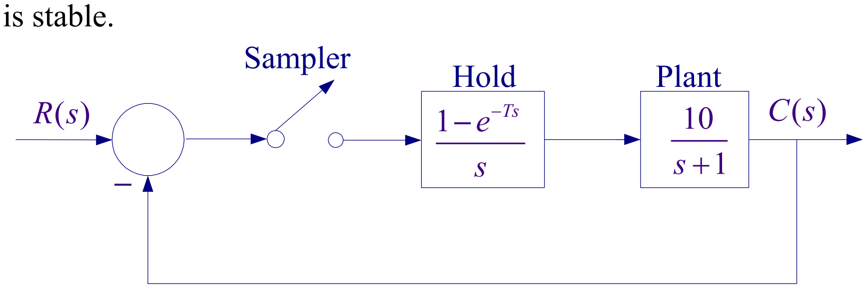


Figure 4.4 Digital system for Example 4.4.

$$G(s) = \frac{1-e^{-Ts}}{s}\frac{10}{s+1} = (1-e^{-Ts})\frac{10}{s(s+1)} = (1-e^{-Ts})\left[\frac{10}{s} + \frac{-10}{s+1}\right]$$

Taking the z-transform

$$(1-z^{-1})\mathcal{Z}\left[\frac{10}{s} + \frac{-10}{s+1}\right] = \frac{z-1}{z}\left[\frac{10z}{z-1} + \frac{-10z}{z-e^{-T}}\right] = 10\frac{1-e^{-T}}{z-e^{-T}}$$

The closed-loop transfer function is

$$T(z) = \frac{G(z)}{1+G(z)} = \frac{10(1-e^{-T})}{z-(11e^{-T}-10)}$$

System is stable if

$$11e^{-T} < 9$$

or

$$e^{T} < \frac{11}{9} \quad \text{or} \quad e^{T} < 1.2222 \quad \Rightarrow \quad 0 < T < 0.20$$

System is stable if $T < 0.2$ second or the sampling frequency $f > \frac{1}{0.2}$ or 5 Hz.

Transient Response

The standard form of the second-order continuous-time transfer function is given by

$$T(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.4}$$

where ω_n is the natural frequency and ζ is the damping ratio. The damping ratio gives us an idea about the nature of the transient response. It gives us a feel for the amount of overshoot and oscillation that the response undergoes. The transient response of a practical control system often exhibits damped oscillations before reaching steady state.

Roots of the characteristic equation $s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$ are

$$\begin{aligned} s_{1,2} &= -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \\ &= \sigma + j\omega \end{aligned} \tag{4.5}$$

The underdamped response ($\zeta < 1$) to a unit step input, subject to zero initial condition, is given by

$$c(t) = 1 - \frac{1}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_n t}\sin(\omega_n\sqrt{1-\zeta^2}\,t + \theta) \tag{4.6}$$

where $\theta = \tan^{-1}\frac{\sqrt{1-\zeta^2}}{\zeta}$

The performance criteria that are used to characterize the transient response to a unit step input include rise time t_r, peak time t_p, percent overshoot $P.O.$, and settling time t_s. To find the time corresponding to the peak value, the derivative of (4.5) is set to zero. This yields the peak time

$$t_p = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}} = \frac{\pi}{\omega} \tag{4.7}$$

where ω is the damped frequency of oscillation. The corresponding peak value is

$$c(t_p) = M_{pt} = 1 + e^{-\zeta\pi/\sqrt{1/\zeta^2}} \tag{4.8}$$

The percent overshoot is defined as

$$P.O. = \frac{\text{maximum value } - \text{ final value}}{\text{final value}} \times 100$$

Since the final value is 1, we have

$$P.O. = e^{-\zeta\pi/\sqrt{1-\zeta^2}} \times 100 \tag{4.9}$$

From (4.5) it is evident that the step response time constant is

$$\tau = \frac{1}{\zeta\omega_n} \tag{4.10}$$

Settling time is the time required for the response to settle within a small percent of its final value. Typically, this value may be assumed to be ± 2 percent of the final value. This is achieved approximately within four time constants, that is,

$$t_s \simeq 4\tau \tag{4.11}$$

The behavior of the step response can easily be predicated from the location of the s-domain poles.

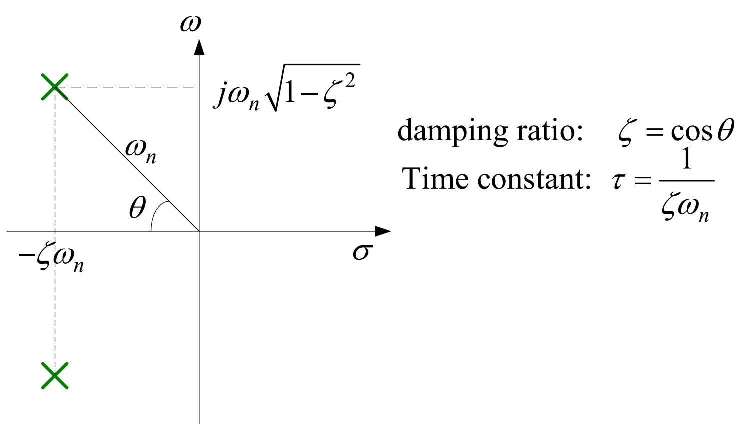


Figure 4.5 s-plane poles of a second-order continuous-time transfer function. The vertical lines on the s-plane are lines of constant τ or constant settling time.

$$s = \sigma_1 + j\omega = -\zeta\omega_n + j\omega = -\frac{1}{\tau} + j\omega$$

Substituting into $z = e^{sT}$

$$z = e^{\sigma_1 T} e^{j\omega} = r_1 e^{j\omega} \tag{4.12}$$

Therefore the vertical lines on the s-plane are mapped into circle of radius $r_1 = e^{\sigma_1 T}$. If σ is negative, the circle has a smaller radius than the unit circle. If σ is positive, i.e., vertical lines in the right-half s-plane are mapped into circles with radius larger than the unit circle as shown in Figure 4.6.

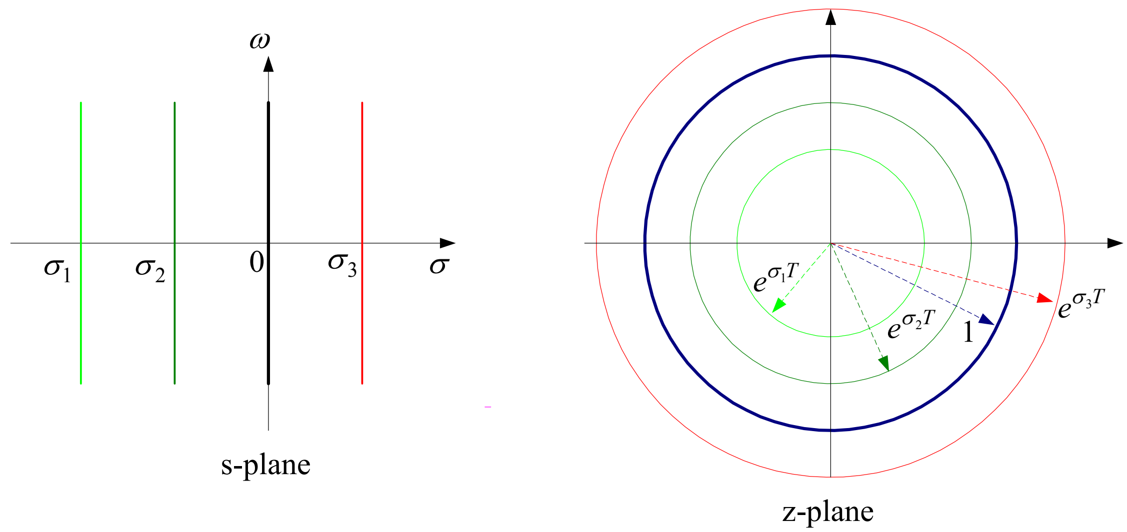


Figure 4.6 Mapping constant vertical lines (constant τ) into z-plane.

From (4.7) it is evident that horizontal lines in s-plane are lines of constant ω or constant peak time t_p. The horizontal lines are characterized by $s = \sigma + j\omega_1$, where ω_1 is constant. Substituting into $z = e^{sT}$, we have

$$z = e^{\sigma T} e^{j\omega_1 T} = e^{\sigma T} e^{j\theta_1} \tag{4.13}$$

Equation (4.13) represents radial lines at an angle of $\theta_1 = \omega_1 T$. If σ is negative, that section of the radial line lies inside the unit circle. If σ is positive, that section of the radial line lies outside the unit circle. Horizontal lines in s-domain are lines of constant frequency, which are mapped into radial lines in z-domain as shown in Figure 4.7.

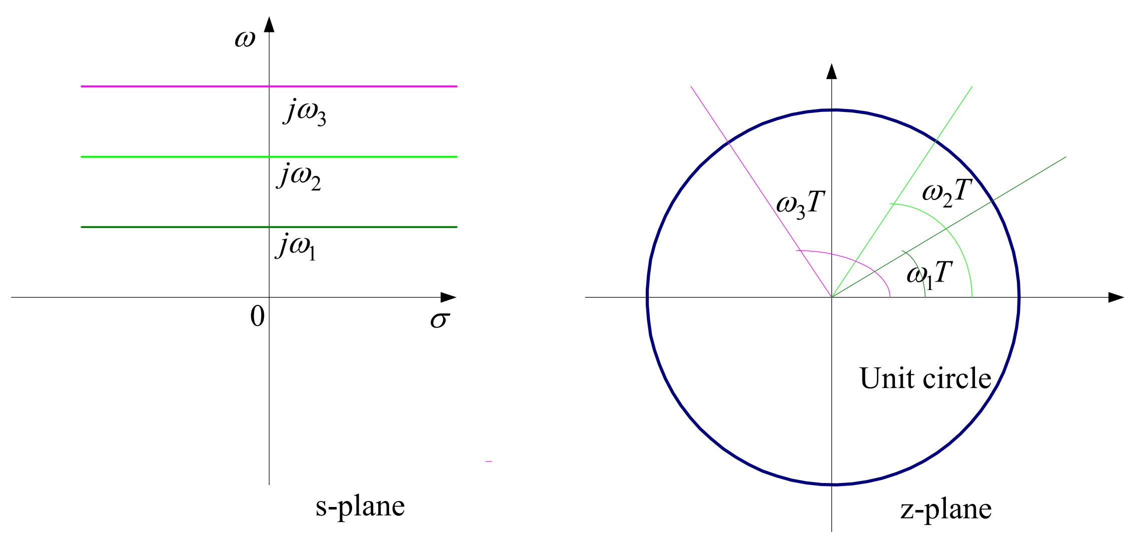


Figure 4.7 Mapping constant frequency lines into z-plane.

Finally, we map radial lines of constant damping ratio ($\zeta = \cos\theta$). From Figure 4.5, we have

$$\tan\theta = \frac{\omega}{\sigma} = \frac{\omega_n\sqrt{1-\zeta^2}}{-\zeta\omega_n} = -\frac{\sqrt{1-\zeta^2}}{\zeta}$$

Therefore

$$\sigma = -\frac{\zeta\omega}{\sqrt{1-\zeta^2}} \qquad \Rightarrow \qquad s = \sigma + j\omega = -\frac{\zeta\omega}{\sqrt{1-\zeta^2}} + j\omega$$

Since $z = e^{sT}$

$$z = e^{-\frac{\zeta}{\sqrt{1-\zeta^2}}\omega T} e^{j\omega T} = e^{-\frac{\zeta}{\sqrt{1-\zeta^2}}\omega T} \angle \omega T \qquad (4.14)$$

The radial lines of constant ζ are mapped into the z-plane in according to equation (4.14).

