Digital Control Systems - Chapter 4 Notes

Open-Loop Discrete-Time Systems

4.1 Introduction

- Derive analysis methods for open-loop discrete-time systems

4.2 Relationship between $E(z)$ and $E^*(s)$

z-Transform has a direct relationship to the starred transform:

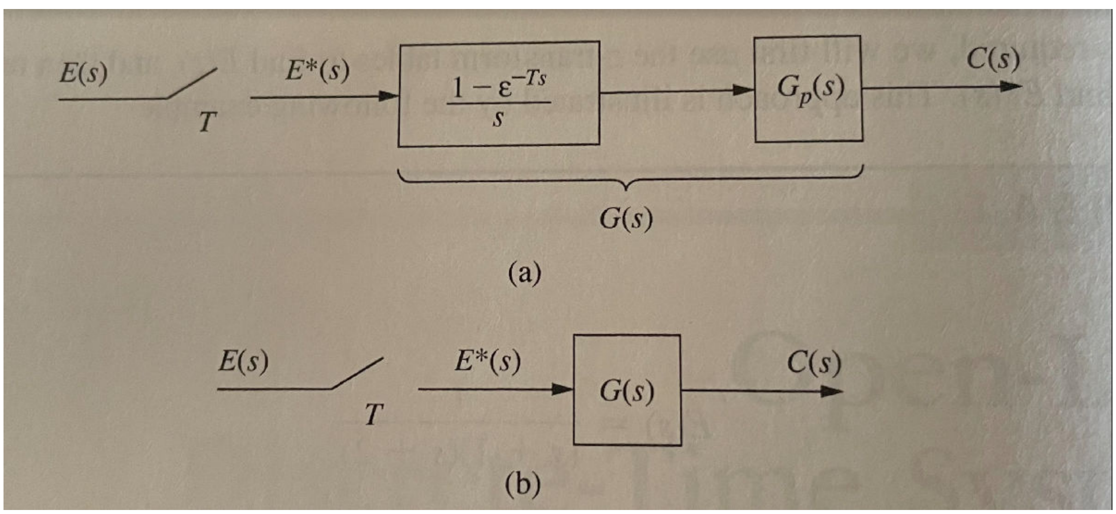
$$E(z) = E^*(s)|_{\varepsilon^{sT}=z}$$

- Moving forward will consider $E(z)$ when working with DT systems, instead of starred transform
- Note: can also use this formula backwards to solve for starred transform

Can use previous equations solving for the poles to derive $E(z)$:

$$E(z) = \sum_{\text{poles of } E(\lambda)} \left[\text{residues of } E(\lambda)\frac{1}{1 - z^{-1}\varepsilon^{T\lambda}}\right]$$

4.3 The Pulse Transfer Function



Transfer function for the above figure:

$$G(s) = \frac{1 - \varepsilon^{-Ts}}{s} G_p(s)$$

- This transfer function of $G(s)$ contains the T.F. of the data hold and plant

$$C(s) = G(s)E^*(s)$$

$$C^*(s) = E^*(s)G^*(s)$$

$$C(z) = E(z)G(z)$$

- $G(z)$ is now called the *Pulse Transfer Function* and is the transfer functions of the sampled input and the output *at the sampling instants*.
- Generally, we choose the sample frequency such that the response at sampling instants gives a very good indication of the response between sampling instants

Example 4.2

Suppose we wish to find the z-transform of

$$A(s) = \frac{1 - \varepsilon^{-\mathrm{Ts}}}{s(s+1)} = \frac{1}{s(s+1)}(1 - \varepsilon^{-\mathrm{Ts}})$$

First, we consider

$$B(s) = \frac{1}{s(s+1)}$$

and

$$F^*(s) = 1 - \varepsilon^{-\mathrm{Ts}} \rightarrow F(z) = 1 - z^{-1} = \frac{z-1}{z}$$

Then, using z-transform tables

$$B(z) = z\left[\frac{1}{s(s+1)}\right] = \frac{z(1-\varepsilon^{-T})}{(z-1)(z-\varepsilon^{-T})}$$

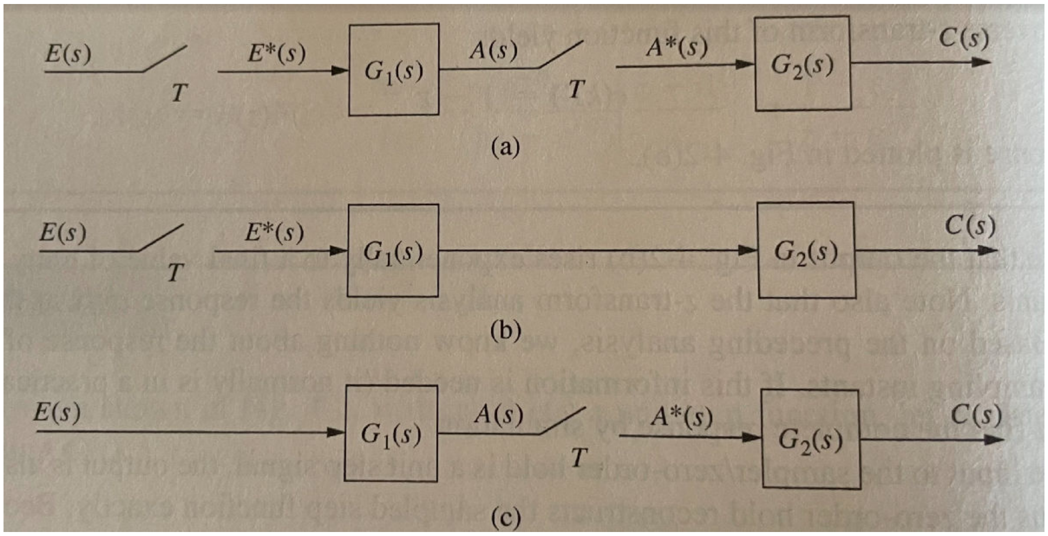
and, putting these all together

$$A(z) = B(z)F(z) = \frac{z(1-\varepsilon^{-T})}{(z-1)(z-\varepsilon^{-T})}\left[\frac{z-1}{z}\right] = \frac{1-\varepsilon^{-T}}{z-\varepsilon^{-T}}$$

- z-Transform analysis yields the response *only* at the sampling instants, and we know nothing about the response of the system between sampling instants
- Given, $c(t)$ we can find $c(\mathrm{kT})$ by replacing t with kT, BUT we *cannot* replace nT with t and have the correct expression for $c(t)$, in general

For many control systems, the steady-state gain for a constant input is important → **DC Gain**

$$\text{dc gain} = \lim_{z \to 1} G(z) = \lim_{s \to 0} G_p(s)$$



Now consider 3 different scenarios:

For Figure 4-3(a):

$$C(s) = G_2(s)A^*(s) \rightarrow C(z) = G_2(z)A(z)$$

Also,

$$A(s) = G_1(s)E^*(s) \rightarrow A(z) = G_1(z)E(z)$$

Then, combining we get:

$$C(z) = G_2(z)G_1(z)E(z)$$

and the total transfer function is the product of the pulse transfer functions.

For Figure 4-3(b):

$$C(s) = G_1(s)G_2(s)E^*(s) \rightarrow C(z) = \overline{G_1G_2}(z)E(z)$$

where

$$\overline{G_1G_2}(z) = z\{G_1(s)G_2(s)\}$$

The bar above a product term indicates that the product must be performed in the s-domain before the z-transform is taken. In addition, note that

$$\overline{G_1G_2}(z) \neq G_1(z)G_2(z)$$

that is, the z-transform of a product of functions *is not equal* to the product of the z-transforms of the functions

For Figure 4-3(c):

$$C(s) = G_2(s)A^*(s) = G_2(s)\overline{G_1E^*}(s)$$

Thus,

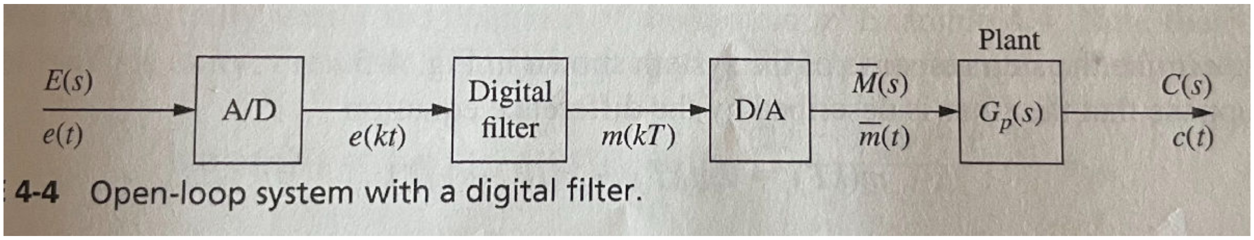
$$C(z) = G_2(z)\overline{G_1E}(z)$$

This transfer function cannot be written, because we cannot factor $E(z)$ from $\overline{G_1E}(z)$

- In general, if the input to a sample-data system is applied directly to a CT part of the system before being sampled, the z-transform of the output of the system cannot be expressed as a function of the z-transform of the input signal!!

4.4 Open-Loop Systems Containing Digital Filters

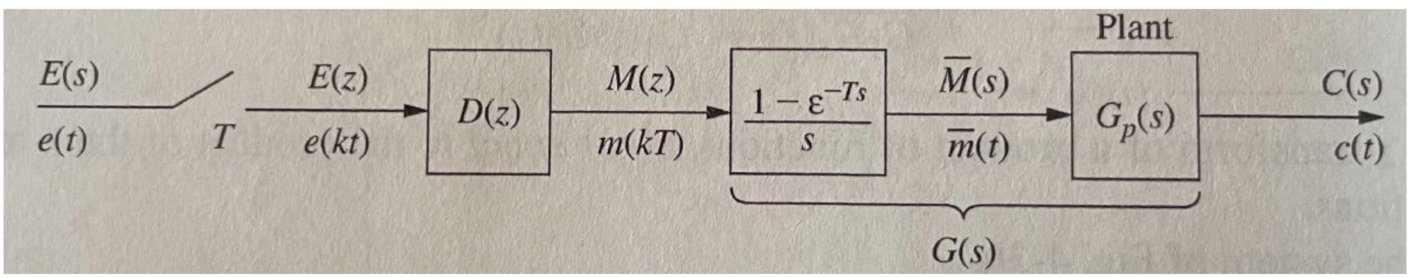
This section examines Open-Loop sampled-data systems that contain digital filters



4-4 Open-loop system with a digital filter.

In the above Figure,

- A/D converter on the filter input converts CT signal $e(t)$ into discrete-sequence $\{e(\mathrm{kT})\}$
- Digital filter processes $\{e(\mathrm{kT})\}$ and generates the output sequence $\{m(\mathrm{kT})\}$
- Then output sequence is converted back to a CT signal - $\overline{m}(t)$ by the D/A converter



$$M(z) = D(z)E(z)$$

or, through the substitution $z = \varepsilon^{\mathrm{sT}}$

$$M^*(s) = D^*(s)E^*(s)$$

The Laplace transform of the signal $\overline{m}(t)$ can be expressed as

$$\overline{M}(s) = \frac{1-\varepsilon^{-\mathrm{Ts}}}{s}M^*(s)$$

Hence,

$$C(s) = G_p(s)\overline{M}(s) = G_p(s)\frac{1-\varepsilon^{-\mathrm{Ts}}}{s}M^*(s)$$

Then,

$$C(s) = G_p(s)\frac{1-\varepsilon^{-\mathrm{Ts}}}{s}D(z)|_{z=\varepsilon^{s\mathrm{T}}}E^*(s)$$

and we see that the filter and associated A/D and D/A converters can be represented

$$C(z) = z\left[G_p(s)\frac{1-\varepsilon^{-\mathrm{Ts}}}{s}\right]D(z)E(z) = G(z)D(z)E(z)$$

Hence the complete model *must be used* as depicted in Fig 4-5 above; that is, the combination of an ideal sampler, $D(z)$, and a zero-order hold does accurately model the combination of the A/D, digital filter, and D/A

4.5 The Modified z-Transform

The concepts of open-loop systems that contain digital filters does not provide analysis techniques towards systems containing ideal time delays

To analyze systems of this type, it is necessary to define the z-transform of a delayed time function $\rightarrow \text{modified } z - \text{transform}$