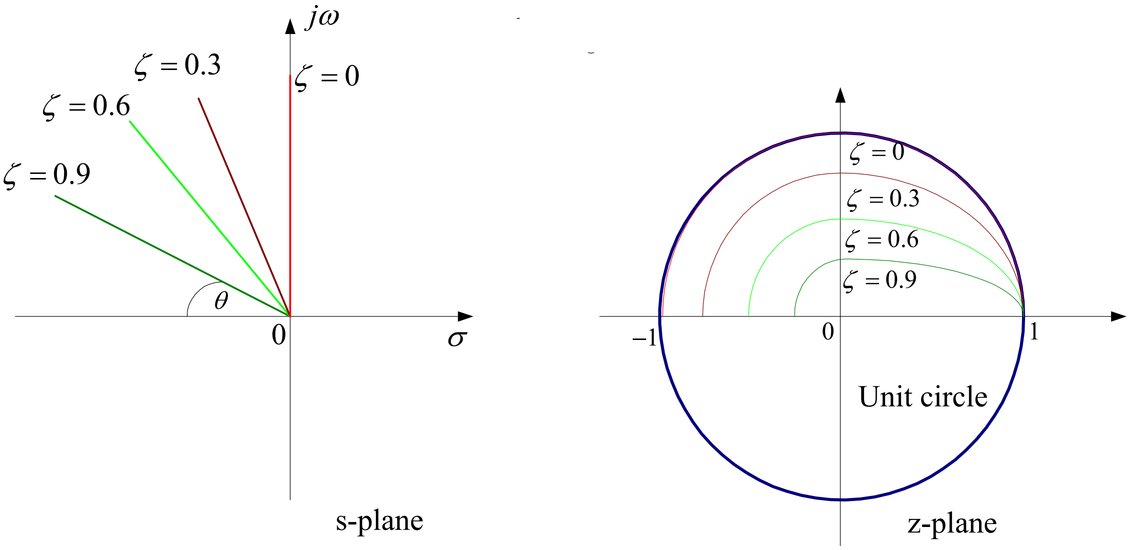


Figure 4.8 Mapping constant damping ratio into z-plane.

Thus, using the mapping illustrated in Figure 4.8-3.8, we can assign z-domain specifications similar to the time-domain specification. This will be illustrated through several examples after we consider the root-locus applied to the z-plane.

Steady-State Errors

We now examine the effect of sampling upon the steady-state error for digital systems. In digital system, the placement of the sampler changes the open-loop transfer function. The derivation of the steady-state error will be based on the typical placement of the sampler after the error. Consider the digital system in Figure 4.9(a), where the digital computer is represented by the sampler and zero-order hold. The transfer function of the plant is $G_p(s)$ and the transfer function of the z.o.h. by $(1-e^{sT})/s$. Letting $G(s)$ equal the product of the z.o.h and $G_p(s)$ and using the block diagram reduction technique for sampled-data system, we can find the sampled error $E^*(s) = E(z)$.

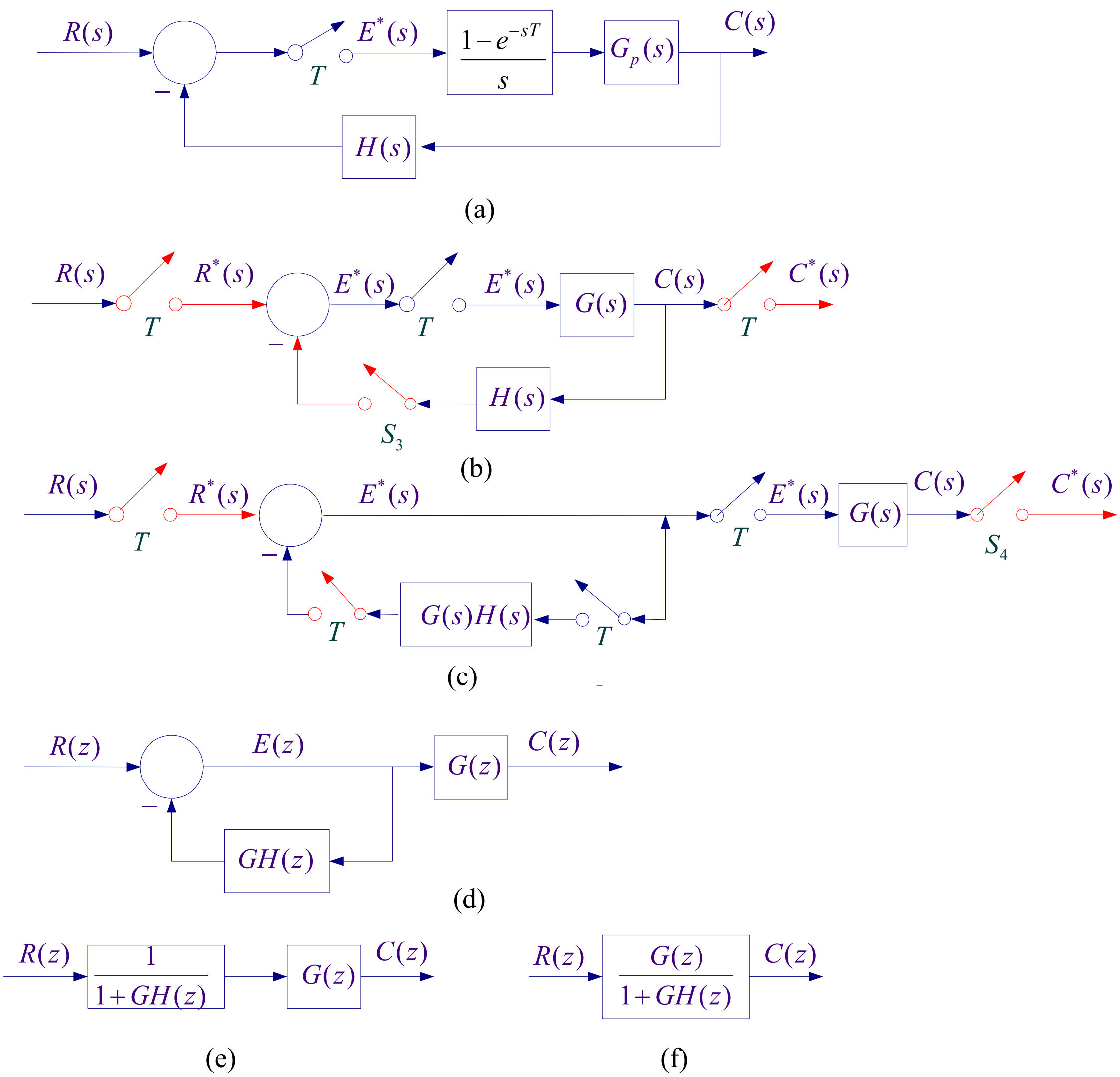


Figure 4.9 Digital feedback control system with z-domain equivalent block diagram.
From Figure 4.9(d)

$$E(z) = \frac{1}{G(z)}C(z) \quad \text{and} \quad C(z) = \frac{G(z)}{1+\overline{GH}(z)}R(z)$$

Therefore

$$E(z) = \frac{1}{\overline{GH}(z)}R(z) \tag{4.15}$$

From the final value theorem for discrete signals, we have

$$e^*(\infty) = \lim_{z\to 1}(1-z^{-1})E(z) \tag{4.16}$$

where $e^*(\infty)$ is the final sampled value of $e(t)$, or the final value $e(kT)$.

Thus,

$$e^*(\infty) = \lim_{z\to 1}(1-z^{-1})\frac{1}{1+\overline{GH}(z)}R(z) = \lim_{z\to 1}(\frac{z-1}{z})\frac{1}{1+\overline{GH}(z)}R(z) \tag{4.17}$$

We now consider three typical inputs, unit step, unit ramp, and unit parabolic inputs.

Unit Step

$$r(kT) = u(kT) \quad \Rightarrow \quad R(z) = \frac{z}{z-1}$$

Substituting in (4.17)

$$e^*(\infty) = \frac{1}{1 + \lim_{z\to 1} \overline{GH}(z)} \quad \text{or} \quad e^*(\infty) = \frac{1}{1+K_p} \quad \text{where} \quad K_p = \lim_{z\to 1} \overline{GH}(z) \quad (4.18)$$

Unit Ramp

Substituting in (4.17)

$$r(kT) = kTu(kT) \quad \Rightarrow \quad R(z) = \frac{Tz}{(z-1)^2}$$

$$e^*(\infty) = \frac{T}{\lim_{z\to 1}(z-1)\overline{GH}(z)} \quad \text{or} \quad e^*(\infty) = \frac{1}{K_v} \quad \text{where} \quad K_v = \frac{1}{T}\lim_{z\to 1}(z-1)\overline{GH}(z) \quad (4.19)$$

Unit Parabolic

Substituting in (4.17)

$$r(kT) = \frac{1}{2}(kT)^2 u(kT) \quad \Rightarrow \quad R(z) = \frac{T^2 z(z+1)}{2(z-1)^3}$$

$$e^*(\infty) = \frac{T}{\lim_{z\to 1}(z-1)^2\overline{GH}(z)} \quad \text{or} \quad e^*(\infty) = \frac{1}{K_a} \quad \text{where} \quad K_a = \frac{1}{T}\lim_{z\to 1}(z-1)^2\overline{GH}(z) \quad (4.20)$$

The equations developed for steady-state error are similar to the equations developed for the analog systems. Whereas multiple placements at the origin of the s-plane reduced steady-state error to zero in the analog case, we can see that multiple pole placements at $z=1$ reduce the steady-state error to zero. This conclusion makes sense when we consider that $s=0$ maps into $z=1$ under $z=e^{sT}$.

Example 4.5

In Figure 4.9(a), the plant transfer function is

$$G_p(s) = \frac{10}{s(s+1)} \quad \text{and} \quad H(s) = 1$$

Find the steady-state error for step, ramp, and parabolic inputs.

$$G(s) = \frac{1-e^{-Ts}}{s}\frac{10}{s(s+1)} = 10(1-e^{-Ts})\left[\frac{1}{s^2} + \frac{-1}{s} + \frac{1}{s+1}\right]$$

The z-transform is then

$$G(z) = 10(1-z^{-1})\left[\frac{Tz}{(z-1)^2} + \frac{-z}{z-1} + \frac{z}{z-e^{-T}}\right]$$

$$= 10\left[\frac{T}{z-1} - 1 + \frac{z-1}{z-e^{-T}}\right]$$

For a step input,

$$K_p = \lim_{z \to 1} G(z) = \infty \qquad \Rightarrow \qquad e^*(\infty) = \frac{1}{1+K_p} = 0$$

For a ramp input,

$$K_v = \frac{1}{T}\lim_{z \to 1}(z-1)G(z) = 10 \qquad \Rightarrow \qquad e^*(\infty) = \frac{1}{K_v} = 0.1$$

For a parabolic input,

$$K_a = \frac{1}{T}\lim_{z \to 1}(z-1)^2 G(z) = 0 \qquad \Rightarrow \qquad e^*(\infty) = \frac{1}{K_a} = \infty$$

Notice that the results obtained are the same as the results obtained for the analog system. Since stability depends on the sampling rate we must check for the stability for a given sampling time T.

Example 4.6 (chd4ex6.m)

For the plant transfer function in Example 4.5 write a MATLAB program to evaluate the steady-state error for step, ramp, and parabolic inputs. Also check for system stability by finding the roots of the sampled-data closed-loop poles if sampling time is $T = 0.1$ second.

We use the following commands

```
num = 10;
den = [1  1  0];
Gp = tf(num, den)                              % Plant transfer function
T = 0.1;                                       %  Sampling time
[numz, denz] = c2dm(num, den, T, 'zoh');    % open-loop sampled-data num, den
Gz = tf(numz, denz, 0.1)                 % Open-loop sampled-data transfer function
Kp = ddcgain(numz, denz)                    % Calculate Kp
e_ss_step = 1/(1+Kp)
numzKv = (1/T)*conv([1  -1], numz);      % Multiply G(z) by (1/T)*(z-1)
Kv = ddcgain(numzKv, denz)                 % Calculate Kv
e_ss_ramp = 1/Kv
numzKa = (1/T)^2*conv([1  -2  1], numz);    % Multiply G(z) by (1/T)^2*(z-1)^2
Ka = ddcgain(numzKa, denz)                 % Calculate Kv
if Ka == 0
   e_ss = inf
else, e_ss_par = 1/Ka
end

% To determine if system is stable we find the sampled data close-loop poles
```

```
Tz = feedback(Gz, 1);             % Closed-loop sampled data transfer function
[numT, denT]= tfdata(Tz, 'v');  %Returns the num, den of the closed-loop SDTF
r = roots(denT);                  % Roots of the sampled-data char. Eq.
rmag = abs(r)                     % Absolute value of the largest roots
```

The result is

```
Transfer function:
 0.04837 z + 0.04679
----------------------
z^2 - 1.905 z + 0.9048

Sampling time: 0.1
Kp =
        Inf
e_ss_step =
         0
Kv =
        10.0
e_ss_ramp =
        0.10
Ka =
        0
e_ss_par =  Inf

rmag =
       0.9755
       0.9755
```

Tee results are the same as the value found in Example 4.5. Also since the root magnitudes are less than 1, the digital control system with sampling rate 10 Hz is stable.

Root Locus

For the sampled-data system shown in Figure 4.10, the transfer function is

$$G(z) = \frac{KG(z)}{1 + K\overline{GH}(z)}$$

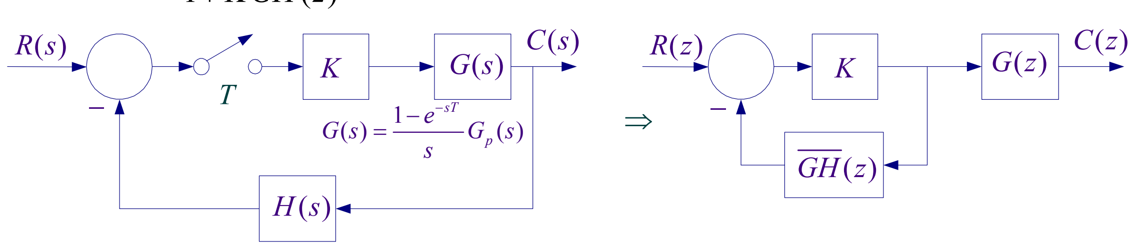


Figure 4.10 Sampled data system

The root locus is a plot of the locus of the roots of the characteristic equation $1+K\overline{GH}(z)=0$ in the z-plane. Thus the rules for the root locus construction are the same as those for continuous-time systems except for a change in variable from s to z. Although the rules for root locus constructions are the same, the region of stability on the z-plane is within the unit circle. This corresponds to the left-half s-plane in the s-domain. The marginal stability in the z-plane is the intersection of the root-locus with the unit circle. In the last section we derived the curves for constant time constant τ, peak time t_p and constant damping ratio ζ. In order to design a digital system for transient response, we find the intersection of the root locus with the appropriate curves as they appear on the z-plane. Review the rules for root-locus construction. These are summarized as follows:

Summary of General Rules for Constructing Root-Loci

1. **Number of loci** For $n > m$ the number of loci, that is, the number of branches of the root-locus, is equal to the number of poles of the open-loop transfer function $GH(z)$. The root-locus is symmetrical with respect to the real axis.

2. **Starting and ending points** As K is increased from zero to infinity, the loci of the closed-loop poles originate from the open-loop poles $K=0$, and proceed toward and terminate at the open-loop zeros, $K=\infty$. Zeros tend to attract root-loci toward them and poles tend to repel them.

3. **Root-locus segments on the real axis** For $K>0$, root-loci occurs on a particular segment of the real axis if and only if there are an odd number of total poles and zeros of the open-loop transfer function laying to the right of that segment.

4. **Unit circle intersection** This will give the value of K for marginal stability.
5. **Asymptotes** For most systems of interest, n is greater than or equal to m. For $n>m$ there are $(n-m)$ zeros at infinity, thus for $0<K<\infty$ root-locus ends at zeros at infinity. Root-locus points are asymptotic to straight lines with angles given by $\theta=\dfrac{180r}{n-m}$ $\quad r=0,1,3,\cdots$

Angle of Asymptotes

n - m	Angle of Asymptotes
0	No asymptote
1	180°
2	$\pm 90^\circ$
3	$\pm 180^\circ, \pm 60^\circ$
4	$\pm 45^\circ, \pm 135^\circ$

The asymptotes intersect the real axis at

$$\sigma_a = \frac{\sum \text{poles of } \overline{GH}(z) - \text{zeros of } \overline{GH}(z)}{n-m}$$

6. **Breakaway and re-entry points** These are points on the real axis where two or more branches of the root-locus depart from or arrive at the real axis. Breakaway points may be determined by expressing the characteristic equation for the gain K as a function z $K = -1/\overline{GH}(z)$, and then solving for the breakaway points z from

$$\frac{dK(z)}{dz} = 0$$

The real roots of this equation which satisfy rule 3 are the breakaway or re-entry points.

In addition to the MATLAB control system toolbox **rlocus(num, den)** for root locus plot, MATALB control system toolbox contain the following functions which are useful for interactively finding the gain at certain pole locations and intersect with the unit circle on the z-plane. These are

zgrid generates a grid over an existing discrete z-plane root locus or pole-zero map. Lines of constant damping factor ζ and natural frequency ω_n are drawn in within the unit Z-plane circle. **zgrid(z, Wn)** plots constant damping and frequency lines for the damping ratios in the vector **z** and the natural frequencies in the vector Wn. **zgrid(0, 0)** plots a unit circle.

[K, poles] = rlocfind(num, den) puts up a crosshair cursor in the graphics window which is used to select a pole location on an existing root locus. The root locus gain associated with this point is returned in **K** and all the system poles for this gain are returned in **poles**.

Example 4.7 (chd4ex7.m)

Sketch the root locus for the system shown in Figure 4.11. Also, determine the range of K, for stability from the root locus plot. Sampling time is 0.1 seconds.

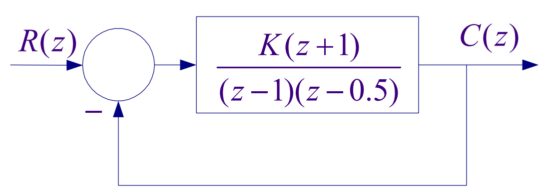


Figure 4.11 Digital feedback control for Example 4.7

We use the following commands numz = [1 1];

```
denz = [1  -1.5  0.5];
Gz = tf(numz, denz, 1)            % Plant transfer function
```

```
rlocus(numz, denz),              % Plot root locus
zgrid(0, 0)                      % Add unit circle to root locus
[K, p] = rlocfind(numz, denz)    % Puts a crosshair on the plot, when clicked at a
                                 % desired location gain K and poles are returned.
```

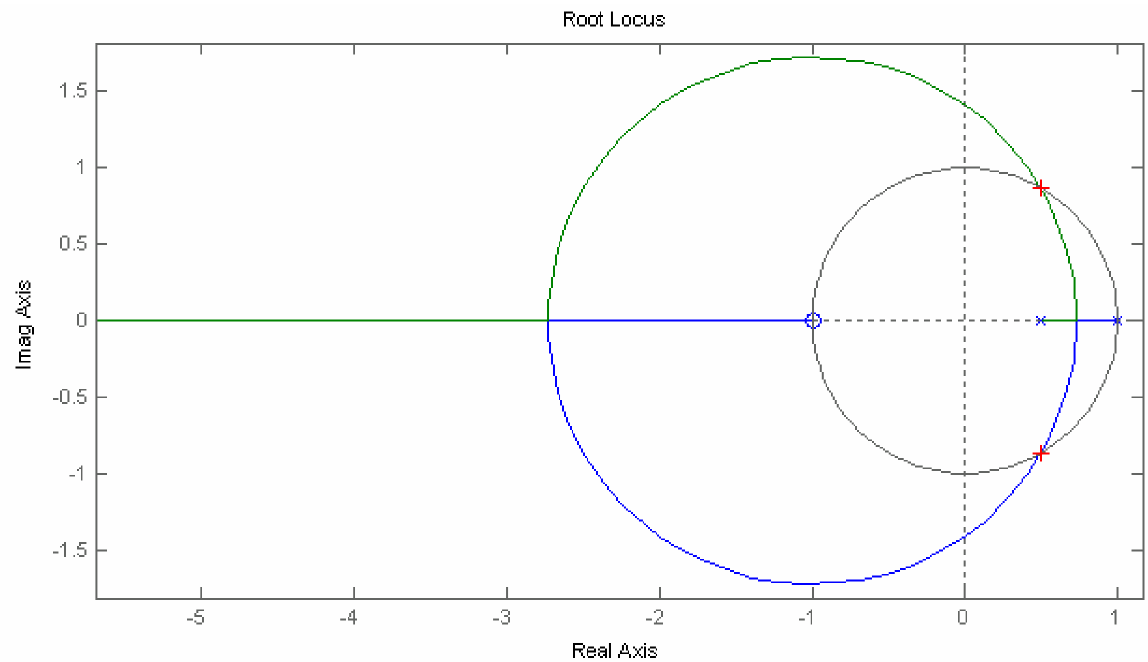


Figure 4.12 Root locus for Example 4.7 with unit circle $\zeta = 0$

```
Transfer function:
     z + 1
-----------------
z^2 - 1.5 z + 0.5

Sampling time: 0.1
Select a point in the graphics window

Selected point =
                      0.4954 + 0.8646i
 k =
        0.50
 p =
        0.4999 + 0.866i
        0.4999 - 0.866i
```

We find the intersection of the root locus with the unit circle at $1\angle 60^\circ$. The gain at this point as given by MATLAB is 0.5. Hence the system is stable for $0 < K < 0.5$. Given the root locus and intersection with the unit circle the gain k to have the closed loop poles at this location is given by

$$k = \frac{\text{products of vector length from poles}}{\text{producta of vector length from zeros}}$$

or

$$k = \frac{(0.866)(1)}{1.732} = 0.5$$

Digital Control System Design

The purpose here is to design a digital controller $D(z)$ that will produce a response in accordance with the design specifications. We consider a cascade controller has shown in Figure 4.13.

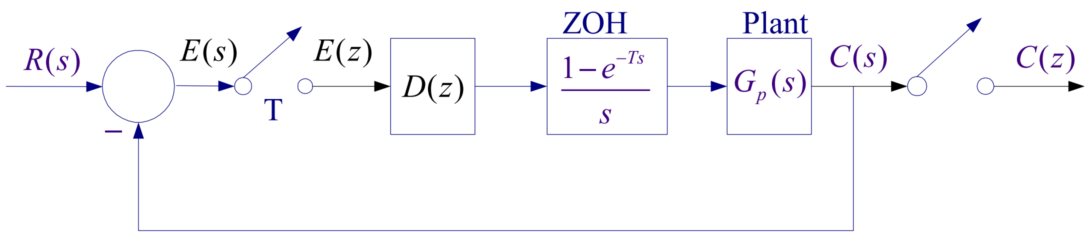


Figure 4.13 Sampled data control system with cascade controller

In the previous section we analyzed digital systems in the z-domain and covered the gain compensation design directly in the z-domain. All of the controllers we studied in EE370 were described by the Laplace transform or differential equations, which are built using analog components. As stated earlier most control systems today use digital computers to implement the controllers. We are now ready to look at the design of various controllers for the sampled data systems that will be implemented in a digital computer. The implementation leads to a small delay (about half the sample period) and aliasing, which need to be addressed in the controller design.

There are several approaches to designing digital controller. One method is called the direct design where the desired closed loop transfer function is specified. All the direct design methods tend to produce intersample oscillations, referred to as ringing, which we will not consider. A second approach is using root locus design directly in the z-plane or frequency response design in the z-plane that will be considered next.
Finally another approach for the digital controller design is to design the controller in the s-plane and then map the compensator to the z-plane using the inverse bilinear mapping. This method is most appealing to the practicing control system designers since they are familiar with the continuous-time control system design. The last approaches lead to a reasonably good digital controller provided the sampling frequency is large enough. Typically for the second-order system a sampling rate of 20 times the damped frequency of oscillation yields very good results.

Root locus design in the z-plane

This approach requires that we find the pulse transfer function $G(z)$, the procedure is illustrated in the following example.

Proportional Controller

In this section we start with the design of a proportional controller with $D(z) = K_P$. In general as the proportional gain K_P is increased, the step response rise time t_r and the steady state error decreases.

Example 4.8 (chd4ex8.m)

Using MATALB obtain the root locus for the system in Example 4.7 and determine the value of K, for a step response damping ratio of 0.707. For this value of K obtain the closed-loop sampled-data transfer function and the step response. Sampling time is 0.1 seconds.