- This technique can be developed by considering a time function $e(t)$ that is delayed by an amount $\Delta T,\ 0 < \Delta \leq 1,$ by considering $e(t-\Delta T)u(t-\Delta T)$

$$z\{e(t-\Delta T)u(t-\Delta T)\} = z\{E(s)\varepsilon^{-\Delta\mathrm{Ts}}\} = \sum_{n=1}^{\infty} e(\mathrm{nT}-\Delta T)z^{-n}$$

Note that the sampling is not delayed; The above version is called the <u>delayed z-transform</u>, and by definition the delayed z-transform of $e(t)$ is

$$E(z,\Delta) = z\{e(t-\Delta T)u(t-\Delta T)\} = z\{E(s)\varepsilon^{-\Delta\mathrm{Ts}}\}$$

Example 4.5

Find $E(z,\Delta)$, is $\Delta = 0.4$, for $e(t) = \varepsilon^{-\mathrm{at}}u(t)$

$$E(z,\Delta) = \varepsilon^{-0.6\mathrm{aT}}z^{-1} + \varepsilon^{-1.6\mathrm{aT}}z^{-2} + \varepsilon^{-2.6\mathrm{aT}}z^{-3} + \ldots$$

$$= \varepsilon^{-0.6\mathrm{aT}}z^{-1}[1 + \varepsilon^{-\mathrm{aT}}z^{-1} + \varepsilon^{-2\mathrm{aT}}z^{-2} + \ldots]$$

$$= \frac{\varepsilon^{-0.6\mathrm{aT}}z^{-1}}{1-\varepsilon^{-\mathrm{aT}}z^{-1}} = \frac{\varepsilon^{-0.6\mathrm{aT}}}{z-\varepsilon^{-\mathrm{aT}}}$$

The modified z-transform is defined from the delayed z-transform. By definition, the modified z-transform of a function is equal to the delayed z-transform with Δ replaced by $1-m$

with $\Delta = 1-m.$ Two properties of the modified z-transform are

$$E(z,1) = E(z,m)|_{m=1} = E(z) - e(0)$$

and

$$E(z,0) = E(z,m)|_{m=0} = z^{-1}E(z)$$

The value of $m = 1$ denotes no delay [but $e(0)$ does not appear], and the value of $m = 0$ denotes a delay of one sample period

The previous equations and transform tables do not apply to the modified z-transform, instead we derive new methods to do so:

$$E(z,m) = z\{E(s)\varepsilon^{-\Delta \mathrm{Ts}}\}|_{\Delta=1-m} = z\{E(s)\varepsilon^{-(1-m)\mathrm{Ts}}\} = z^{-1}z\{E(s)\varepsilon^{\mathrm{mTs}}\}$$

$$E(z,m) = z^{-1}\sum_{\text{poles of } E(\lambda)} \text{residues of } E(\lambda)\varepsilon^{\mathrm{mT}\lambda}\frac{1}{1-z^{-1}\varepsilon^{T\lambda}}$$

Also,

$$E^{*}(s,m) = \frac{1}{T}\sum_{n=-\infty}^{\infty} E(s+\mathrm{jn}\omega_s)\varepsilon^{-(1-m)\left(s+\mathrm{jn}\omega_s\right)T}$$

Properties of Modified z-Transform

Shifting theorem:

$$z_m\{E(s)\} = E(z,m) = z_m\{\varepsilon^{-\Delta \mathrm{Ts}}E(s)\}|_{\Delta=1-m}$$

Then, by the shifting theorem, for k a positive integer,

$$z_m\{\varepsilon^{-\mathrm{kTs}}E(s)\} = z^{-k}z_m\{E(s)\} = z^{-k}E(z,m)$$

Example 4.7

Wish to find the modified z-transform of the function $e(t) = t \to E(s) = 1/s^2$. This function has pole of order 2 at $s = 0$. Therefore the modified z-transform can be obtained as

$$E(z,m) = z^{-1}\left[\frac{d}{d\lambda}\left[\frac{\varepsilon^{\mathrm{mT}\lambda}}{1-z^{-1}\varepsilon^{T\lambda}}\right]_{\lambda=0}\right]$$

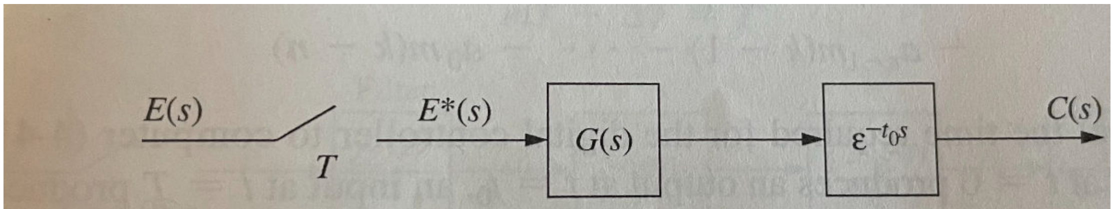
$$= z^{-1}\left[\frac{(1-z^{-1}\varepsilon^{T\lambda})\mathrm{mT}\varepsilon^{\mathrm{mT}\lambda} - \varepsilon^{\mathrm{mT}\lambda}(-\mathrm{T}z^{-1}\varepsilon^{T\lambda})}{(1-z^{-1}\varepsilon^{T\lambda})^2}\right]_{\lambda=0}$$

$$= z^{-1}\left[\frac{\mathrm{mT}(1-z^{-1}) + \mathrm{T}z^{-1}}{(1-z^{-1})^2}\right]$$

$$= \frac{\mathrm{mT}(z-1)+T}{(z-1)^2}$$

4.6 Systems with Time Delays

The modified z-transform may be used to deteremine the pulse transfer functions of discrete-time systems containing ideal time delays



System with ideal time delay.

$$C(s) = G(s)\varepsilon^{-t_0 s}E^*(s)$$

Thus

$$C(z) = z\left\{G(s)\varepsilon^{-t_0 s}\right\}E(z)$$

If we now let

$$t_0 = \mathrm{kT} + \Delta T, \qquad 0 < \Delta < 1$$

where k is a positive integer, then

$$C(z) = z^{-k}z\{G(s)\varepsilon^{-\Delta \mathrm{Ts}}\}E(z) = z^{-k}G(z,m)E(z)$$

where $m = 1 - \Delta$

Example 4.8

In Fig 4-8, let the input be a unit step, $t_0 = 0.4T$, and

$$G(s) = \frac{1-\varepsilon^{-\mathrm{Ts}}}{s(s+1)}$$

This system was considered in Eample 4.3 with no delay. From the shifting theorem and modified z-transform tables,

$$G(z,m) = z_m\left\{\frac{1-\varepsilon^{-\mathrm{Ts}}}{s(s+1)}\right\} = (1-z^{-1})z_m\left\{\frac{1}{s(s+1)}\right\}$$

$$= \frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-\mathrm{mT}}) + \varepsilon^{-\mathrm{mT}} - \varepsilon^{-T}}{(z-1)(z-\varepsilon^{-T})}\right]$$