We use the following commands

```
numz = [1  1];
denz = [1  -1.5  0.5];
zeta = input('Enter the desired damping ratio ');
Gz = tf(numz, denz, 0.1)              % Plant transfer function
figure(1), rlocus(numz, denz),   % Plot root locus
zgrid(zeta ,1)                          % Add unit circle to root locus
[K, p] = rlocfind(numz, denz)    % Puts a crosshair, when clicked at a location
                                        %  Gain K and poles are returned
Tz = feedback(K*Gz, 1)            % Closed-loop sampled data transfer function
[numT, denT] = tfdata(Tz, 'v');  % Returns the num and den of the closed-loop TF
figure(2), ltiview('step', Tz)       % Step response
```

```
The result is
Enter the desired damping ratio 0.707

Transfer function:
                 z + 1
              -----------------
              z^2 - 1.5 z + 0.5
Sampling time: 1
Select a point in the graphics window
Selected point =
                 0.7191 + 0.2209i
        K =
              0.0642
        p =
              0.7179 + 0.2208i
```

0.7179 - 0.2208i

Transfer function:

```
0.06416 z + 0.06416
----------------------
z^2 - 1.436 z + 0.5642
```

The plots are shown in Figures 3.14and 3.15.

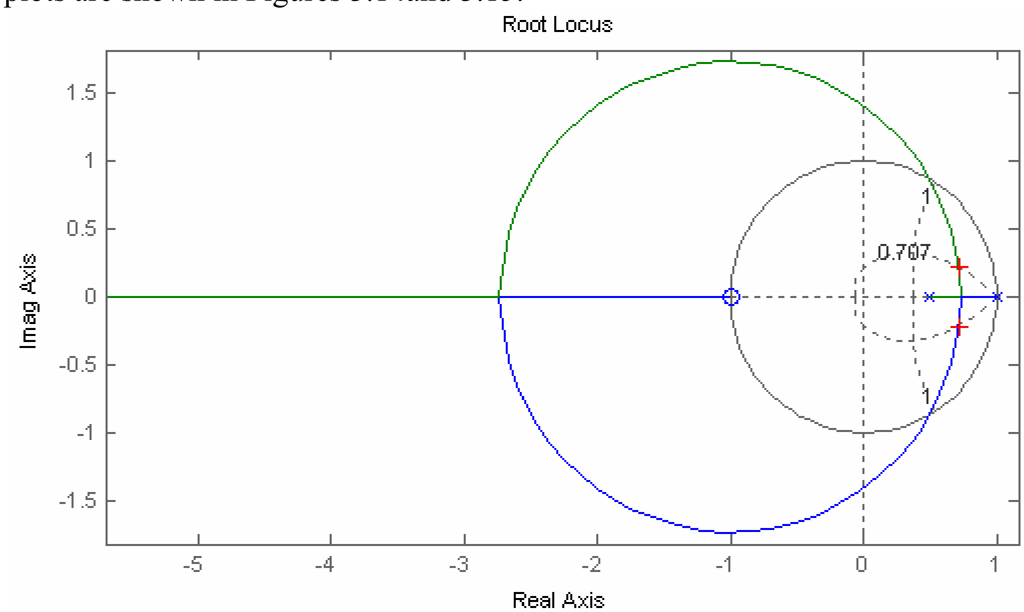


Figure 4.14 Root-locus for Example 4.7 with constant $\zeta = 0.707$ curve.

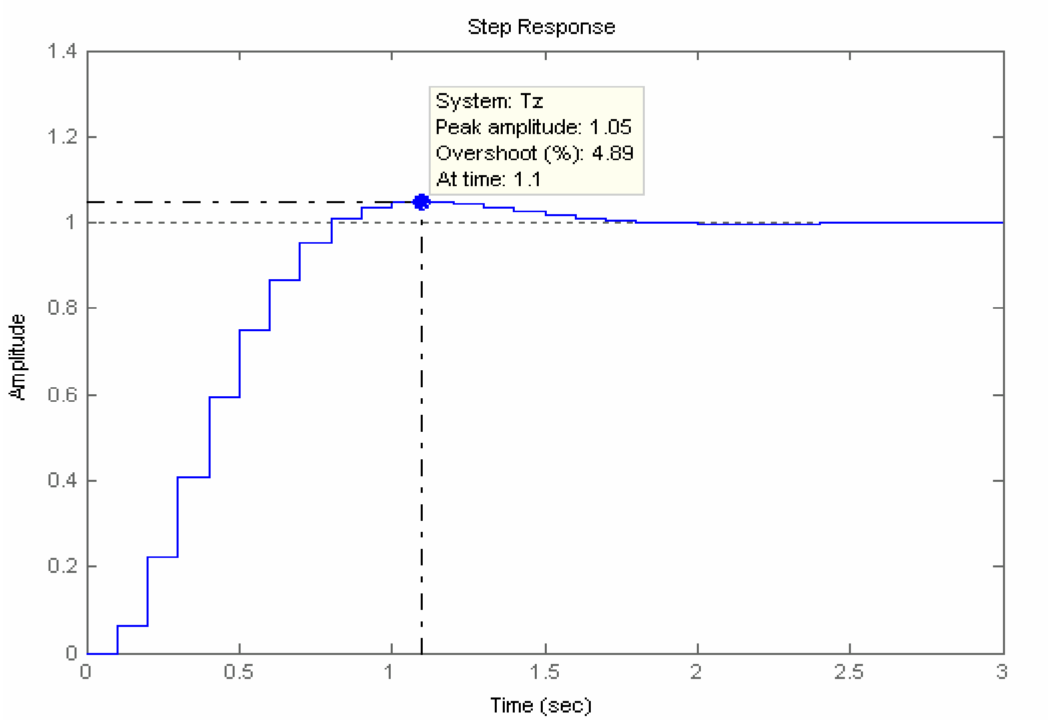


Figure 4.15 Sampled step response for Example 4.7 with unit circle $\zeta = 0$.

Digital PID Controller

Generally proportional controller alone cannot result in good damping and fast response and may have unacceptable steady-state error. Introducing an integral term in the controller can eliminate steady-state errors but may adversely affect the transient response. A term proportional to the derivative of the error can improve the transient behavior but does not reduce the steady-state error. A controller that combines all the tree terms, known as the PID controller can be used to improve both the transient behavior and the steady-state response.

Example 4.9

For the feedback control system shown in Figure 4.16 the plant transfer function and the sensor are

$$G_p(s) = \frac{8}{(s+2)(s+8)} \quad \text{and} \quad H(s) = \frac{20}{s+20}$$

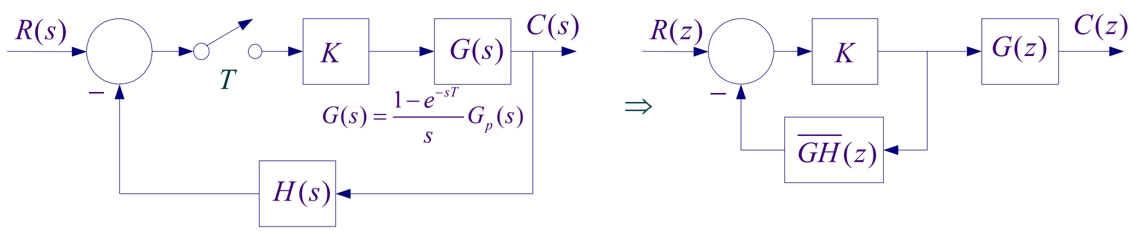


Figure 4.16 Sampled-data system with phase proportional controller

The sampling time is 0.1 second. Using MATLAB

(a) Find the discrete-time transfer function of the process and the sensor
(b) Draw the root locus in the z-plane
(c) Use **rlocfind** command to find the gain K for which the closed loop system becomes unstable.
(d) Design a proportional controller and find K for the step response dominant complex conjugate poles to have a damping ratio of 0.8.
(e) For the value of K found in (d) determine the step response.

We use the following commands

```
numGp = [0 0 8];
denGp = conv([1 2],[1,8]);
Gp = tf(numGp, denGp);          % plant transfer function
numH = [0 20];
denH = [1 20];
H = tf(numH, denH);             % plant transfer function
HGp = H*Gp;                     % Continuous open loop TF
T = 0.1;                        % Sampling time
G = c2d(Gp, T, 'zoh')           % Sampled-data plant TF
```

```
HG = c2d(HGp, T, 'zoh')          % Sampled-data open loop TF
figure(1), rlocus(HG)            % Plot root locus
zeta = 0.707
zgrid(zeta, 1)                   % Add unit circle and zline to root locus
[Kc, pc] = rlocfind(HG)          % Puts a crosshair, when clicked at a location gain K
                                 % and poles are returned
[K, p] = rlocfind(HG)            % Find K for zeta = 0.707
Tz = K*G/(1 + HG*K)              % Closed-loop sampled data transfer function
figure(2), ltiview('step', Tz)   % Step response
```

The result is

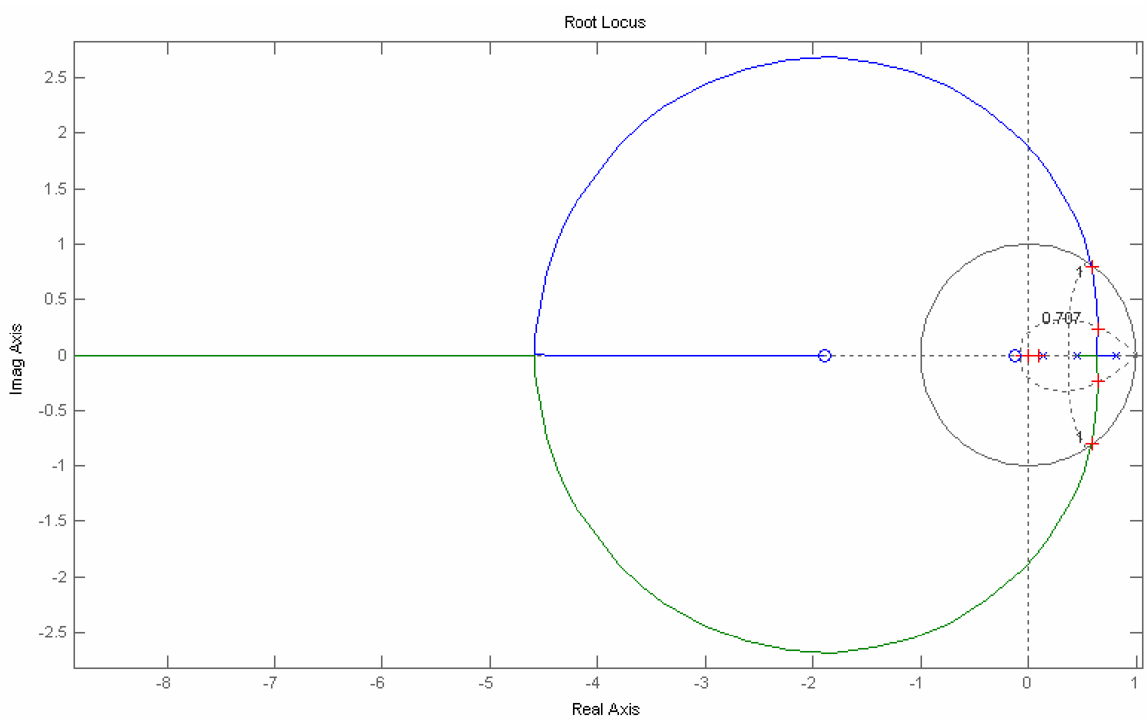
```
Transfer function:
            0.02907 z + 0.02084
            ----------------------
            z^2 - 1.268 z + 0.3679

            Sampling time: 0.1

            Transfer function:
            0.01333 z^2 + 0.02682 z + 0.002997
            ------------------------------------
            z^3 - 1.403 z^2 + 0.5395 z - 0.04979

            Sampling time: 0.1

            zeta =
                      0.7070
            Select a point in the graphics window

            selected_point =
                               0.5879 + 0.8115i
            Kc =
                     17.2306
            pc =
                   0.5877 + 0.8114i
                   0.5877 - 0.8114i
                  -0.0019
           Select a point in the graphics window

            selected_point =
                              0.6434 + 0.2531i
            K =
                    2.0424
            p =
                   0.6423 + 0.2531i
```

0.6423 - 0.2531i
0.0916

Transfer function:

0.05937 z^4 - 0.04075 z^3 - 0.02771 z^2 + 0.02001 z - 0.002119
----
z^5 - 2.644 z^4 + 2.707 z^3 - 1.304 z^2 + 0.274 z - 0.01606

Sampling time: 0.1



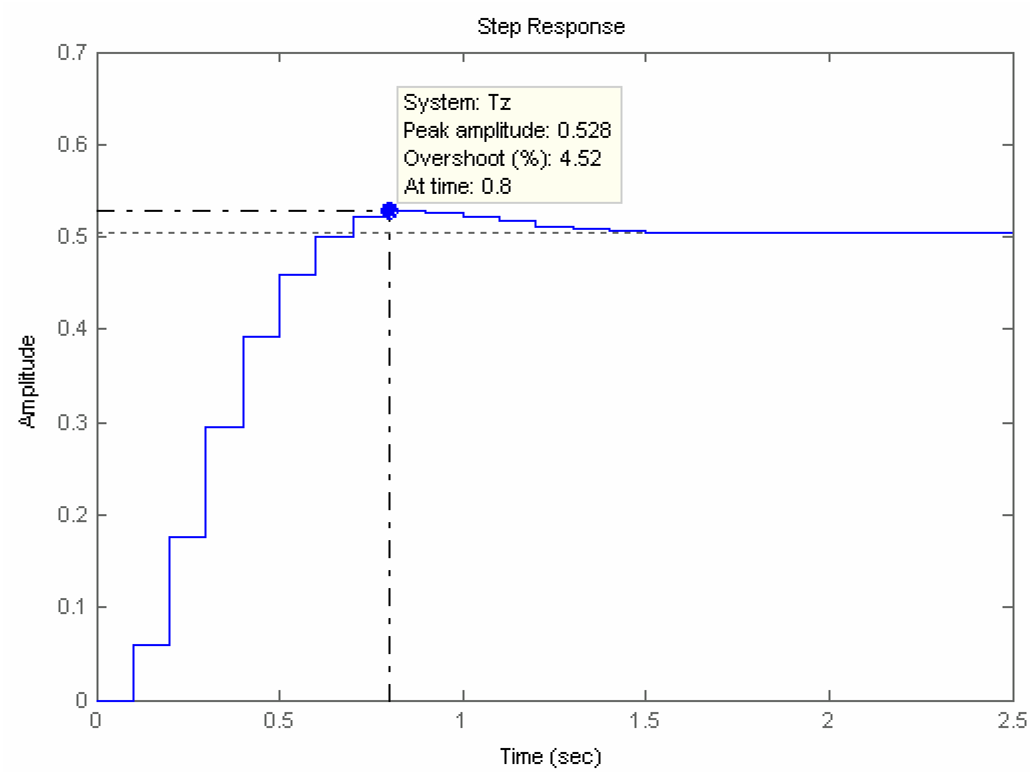


Figure 4.17 Root locus and step response for Example 4.9

Example 4.10

For the digital control system shown design a cascade discrete controller that meets the following time-domain specifications

- The step response damping ratio of $\zeta = 0.707$
- The step response time constant 0.5 seconds

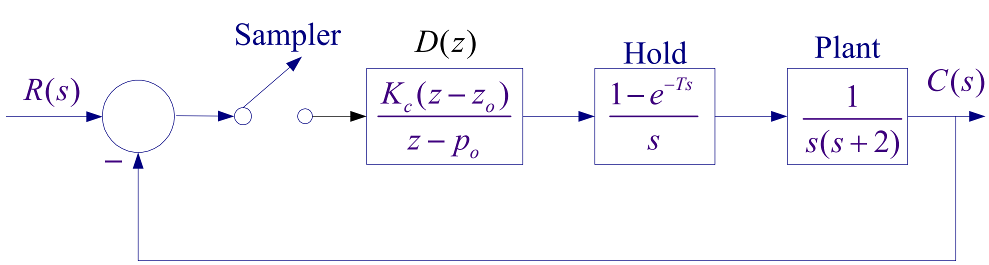


Figure 4.18 Sampled-data system with phase lead controller

The plant transfer function of a system is given by

$$G(s) = \frac{1}{s(s+2)}$$

First we find the desired location of the closed loop poles.

$$\tau = 0.5 = \frac{1}{\zeta\omega_n} \qquad \Rightarrow \qquad \zeta\omega_n = 2$$

$$\zeta = 0.707 = \cos\theta \qquad \Rightarrow \qquad \theta = 45^\circ$$

Therefore

$$s_1 = -2 + j2$$

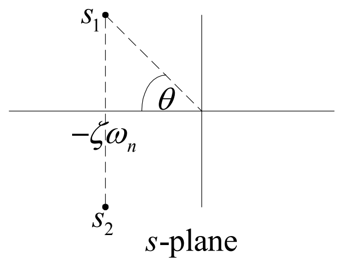


Figure 4.19

The desired closed loop poles in the z-plane is

$$z = e^{Ts} = e^{-0.2 + j0.2} = 0.8024 + \text{j } 0.1627$$

The pulse transfer function is

$$G(s) = \frac{1 - e^{-Ts}}{1} \frac{1}{s^2(s+2)} = (1 - e^{-Ts})\left[\frac{0.5}{s^2} + \frac{0.25}{s+2} + \frac{-0.25}{s}\right]$$

The z-transform is then

$$G(z) = \left[\frac{z-1}{z}\right]\left[\frac{0.5Tz}{(z-1)^2} + \frac{-0.25z}{z-1} + \frac{0.25z}{z - e^{-2T}}\right] = \frac{0.05}{z-1} - 0.25 + \frac{0.25(z-1)}{z - 0.81873}$$

$$= \frac{0.004683(z + 0.9355)}{(z-1)(z-0.81873)} = \frac{0.004683z + 0.004381}{(z^2 - 1.81873 + 0.81873)}$$

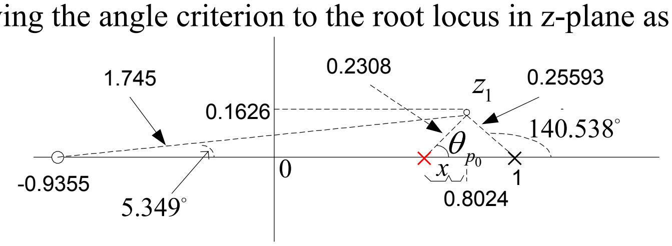
Consider the controller

$$D(z) = \frac{K_c(z - z_0)}{z - p_0}$$

In this design we use pole cancellation, i.e., we select the controller zero equal to 0.81873 and choose the controller pole to satisfy the design specifications. Therefore the controller is

$$D(z) = \frac{K_c(z - 0.81873)}{z - p_0}$$

Applying the angle criterion to the root locus in z-plane as shown in Figure 4.16.



$$\sum \theta_{zi} - \sum \theta_{pi} = -180$$

Therefore

$$5.349-(\theta_{po}+140.538)=-180 \quad \Rightarrow \quad \theta_{po}=44.81^{\circ}$$

$$\tan 44.81=\frac{0.1626}{x} \Rightarrow x=0.16374$$

$$p_o=0.8024-0.16374=0.6387$$

$$D(z)G_p(z)==\frac{0.004683K_c(z+0.9348)}{(z-1)(z-0.6387)}$$

Applying the magnitude criterion

$$0.004683K_c=\frac{(0.2559)(0.2308)}{1.745} \quad \Rightarrow \quad K_c=7.227$$

Therefore the controller is

$$D(z)=\frac{7.227(z-0.81873)}{z-0.6387}$$

The compensated open loop transfer function is

$$D(z)G_p(z)=\frac{0.03384(z+0.9355)}{(z-1)(z-0.6387)}=\frac{0.03384z+0.03166)}{(z^2-1.639z+0.6387)}$$

The compensated closed loop transfer function is

$$T(s)=\frac{D(z)G(z)}{1+D(z)G(z)}=\frac{0.03384z+0.03166)}{(z^2-1.605z+0.6703)}$$

The following MATLAB commands are used to obtain the controller transfer function, the closed loop pulse transfer function and the step response of the compensated system.

```
numGp=[0 1];
denGp=[1 2 0];
Gp=tf(numGp, denGp)
T = input('Enter the sampling time in sec ');                  % Sampling time
Gz = c2d(Gp, T, 'zoh')              % open-loop sampled-data transfer function
[numGz,denGz]=tfdata(Gz, 'v');              % Returns the num and den of Gz
rn=roots(numGz)                                                  % zeros of Gz
rd=roots(denGz)                                                  % poles of Gz
s1 = -2 +j*2;              %desired location of the closed loop pole in s-plane
z1 = exp(T*s1);            %desired location of the closed loop pole in z-plane
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                       % The following commands evaluates the vector length
                    % from poles and zeros, their angles, controller pole, and Kc
                                                      % Using graphical method
zh=-rn+real(z1); zv=imag(z1);
Mz=sqrt(zh^2+zv^2); thetaz=atan(zv/zh);   %Vector length and angle to zero
ph=rd(1)-real(z1); pv=imag(z1);
Mp=sqrt(ph^2+pv^2);thetap=pi-atan(pv/ph);%Vector length and angle to zero
thetapo=pi+thetaz-thetap;                   % Angle of po applying angle criteria
```

```
x=pv/tan(thetapo);  po=real(z1)-x
Mpo=sqrt(x^2+pv^2);
Kc=Mp*Mpo/(Mz*numGz(2))              % Kc applying magnitude criteria
numD=Kc*[1 -rd(2)];
denD=[1  -po];
D=tf(numD, denD, T)                          % Controller transfer function
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
DGz=D*Gz;                                    % Open loop transfer function
DG=minreal(DGz)                       % eliminate duplicate poles and zeros
Tz=feedback(DG, 1)                         % Closed loop transfer function
ltiview('step', Tz)
```

The response meets the design specification and the result is in close agreement with the response of the analog system.

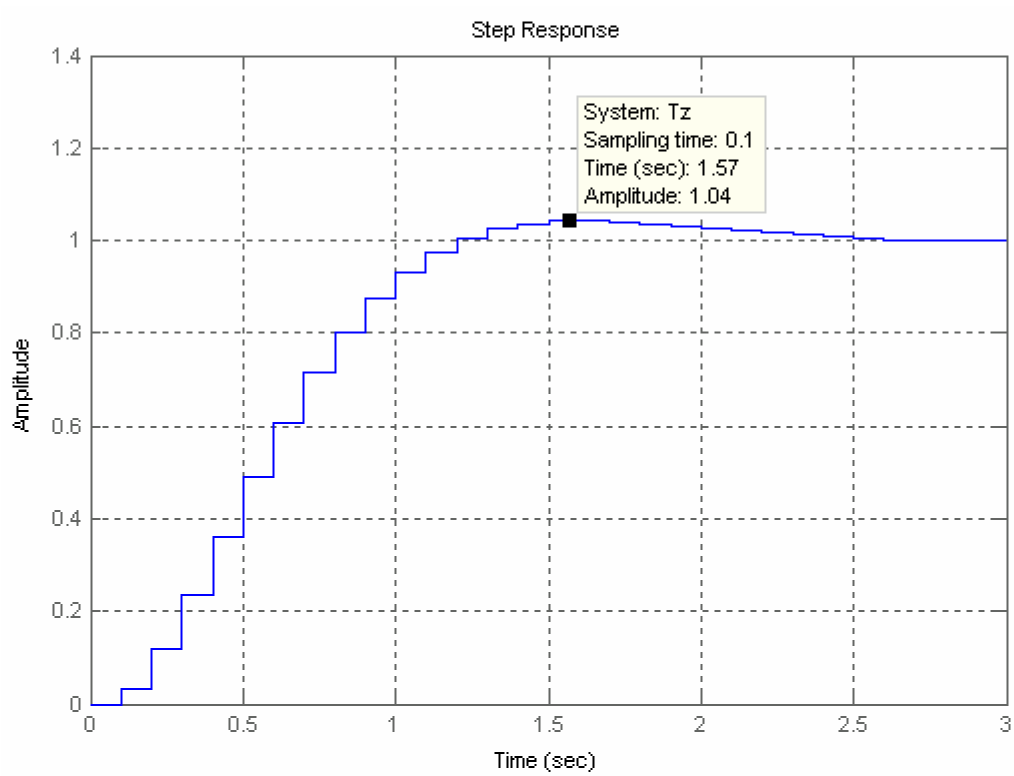


Figure 4.20 Step response for Example 4.10

PID Controller

As we have seen proportional controller K_P can be used to improve the steady-state error. As K_P is increased the steady state error is reduced, however, the step response overshoot will increase and for characteristic equation higher than the second order it may cause system to become unstable. In the case of continuous-time control system if we desire to eliminate the steady state error to zero we use an integrator that is we introduce a pole at origin. For example a system type zero has a steady state error with a unit step input.