Thus, since $\mathrm{mT} = T - \Delta T = 0.6T$,

$$G(z,m) = \frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-0.6T}) + \varepsilon^{-0.6T} - \varepsilon^{-T}}{(z-1)(z-\varepsilon^{-T})}\right]$$

Since $k = 0$, $C(z)$ is seen to be

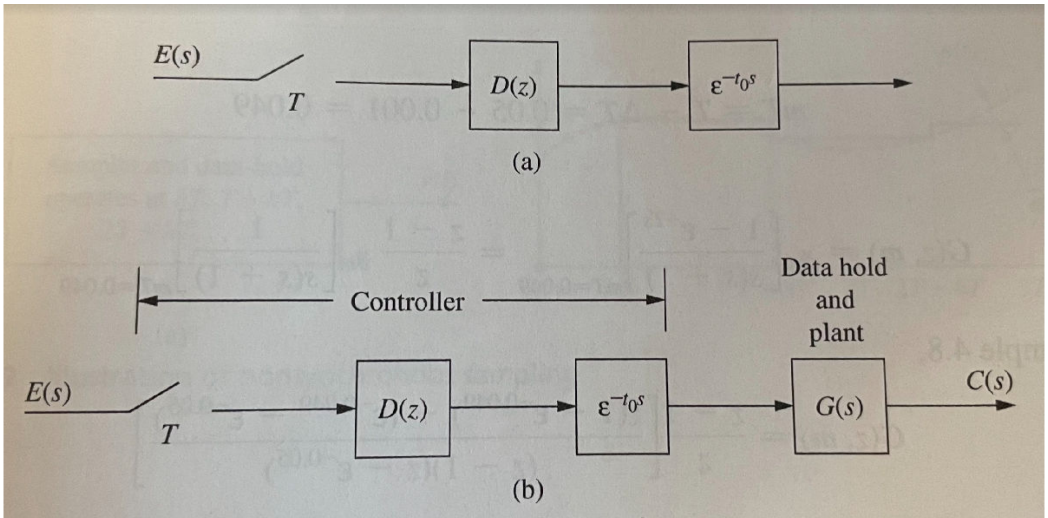
$$C(z) = G(z,m)\frac{z}{z-1} = \frac{z(1-\varepsilon^{-0.6T}) + \varepsilon^{-0.6T} - \varepsilon^{-T}}{(z-1)(z-\varepsilon^{-T})}$$

By the power-series method of Section 2.6,

$$C(z) = (1 - \varepsilon^{-0.6T})z^{-1} + (1 - \varepsilon^{-1.6T})z^{-2} + (1 - \varepsilon^{-2.6T})z^{-3} + \ldots$$

From Example 4.3, the response of this system with no delay is $c(\mathrm{nT}) = 1 - \varepsilon^{-\mathrm{nT}}$. This response delayed by $0.4T$ is then

$$c(\mathrm{nT})|_{n \leftarrow (n-0.4)} = 1 - \varepsilon^{-(n-0.4)T}, \quad n \geq 1$$



Digital controller with nonzero computation time.

The modified z-transform may also be used to determine the pulse transfer functions of digital control systems in which the computation time of the digital computer cannot be neglected

Let the time required for the digital controller to commputer be t_0 seconds.

- Thus an input at $t = 0$ produces an output at $t = t_0$
- An input at $t = T$ produces an output at $t = T + t_0$, and so on

Hence, the digital controller may be modeled as a digital controller without time delay, followed by an ideal time delay of t_0 seconds, as shown in Fig 4-9(a) above. An open-loop system containing this controller may be modeled as shown in Fig 4-9(b),

$$C(z) = z\left[G(s)\varepsilon^{-t_0 s}\right]D(z)E(z)$$

If we let

$$t_0 = \mathrm{kT} + \Delta T, \quad 0 < \Delta < 1$$

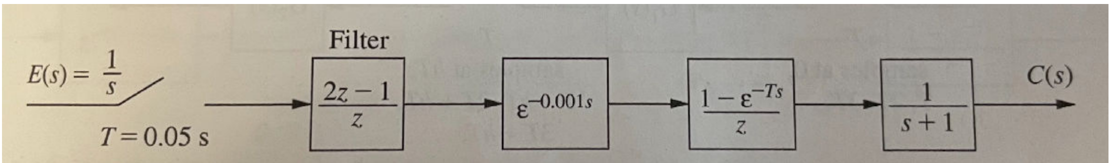
with k being a positive integer, then we obtain

$$C(z) = z^{-k}G(z, m)D(z)E(z)$$

where $m = 1 - \Delta$

Example 4.9

Consider the system of Fig. 4-10



4-10 System for Example 4.9.

This system is that of Example 4.4, with a computational delay added for the filter. The delay is $1 \text{ ms} \left(t_0 = 10^{-3}s\right)$ and $T = 0.05s$. Thus, for this system,

$$D(z) = \frac{2z-1}{z}$$

Now

$$\text{mT} + \Delta T = T$$

or

$$\text{mT} = T - \Delta T = 0.05 - 0.001 = 0.049$$

Then

$$G(z,m) = z_m\left\{\frac{1-\varepsilon^{-\text{Ts}}}{s(s+1)}\right\}_{\text{mT}=0.049} = \frac{z-1}{z} z_m\left\{\frac{1}{s(s+1)}\right\}_{\text{mT}=0.049}$$

From Example 4.8,

$$G(z,m) = \frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-0.049}) + \varepsilon^{-0.049} - \varepsilon^{-0.05}}{(z-1)(z-\varepsilon^{-0.05})}\right]$$