Introducing an integral controller will reduces the steady state error to zero. The Transfer function of a PI controller is

$$G_c(s) = K_P + \frac{K_I}{s}$$

This can be written as

$$G_c(s) = \frac{K_P(s + \frac{K_I}{K_P})}{s} = \frac{K_P(s - z_0)}{s}$$

That is in addition to a pole at origin; the PI controller introduces a zero at $z_0 = -\frac{K_I}{K_P}$

Similarly a PI controller in the z-domain can be considered to introduce a zero and a pole at $z = 1$.

$$D(z) = \frac{K_P(z + \frac{K_I}{K_P})}{z - 1} = \frac{K_P(z - z_0)}{z - 1}$$

While the PI controller reduces the steady-state error to zero, it has destabilization effect because of the addition of poles at origin in the s-plane or at $z = 1$ in the z-plane. In order to improve the transient characteristics a derivative controller is also added. In the s-domain the PD controller transfer function is

$$G_c(s) = K_P + K_D s = K_D(s + \frac{K_P}{K_D}) = K_D(s - z_0)$$

That is the controller introduces a zero at $z_0 = -\frac{K_P}{K_D}$

Similarly a digital PD controller introduces a zero in the z-plane.

Example 4.11

For the digital control system shown design a PI controller that meets the following time-domain specifications

- The step response dominant complex closed loop poles damping ratio of $\zeta = 0.707$
- The step response dominant complex closed loop poles time constant 1 second.

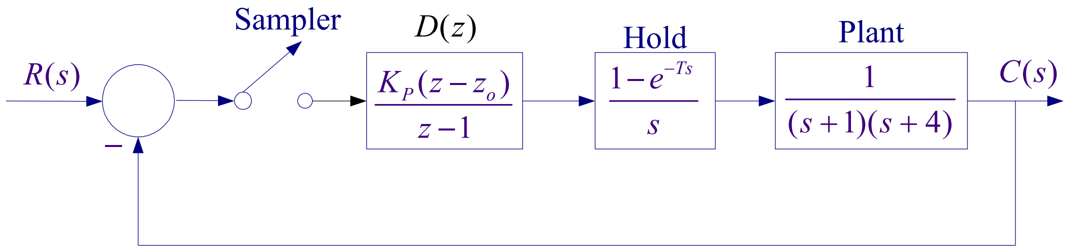


Figure 4.21 Sampled-data system with a PI controller

Sampling time is 0.1 second.

The plant transfer function of a system is given by

$$G(s) = \frac{1}{(s+1)(s+4)}$$

First we find the desired location of the closed loop poles.

$$\tau = 1 = \frac{1}{\zeta\omega_n} \qquad \Rightarrow \qquad \zeta\omega_n = 1$$

$$\zeta = 0.707 = \cos\theta \qquad \Rightarrow \qquad \theta = 45^\circ$$

Therefore

$$s_1 = -1 + j1$$

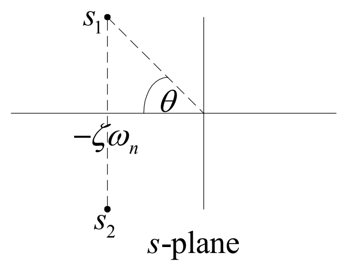


Figure 4.22

The desired closed loop poles in the z-plane is

$$z = e^{Ts} = e^{-0.1 + j0.1} = 0.90032 + j0.090333$$

The pulse transfer function is

$$G(s) = \frac{1-e^{-Ts}}{1}\frac{1}{s(s+1)(s+4)} = (1-e^{-Ts})\left[\frac{0.25}{s} + \frac{-1/3}{s+1} + \frac{-1/12}{s+4}\right]$$

The z-transform is then

$$G(z) = \left[\frac{z-1}{z}\right]\left[\frac{0.25}{z-1} + \frac{-1/3}{z-e^{-T}} + \frac{-1/12}{z-e^{-4T}}\right]$$

$$= \frac{0.004248(z+0.84655)}{(z-0.90484\ )(z-0.67032)} = \frac{0.004248z + 0.003596}{(z^2 - 1.575 + 0.6065)}$$

Consider the controller

$$D(z) = \frac{K_P(z-z_0)}{z-1}$$

The controller introduces a pole at $z = 1$. The controller zero is found by applying the angle criterion as shown in Figure 4.326.

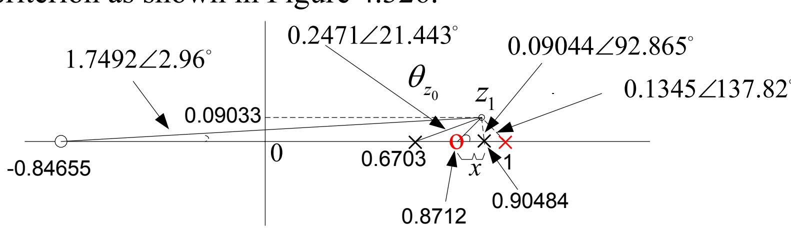


Figure 4.23

$$\sum \theta_{zi} - \sum \theta_{pi} = -180$$

Therefore

$$2.96 + \theta_{zo} - (21.443 + 92.865 + 137.82) = -180 \quad \Rightarrow \quad \theta_{zo} = 69.164^\circ$$

$$\tan 69.164 = \frac{0.09033}{x} \Rightarrow x = 0.034379$$

$$z_o = 0.90032 - 0.034379 = 0.866$$

$$D(z)G_p(z) == \frac{0.004248K_P(z + 0.84655)}{(z - 0.90484)(z - 0.67032)(z - 1)}$$

Applying the magnitude criterion

$$0.004248K_P = \frac{(1.7492)(0.096654)}{(0.09044)(0.2471)(0.1345)} \quad \Rightarrow \quad K_P = 4.2$$

Therefore the controller is

$$D(z) = \frac{4.2(z - 0.866)}{z - 1}$$

The compensated open loop transfer function is

$$D(z)G_p(z) = \frac{0.01778\, z^2 - 0.0003447\, z - 0.01304}{z^3 \text{ - } 2.575\, z^2 \;+\; 2.182\, z \text{ - } 0.6065}$$

The compensated closed loop transfer function is

$$T(s) = \frac{D(z)G(z)}{1 + D(z)G(z)} = \frac{0.01778z^2 - 0.0003447z - 0.01304}{z^3 - 2.557\, z^2 + 2.181\, z - 0.6196}$$

Roots of the z-domain characteristic equation are

```
r =
    0.9003 + 0.0903i
    0.9003 - 0.0903i
    0.7567
```

The step response is as shown in Figure 4.24.

From $z = e^{Ts}$, the s-plane roots are given by $s = \ln(z)/T$, or

```
s =
   -1.0000 + 1.0000i
   -1.0000 - 1.0000i
   -2.7873
```

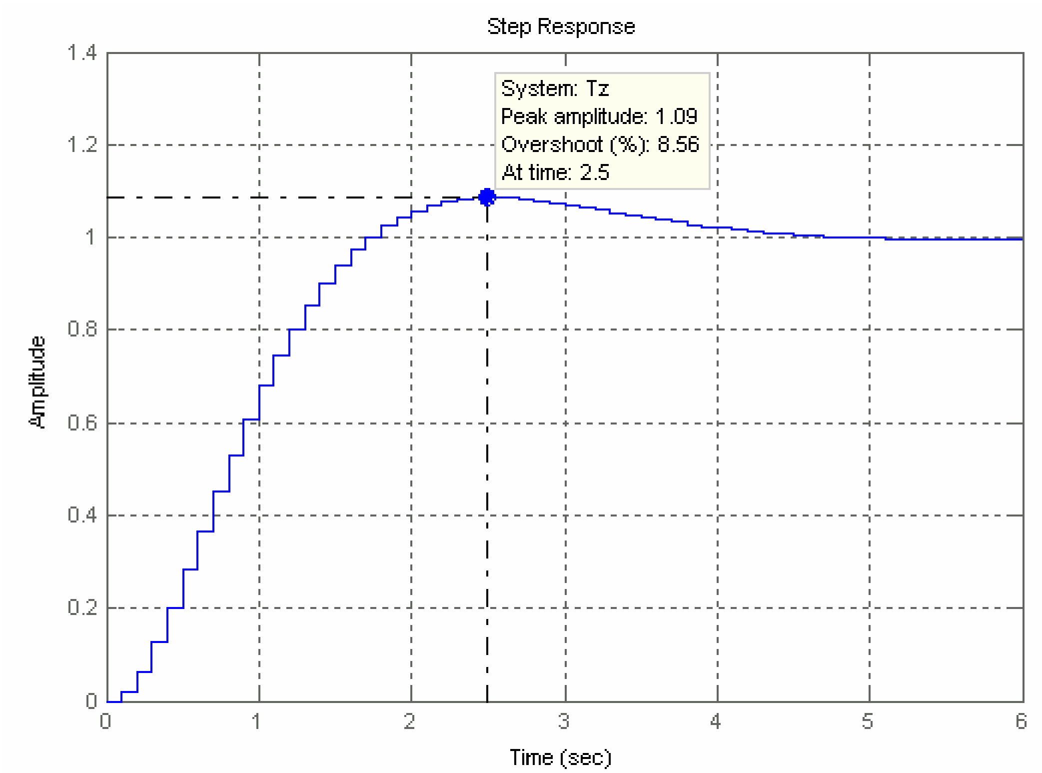


Figure 4.24 Step response for system in Example 4.11.

Root locus design in the s-plane

One method for determining the digital controller $D(z)$ is to ignore the sampler and obtain the analog controller $G_c(s)$ using the techniques for the continuous-time control system. The controller is then converted to $D(z)$ using the inverse bilinear transformation. The inverse bilinear transformation

$$s = \frac{2}{T}\frac{z-1}{z+1} \tag{4.21}$$

preserves the frequency response and root locus and the stability property of the closed loop system. This way the stable poles in the s domain maps into stable poles in the z-domain. The design for system in Example 4.9 is obtained in s-plane is illustrated in

Example 4.12

Design the digital controller for Example 4.10 using root locus design in s-plane. We consider the design of the controller $G_c(s)$

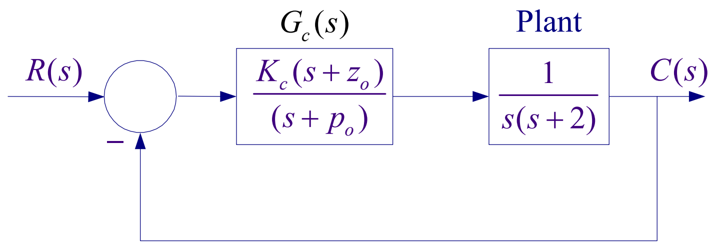


Figure 4.25 Continuous-time control system with phase lead controller

For the design specification given in Example 4.10 ($\zeta = 0.707$, $\tau = 0.5$ second), the desired close loop pole location is

$$s_1 = -2 + j2$$

Using the pole cancellation, for the controller $z_o = 2$.

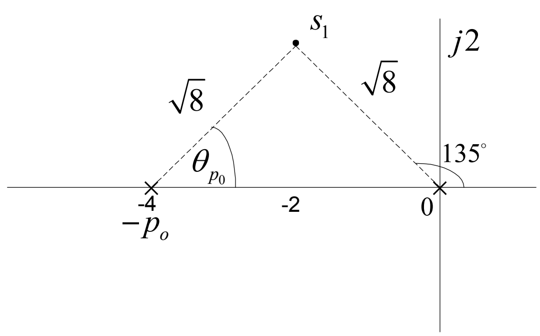


Figure 4.26

Applying the angle criterion

$$\sum \theta_{zi} - \sum \theta_{pi} = -180$$

Therefore

$$0 - (\theta_{po} + 135) = -180 \quad \Rightarrow \quad \theta_{po} = 45^\circ$$

$$\tan 45 = \frac{2}{x} \Rightarrow x = 2$$

$$-p_o = -2 - 2 = -4$$

$$G_c(z)G_p(z) == \frac{K_c(s+2)}{(s+4)}$$

Applying the magnitude criterion

$$K_c = \sqrt{8}\ \sqrt{8} \quad \Rightarrow \quad K_c = 8$$

Therefore the controller is

$$G_c(s) = \frac{8(s+2)}{s+4}$$

We can now map the compensator $G_c(s)$ to the z-plane using the inverse bilinear transformation

$$D(z) = \left[\frac{8(s+2)}{(s+4)}\right]_{s=20\frac{z-1}{z+1}} = \frac{8\left(20\dfrac{z-1}{z+1}+2\right)}{20\dfrac{z-1}{z+1}+4} = \frac{8(22z-18)}{24z-16}$$

or

$$D(z) = \frac{7.33(z-0.818)}{(z-0.667)}$$

The z-domain design was found to be $D(z) = \dfrac{7.33(z-0.818)}{(z-0.667)}$

This is approximately the same controller as the one designed in the z-domain for Example 4.10. For sampling time less than $T = 0.1$ second the design in s-plane will be remarkably accurate. It is therefore convenient to carry out the design in the s-plane and then convert to a digital filter. In the following section we shall review the root locus design for various controllers by means of several examples

MATLAB has a very useful command for inverse bilinear transformation from s-plane into the z-plane.

[numz, denz] = bilinear(num, den fs) converts the s-domain transfer function to a z-transform discrete equivalent obtained from the bilinear transformation defined in (4.21) Where f_s is the sample frequency in Hz. The above function can be used when the s-domain transfer function is expressed in terms of zeros and poles, or when the s-domain system is modeled in state variable representation.

Example 4.13

Convert the controller $G_c(s)$ in Example 4.12 to the digital controller using the **bilinear** function. The sampling time is 0.1 second, i.e., $f_s = 1/0.1 = 10$ Hz.

We use the following commands

```
fs=10;
num=8*[1  2];
den=[1  4];
[numz, denz]=bilinear(num, den, fs)
Gz=tf(numz, denz, 0.1)
```

The result is

```
Transfer function:
7.333 z - 6
-----------
z - 0.6667
```

Example 4.14

Design the digital controller for Example 4.11 using root locus design in s-plane. We consider the design of the controller $G_c(s)$

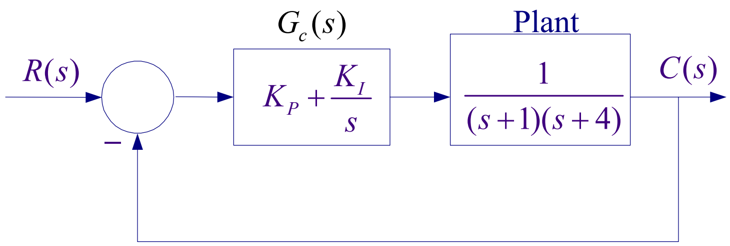


Figure 4.25 Continuous-time control system with phase lead controller

$$G_c(s) = K_P + \frac{K_I}{s} = K_P\frac{K_P(s+\frac{K_I}{K_P})}{s} = \frac{K_P(s+z_0)}{s}$$

For the design specification given in Example 4.10 ($\zeta = 0.707$, $\tau = 0.5$ second), the desired close loop pole location is

$$s_1 = -1 + j1$$

Applying angle criterion

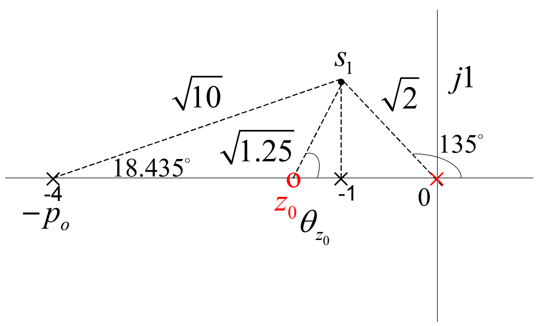


Figure 4.26

Applying the angle criterion

$$\sum\theta_{zi} - \sum\theta_{pi} = -180$$

Therefore

$$\theta_{zo} - (135 + 18.435 + 90) = -180 \quad \Rightarrow \quad \theta_{zo} = 63.435^\circ$$

$$\tan 63.435 = \frac{1}{x} \Rightarrow x = 0.5$$

$$-z_o = -1 - 0.5 = -1.5$$

$$G_c(z)G_p(z) == \frac{K_P(s+1.5)}{s(s+1)(s+4)}$$

Applying the magnitude criterion

$$K_P = \frac{\sqrt{2}\ (1)\sqrt{10}}{\sqrt{1.25}} \quad \Rightarrow \quad K_P = 4$$

Therefore the controller is

$$G_c(s) = \frac{4(s+1.5)}{s} = 4 + \frac{6}{s}$$

We can now map the compensator $G_c(s)$ to the z-plane using the inverse bilinear transformation

$$D(z) = \left[4 + \frac{6}{s}\right]_{s=20\frac{z-1}{z+1}} = 4 + \frac{0.3(z+1)}{z-1} = \frac{4.3z-3.7}{z-1}$$

or

$$D(z) = \frac{4.3(z-0.8605)}{z-1}$$

The z-domain design was found to be

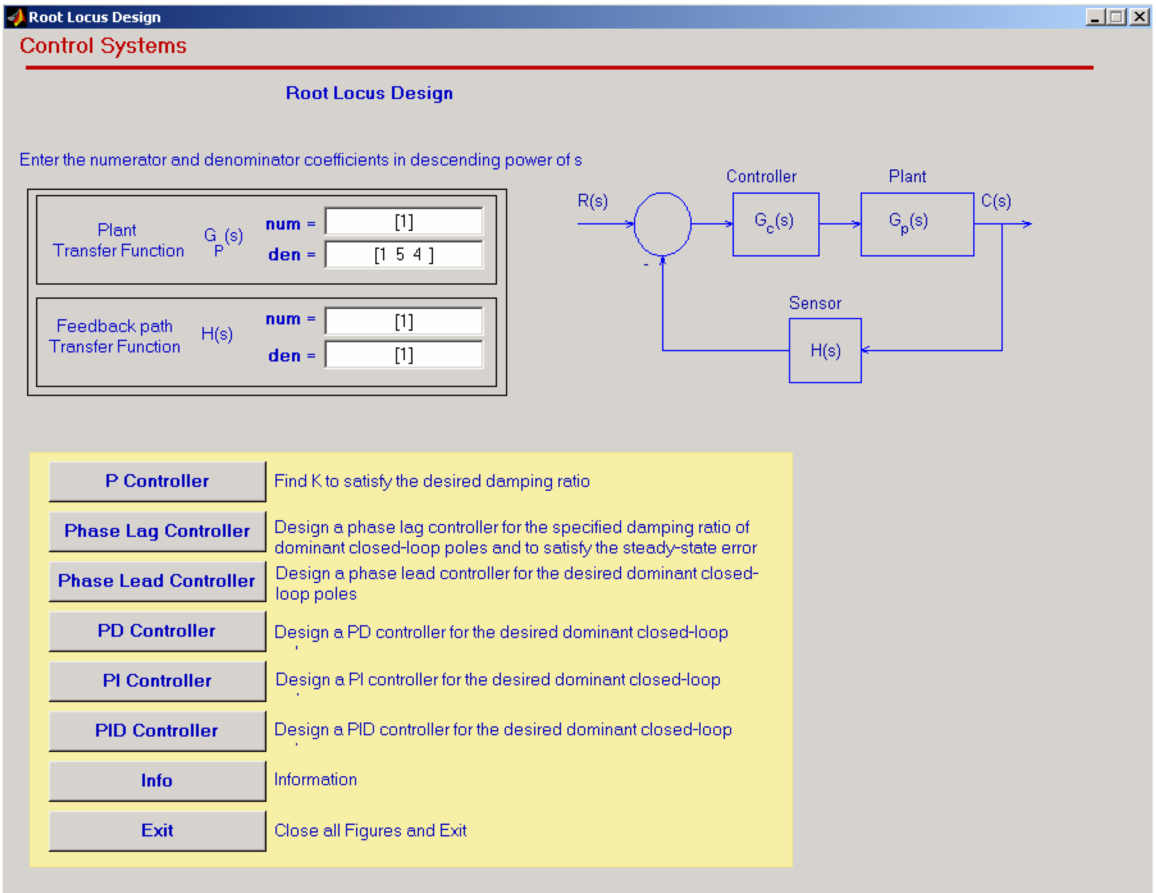
$$D(z) = \frac{4.2(z-0.866)}{(z-1)}$$

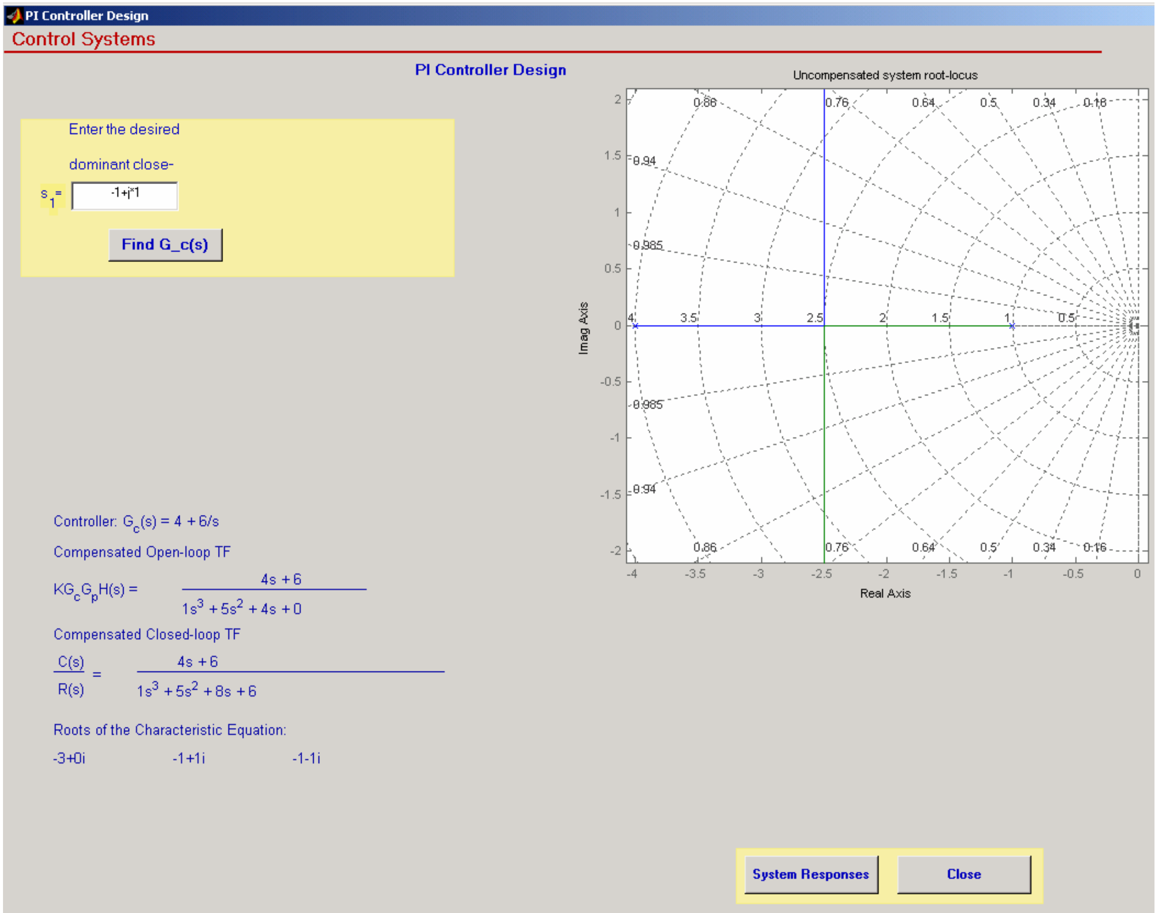
This is approximately the same controller as the one designed in the z-domain for Example 4.11.