Since the inpput is a unit step, then

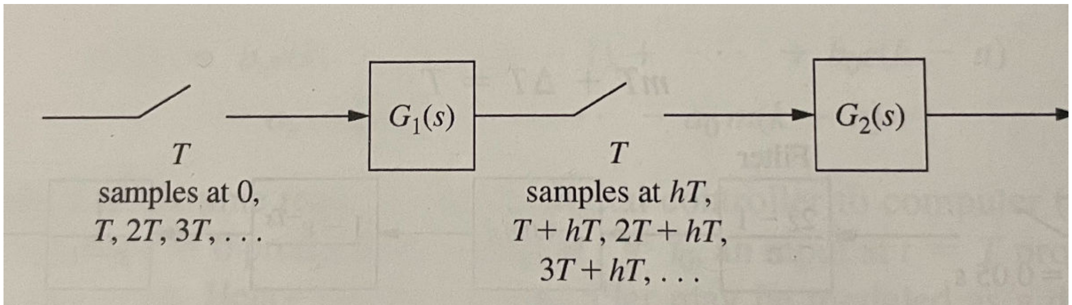
$$C(z) = G(z,m)D(z)E(z)$$

$$= \frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-0.049}) + \varepsilon^{-0.049} - \varepsilon^{-0.05}}{(z-1)(z-\varepsilon^{-0.05})}\right]\left[\frac{2z-1}{z}\right]\left[\frac{z}{z-1}\right]$$

$$= \frac{(2z-1)[z(1-\varepsilon^{-0.049}) + (\varepsilon^{-0.049} - \varepsilon^{-0.05})]}{z(z-1)(z-\varepsilon^{-0.05})}$$

4.7 Nonsynchronous Sampling

In this section open-loop systems with nonsynchronous sampling are analyzed



System with nonsynchronous sampling.

Nonsynchronous sampling can be defined by considering the system of Fig. 4-11, where in this system, both samplers operate at the same rate, but are not synchronous. The output of this system can be derived using the modified z-transform

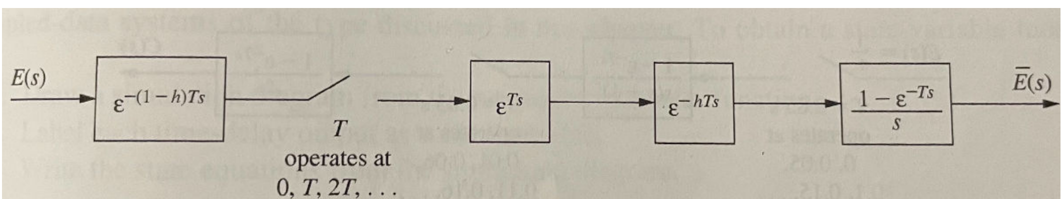
The data-hold output output is $\bar{e}(t)$, that has a modified z-transform of

$$\bar{E}(s) = \frac{1 - \varepsilon^{-\mathrm{T}s}}{s} \varepsilon^{\mathrm{T}s} \varepsilon^{-\mathrm{hT}s} E(z, m)|_{m=h,\, z=\varepsilon^{\mathrm{T}s}}$$

Since

$$E(z, m) = z\{E(s)\varepsilon^{-\Delta \mathrm{T}s}\}|_{\Delta = 1-m}$$

Fig. 4-11 can then be modeled as



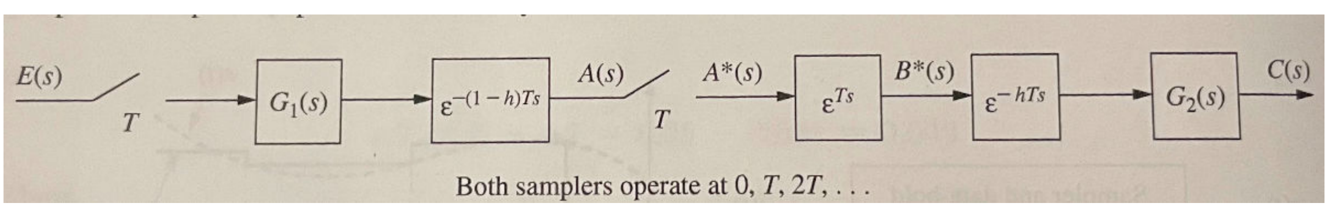
JRE 4-13 Model of a sampler and data hold.

where the sampler operates at $t = 0, T, 2T, \ldots$

Note: this model

- delays the input
- samples the delayed signal
- advances the delayed sampled signal, such that the total delay in the signal is zero

From this development, the model of Fig. 4-11 with nonsynchronous samplers, may be modeled as shown below in Fig. 4-14



The figure above has samplers that are synchronous and now

$$A(s) = \varepsilon^{-(1-h)\mathrm{Ts}} G_1(s) E^*(s)$$

and thus

$$A(z) = G_1(z, m)|_{m=h} E(z)$$

Also,

$$B^*(s) = \varepsilon^{\mathrm{Ts}} A^*(s)$$

yielding

$$B(z) = \mathrm{zA}(z) = \mathrm{zE}(z) G_1(z, m)|_{m=h}$$

Then

$$C(s) = \varepsilon^{-\mathrm{hTs}} G_2(s) B^*(s)$$

yielding

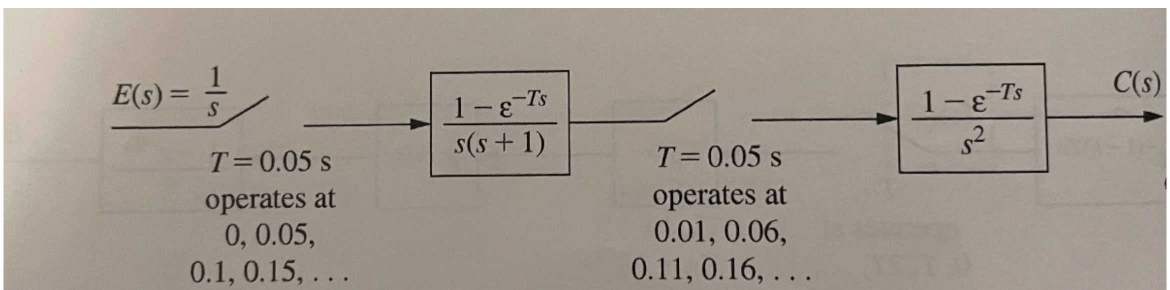
$$C(z) = G_2(z, m)|_{m=1-h} B(z)$$

From these equations, we find $C(z)$ to be given by

$$C(z) = \mathrm{zE}(z) G_1(z, m)|_{m=h} G_2(z, m)|_{m=1-h}$$

Example 4.10

We wish to find $C(z)$ for the system below, which contains nonsynchronous sampling



4-15 System for Example 4.10.