I have developed a GUI program for the controller design in the s-plane. To use this program at the MATLAB prompt type

>> rldesigngui

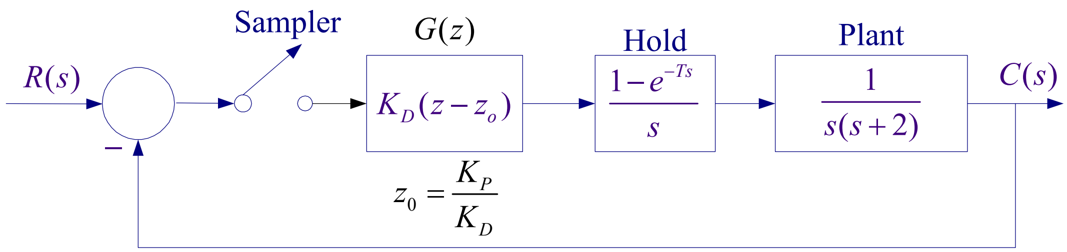
Define the plant transfer function and select the desired controller

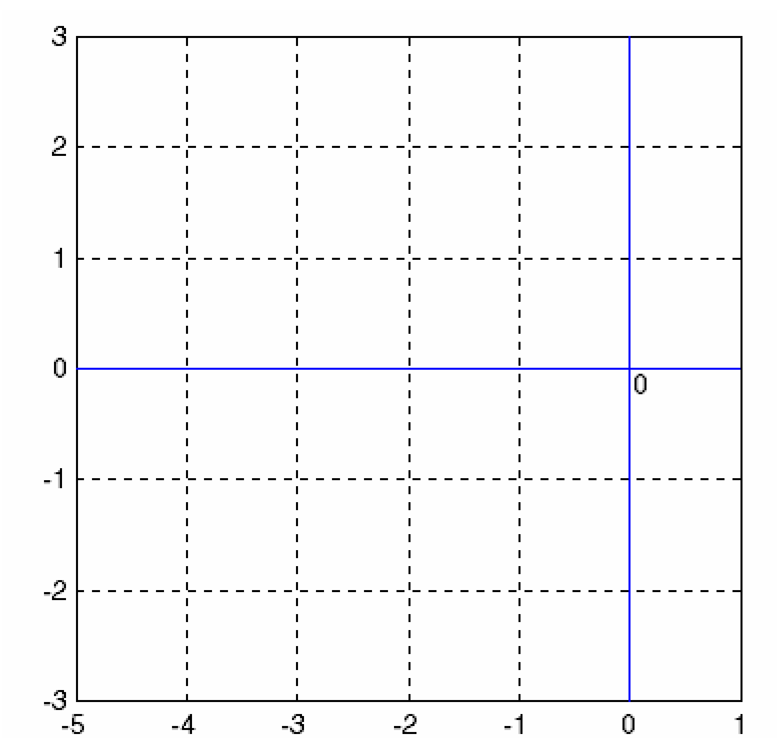



PI Controller Design
Control Systems
PI Controller Design
Enter the desired
dominant close-
s1 =
-1+i*1
Find G_c(s)
Controller: Gc(s) = 4 + 6/s
Compensated Open-loop TF
KGcGpH(s) = 4s + 6 / 1s3 + 5s2 + 4s + 0
Compensated Closed-loop TF
C(s)/R(s) = 4s + 6 / 1s3 + 5s2 + 8s + 6
Roots of the Characteristic Equation:
-3+0i
-1+1i
-1-1i
Uncompensated system root-locus
Imag Axis
Real Axis
System Responses
Close

Homework
In the following problems you may design the controller in the s-domain and use the bilinear transformation to obtain the digital controller

1. For the control system shown, design a PD controller for the step response dominant poles to have a damping ratio of $\zeta = 0.707$ and a time constant of $\tau = 0.5\,\text{sec}$.



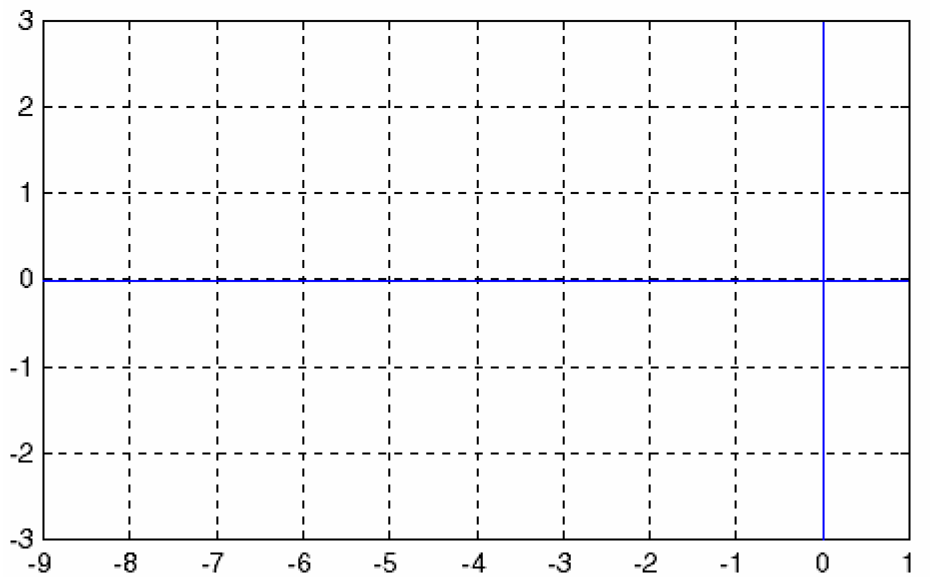


2. The open loop transfer function of a plant is

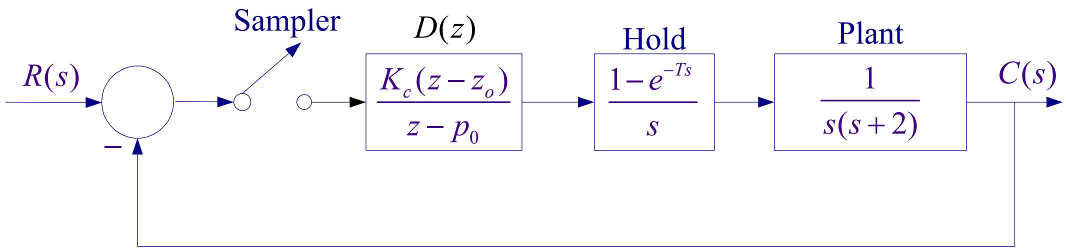
$$G_P(s) = \frac{1}{(s+1)(s+4)}$$

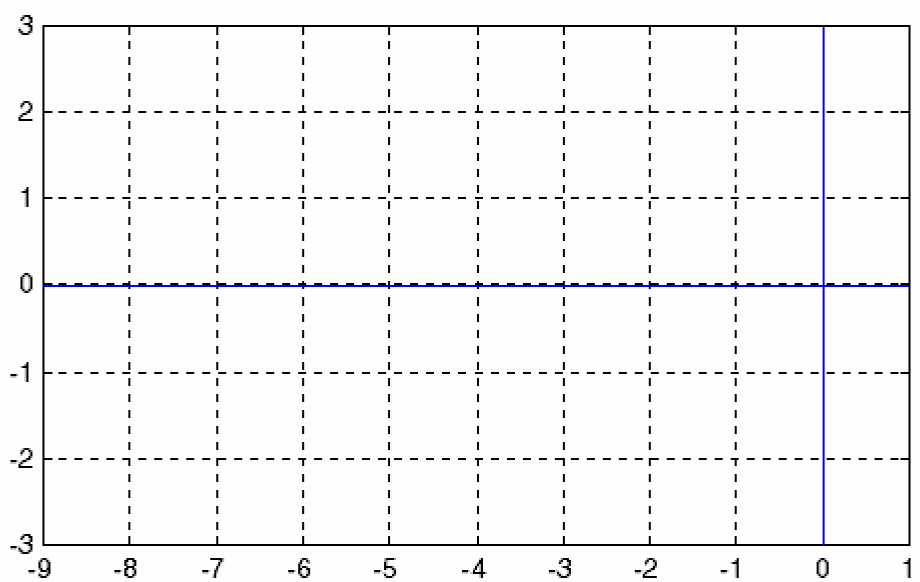
Design a PD controller $G_c(s) = K_P + sK_D$ for the step response to meet the following specifications: Use the bilinear transformation to find the digital controller

- Damping ratio of the complex conjugate poles $\zeta = 0.8$
- Time constant of the complex conjugate poles $\tau = 0.25$ second



3. For the control system shown, design a phase lead controller for the step response dominant poles to have a damping ratio of $\zeta = 0.707$ and a time constant of $\tau = 0.5\,\text{sec}$. Use the bilinear transformation to find the digital controller

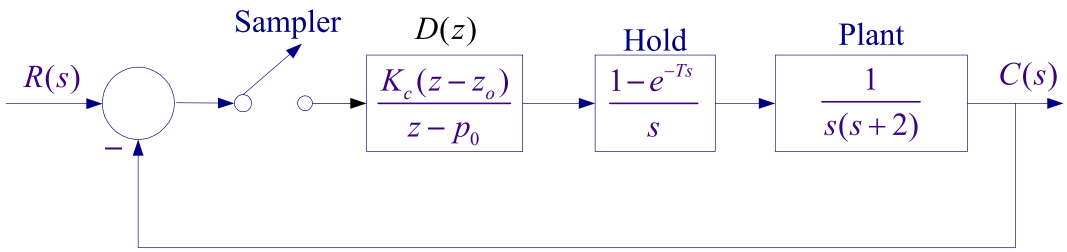


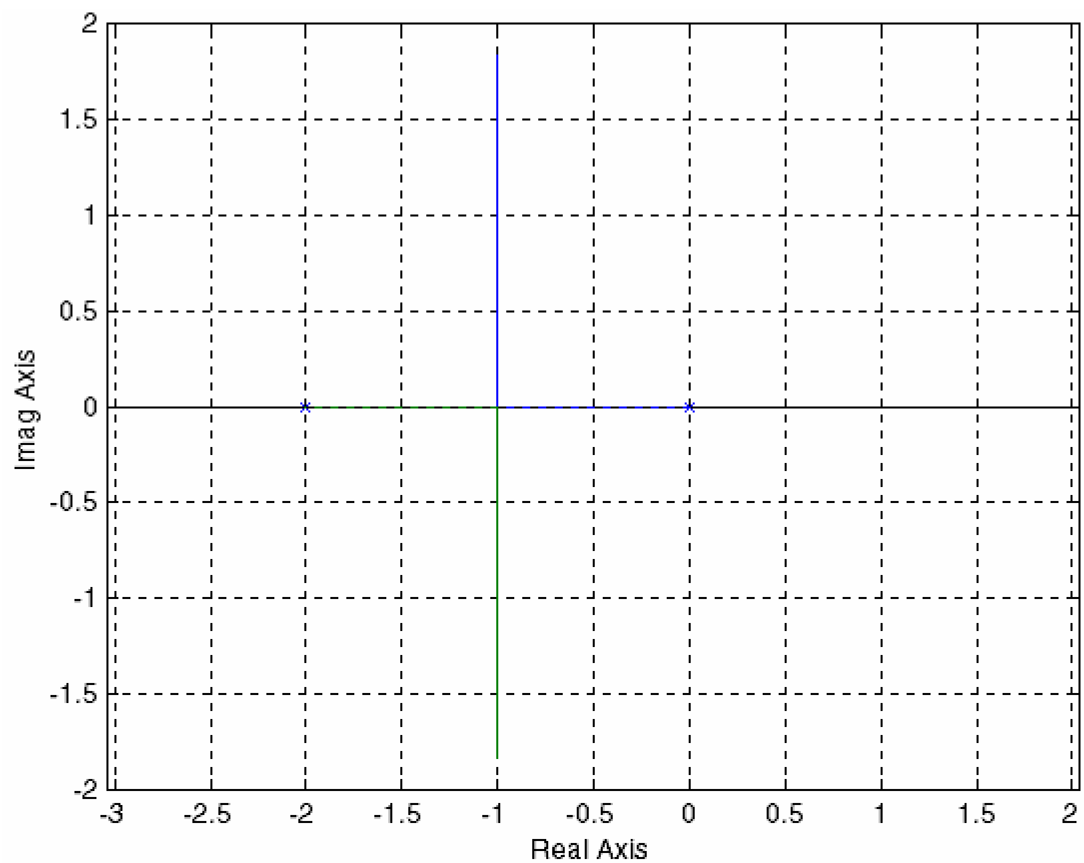


In s-domain select the controller zero at $-Z_0 = -3$

4. For the control system in Example 7, design a phase lag controller for the following specifications:

1. The step response dominant poles to have a damping ratio of $\zeta = 0.707$
2. Steady-state error due to a unit ramp input $e_{ss} = 0.01$





Obtain the compensated open loop transfer function.

Check your s-domain design using the **rldesigngui** program