Now,

$$E(z) = z\left\{\frac{1}{s}\right\} = \frac{z}{z-1}$$

For the system, $T = 0.05$ and $\mathrm{hT} = 0.01$. From Example 4.8,

$$G_1(z, m) = z_m\left\{\frac{1 - \varepsilon^{-\mathrm{Ts}}}{s(s+1)}\right\} = \frac{z-1}{z}\left[\frac{z(1 - \varepsilon^{-\mathrm{mT}}) + \varepsilon^{-\mathrm{mT}} - \varepsilon^{-T}}{(z-1)(z - \varepsilon^{-T})}\right]$$

Then

$$G_1(z,m) = \frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-0.01}) + \varepsilon^{-0.01} - \varepsilon^{-0.05}}{(z-1)(z-\varepsilon^{-0.05})}\right]$$

Also, from the modified z-transform tables,

$$G_2(z,m) = z_m\left\{\frac{1-\varepsilon^{-\mathrm{Ts}}}{s^2}\right\} = \frac{z-1}{z}\left[\frac{\mathrm{mTz} - \mathrm{mT} + T}{(z-1)^2}\right]$$

or

$$G_2(z,m)|_{m=1-h} = \frac{0.04z + 0.01}{z(z-1)}$$

Then, from the development above

$$C(z) = z\left[\frac{z}{z-1}\right]\frac{z-1}{z}\left[\frac{z(1-\varepsilon^{-0.01}) + \varepsilon^{-0.01} - \varepsilon^{-0.05}}{(z-1)(z-\varepsilon^{-0.05})}\right]\frac{0.04z + 0.01}{z(z-1)}$$

$$= \frac{(0.04z + 0.01)[z(1-\varepsilon^{-0.01}) + \varepsilon^{-0.01} - \varepsilon^{-0.05}]}{(z-1)^2(z-\varepsilon^{-0.05})}$$

4.8 State-Variable Models

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k)$$

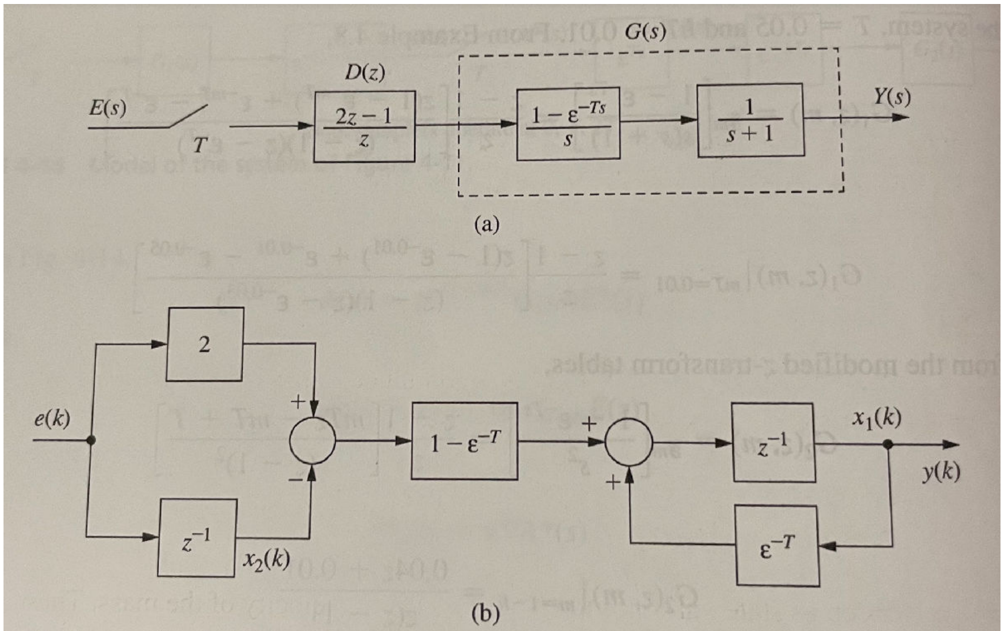
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k)$$

To obtain a state-variable model:

1. Draw a simulation diagram from the z-transform transfer function
2. Label each time-delay output as a state variable
3. Write the state equations from the simulation diagram

Example 4.11

Consider the system of Fig. 4-16(a)



-16 System for Example 4.11.

Here, we denote the output as $Y(s)$ to prevent any notational confusion with the $\mathbf{C}$ matrix.

$$G(z) = z\left\{\frac{1-\varepsilon^{-\mathrm{Ts}}}{s(s+1)}\right\} = \frac{1-\varepsilon^{-T}}{z-\varepsilon^{-T}}$$

ands as shown in Fig. 4-16(a)

$$D(z) = \frac{2z-1}{z}$$

A simulation diargram is given in Fig. 4-16(b). Next each delay output is labeled as a state variable. Then, from this Fig, we write

$$\mathbf{x}(k+1) = \begin{bmatrix} \varepsilon^{-T} & -1+\varepsilon^{-T} \\ 0 & 0 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 2(1-\varepsilon^{-T}) \\ 1 \end{bmatrix} e(k)$$

$$y(k) = [1 \quad 0]\,\mathbf{x}(k)$$

4.9 Review of Continuous-Time State Variables

Two main disadvantages of modeling system from linear time-variant discrete-time system

1. In taking transfer-function approach to discrete state modeling, we have difficulty in choosing velocity as 2nd state variable, Thus we lose the natural, and desirable, states of the system
2. Difficulty in deriving the pulse transfer functions for higher-order systems

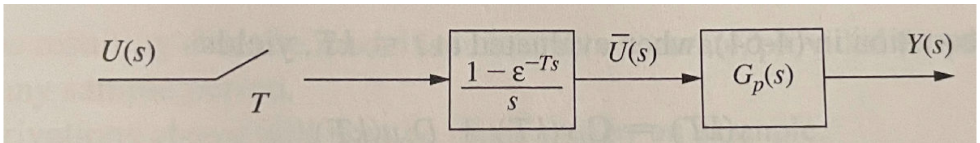
$$\dot{\mathbf{v}}(t) = \mathbf{A}_c\mathbf{v}(t) + \mathbf{B}_c\mathbf{u}(t)$$

$$\mathbf{y}(t) = \mathbf{C}_c\mathbf{v}(t) + \mathbf{D}_c\mathbf{u}(t)$$

4.10 Discrete-Time State Equations

In this section a technique is developed for determining the discrete time equations of a sampled-data system directly from the continuous-time equations. The states of the continuous-time model become the states of the discrete model. Thus the natural states of the system are preserved.

To develop this technique, consider the state equations for the continuous-time portion of the system



Sampled-data system.

$$\dot{\mathbf{v}}(t) = \mathbf{A}_c\mathbf{v}(t) + \mathbf{B}_c\mathbf{u}(t)$$

$$\mathbf{y}(t) = \mathbf{C}_c\mathbf{v}(t) + \mathbf{D}_c\mathbf{u}(t)$$

As shown in Section 4.9, the solution to these equations is

$$\mathbf{v}(t) = \mathbf{\Phi}_c(t - t_0)\mathbf{v}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}_c(t - \tau)\mathbf{B}_c y(\tau)d\tau$$

where the initial time is t_0, and where,

$$\mathbf{\Phi}_c(t - t_0) = \sum_{k=0}^{\infty} \frac{\mathbf{A}_c^k(t - t_0)^k}{k!}$$

To obtain the discrete model we evaluate the previous equation at $t = \mathrm{kT} + T$ with $t_0 = \mathrm{kT}$, that is,

$$\mathbf{v}(kT + T) = \mathbf{\Phi}_c(T)\mathbf{v}(kT) + u(kT)\int_{kT}^{kT+T} \mathbf{\Phi}_c(kT + T - \tau)\mathbf{B}_c d\tau$$

Note that we have replace $u(t)$ with $u(\mathrm{kT})$. It is emphasized that this development is valid only if $u(t)$ is the output of a *zero-order hold.*

Compare with the discrete state equations

$$\mathbf{x}(k + 1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k)$$

Thus, if we let

$$\mathbf{x}(kT) = \mathbf{v}(kT)$$

$$\mathbf{A} = \mathbf{\Phi}_c(T)$$

$$\mathbf{B} = \int_{kT}^{kT+T} \mathbf{\Phi}_c(kT + T - \tau)\mathbf{B}_c d\tau$$

we obtain the discrete state equations for the sampled-data system. Hence the discrete-system $\mathbf{A}$ and $\mathbf{B}$ matrices are give by the above equations

The output equation, when evaluated at $t = \mathrm{kT}$, yields

$$y(kT) = \mathbf{C}_c\mathbf{v}(kT) + D_c u(kT)$$

$$y(kT) = \mathbf{C}_c\mathbf{x}(kT) + D_c u(kT)$$

Thus the discrete $\mathbf{C}$ and D matrices are equal to the continuous-time $\mathbf{C}_c$ and D_c matrices, respectively.

The relationship for $\mathbf{B}$ can be simplified, let $kT - \tau = -\sigma$

$$\mathbf{B} = \left[\int_0^T \mathbf{\Phi}_c(T-\sigma)d\sigma\right]\mathbf{B}_c$$

Example 4.13

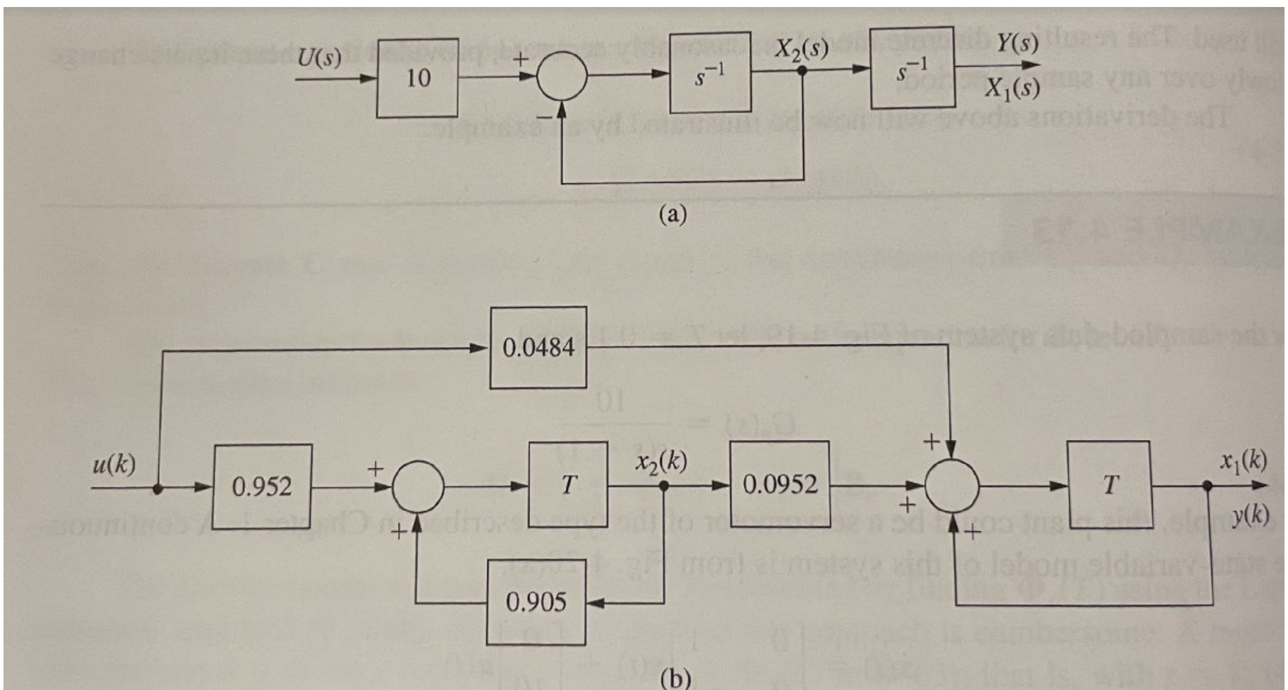


FIGURE 4-20 Simulation diagram for the system of Example 4.13.

For the sampled-data system of Fig. 4-19, let $T = 0.1s$ and

$$G_p(s) = \frac{10}{s(s+1)}$$

A continous-time state-variable model of this system is from Fig. 4-20(a)

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}\mathbf{x}(k) + \begin{bmatrix} 0 \\ 10 \end{bmatrix}\mathbf{u}(t)$$

$$y(t) = [1 \quad 0]\,\mathbf{x}(k)$$

For this example, since the system is second order, $\mathbf{\Phi}_c(t)$ will be found.

$$\mathbf{\Phi}_c(t) = L^{-1}\left[[s\mathbf{I} - \mathbf{A}_c]^{-1}\right]$$

$$= L^{-1}\begin{bmatrix} s & -1 \\ 0 & s+1 \end{bmatrix}^{-1} = L^{-1}\begin{bmatrix} \frac{1}{s} & \frac{1}{s(s+1)} \\ 0 & \frac{1}{s+1} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1-\varepsilon^{-t} \\ 0 & \varepsilon^{-t} \end{bmatrix}$$

Also,

$$\int_0^T \mathbf{\Phi}_c(\tau)d\tau = \begin{bmatrix} \tau & \tau+\varepsilon^{-\tau} \\ 0 & -\varepsilon^{-\tau} \end{bmatrix}_0^T = \begin{bmatrix} T & T-1+\varepsilon^{-T} \\ 0 & 1-\varepsilon^{-T} \end{bmatrix}$$

Then

$$\mathbf{A} = \mathbf{\Phi}_c(T)|_{T=0.1} = \begin{bmatrix} 1 & 0.0952 \\ 0 & 0.905 \end{bmatrix}$$

and

$$\mathbf{B} = \left[\int_0^T \mathbf{\Phi}_c(T-\sigma)d\sigma\right]\mathbf{B}_c = \begin{bmatrix} 0.1 & 0.00484 \\ 0 & 0.0952 \end{bmatrix}\begin{bmatrix} 0 \\ 10 \end{bmatrix}$$

$$= \begin{bmatrix} 0.0484 \\ 0.952 \end{bmatrix}$$

Hence the discrete state equations are

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & 0.0952 \\ 0 & 0.905 \end{bmatrix}\mathbf{x}(k) + \begin{bmatrix} 0.0484 \\ 0.952 \end{bmatrix}u(k)$$

$$y(t) = [1 \quad 0]\,\mathbf{x}(k)$$

A simulation diagram of this model is shown in Fig. 4-20(b). Fig. 4-20(a) is the simulation diagram of the analog plant. Even though the states, the input, and the output of the two diagrams in Fig. 4-20 are equal at the sampling instants, the two diagrams bear no resemblance to each other. In general, the two simulation diagrams for such a system are not similar

4.11 Practical Calculations

All calculations required to develop the discrete model of an analog plant may be performed by computer. Calculatoin by computer is required for high-order systems, and is preferred for low-order systems to reduce errors.

List of necessary steps for the calculations

1. Derive the state model for the analog part of the system, in the form

$$\dot{\mathbf{x}}(t) = \mathbf{A}_c\mathbf{x}(t) + \mathbf{B}_c\mathbf{u}(t)$$

$$\mathbf{y}(t) = \mathbf{C}_c\mathbf{x}(t) + \mathbf{D}_c\mathbf{u}(t)$$

2. If the transfer function of the analog part of the system is required, calculate

$$G_p(s) = \mathbf{C}_c[s\mathbf{I} - \mathbf{A}_c]^{-1}\mathbf{B}_c + D_c$$

3. Calculate the discrete matrices of the analog part of the system

$$\mathbf{A} = \mathbf{I} + \mathbf{A}_cT + \mathbf{A}_c^2T^2/2 + \ldots$$

$$\mathbf{B} = (\mathbf{I}T + \mathbf{A}_cT^2/2! + \mathbf{A}_c^2T^3/3! + \ldots)\mathbf{B}_c$$

$$\mathbf{C} = \mathbf{C}_c, \qquad D = D_c$$

4. Calculate the pulse transfer function using

$$G(z) = \mathbf{C}[z\mathbf{I} - \mathbf{A}]^{-1}\mathbf{B} + \mathbf{D}$$

The calculations required in steps 2, 3, and 4 may be implemented in MATLAB by the statements

- [numc,denc] = ss2tf(Ac,Bc,Cc,Dc)
- [A,B] = c2d(Ac,Bc,T)
- [numz,denz] = ss2tf(A,B,C,D)

In these statements, numc are the numerator-polynomial coefficients of $G_p(s)$, and denc aere those of the denominator.

Note that these procedures give the analog transfer function, the discrete model, and the pulse transfer function. We do not directly use the z-transform tables in any of the steps; all steps are performed on the computer. However, we must, by some procedure, derive the analog state model. An example is given below

Example 4.14

Consider again the system of Example 4.13. A MATLAB program that calculates the discrete state mat4rices A, B, C, and D, and the plant transfer function $G(z)$ = numz/denz is given by

```
Ac = [0 1; 0 -1];
Bc = [0; 10];
C = [1 0];
D = 0;
T = 0.1;
[A,B] = c2d(Ac,Bc,T)
```

```
A = 2×2
    1.0000    0.0952
         0    0.9048
B = 2×1
    0.0484
    0.9516
```

```
[numz,denz] = ss2tf(A,B,C,D)
```

```
numz = 1×3
         0    0.0484    0.0468
denz = 1×3
    1.0000   -1.9048    0.9048
```

```
Gz = tf(numz,denz,T)
```

```
Gz =
 
   0.04837 z + 0.04679
  ----------------------
  z^2 - 1.905 z + 0.9048
 
Sample time: 0.1 seconds
Discrete-time transfer function.
```

4.12 Summary

In this chapter we examined various aspects of the open-loop discrete-time systems

- Starred transform and showed it possesses the properties of the z-transform
- Starred transform was used to find the pulse transfer function of open-loop systems
- Pulse transfer function was extended to the analysis of oopen-loop systems containing digital filters
- The modified z-transform and its properties were derived in order to analyze systems containing ideal time delays
- Techniques for finding discrete state-variable models of open-loop sampled-data systems were presented
- These developments form the basis for the computer calculations of discrete state models and the pulse transfer function
- The foundation built in this chapter will serve as a basis for presenting the analysis of closed-loop discrete-time systems in Chapter 5