ΨΗΦΙΑΚΟΣ ΕΛΕΓΧΟΣ

(22Δ802)

Β΄ ΕΞΑΜΗΝΟ 2015-16

State Space Modeling - Analog to Digital

Καθηγητής Πέτρος Π. Γρουμπός

☎ 2610 996 449

Ώρες Γραφείου: Τετάρτη Πέμπτη Παρασκευή 11:00-12:00

Γραφείο: 1ος όροφος

Τομέας Συστημάτων & Αυτομάτου Ελέγχου

Τμήμα ΗΜ&ΤΥ

Contents

State Equations for Analog/Digital Systems

1 Continuous Time State Equations

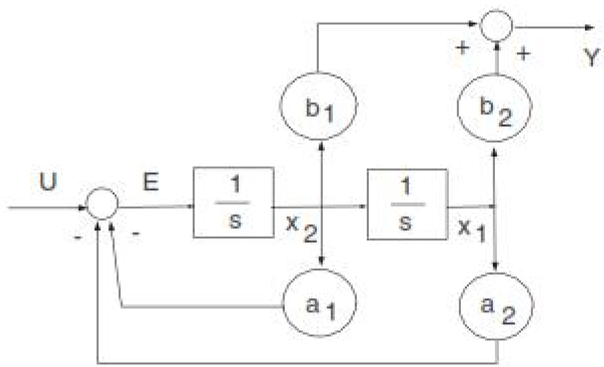
Consider a second order transfer function

$$\begin{aligned}\frac{Y(s)}{U(s)} &= \frac{b_1 s + b_2}{s^2 + a_1 s + a_2} \\ &= \frac{b_1 \frac{1}{s} + b_2 \frac{1}{s^2}}{1 + a_1 \frac{1}{s} + a_2 \frac{1}{s^2}} \\ &= \frac{b_1 \frac{1}{s} + b_2 \frac{1}{s^2}}{1 + a_1 \frac{1}{s} + a_2 \frac{1}{s^2}} \frac{E(s)}{E(s)}\end{aligned}$$

Two equations derived from the above yield a block diagram of canonical structure.

$$Y(s) = (b_1 \frac{1}{s} + b_2 \frac{1}{s^2}) E(s)$$

$$E(s) = U(s) - (a_1 \frac{1}{s} + a_2 \frac{1}{s^2}) E(s)$$



Using the state variables $x_1(t)$ and $x_2(t)$ shown in the block diagram, we can write

$$\begin{aligned} sX_1(s) &= X_2(s) \\ sX_2(s) &= E(s) = -a_2X_1(s) - a_1X_2(s) + U(s) \\ Y(s) &= b_2X_1(s) + b_1X_2(s) \end{aligned}$$

from which we can write a set of simultaneous first order differential equations.

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -a_2x_1(t) - a_1x_2(t) + u(t) \\ y(t) &= b_2x_1(t) + b_1x_2(t) \end{aligned}$$

or $$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

Example: Consider a 2nd order transfer function,

$$\frac{Y(s)}{U(s)} = \frac{1}{s(s+a)}$$

This is the case of $a_1 = a,\ a_2 = 0$ and $b_1 = 0,\ b_2 = 1$. Using $[x_1, x_2]$ as state variables, we have,

$$\left.\begin{aligned} sX_1(s) &= X_2(s) \\ sX_2(s) &= -aX_2(s) + U(s) \end{aligned}\right\}$$

We can write the state equation,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -a \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

The state equation is written in a vector form as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t),$$

along with an output equation (in case of single input, single output)

$$y(t) = \mathbf{c}\mathbf{x}(t) + du(t).$$

- Exercise: Write the continuous state equation and the output equation for a system given by

$$E(s) = \frac{s+3}{(s+1)(s+2)}$$

2 Solution of State Equation

The Laplace transform of the state equation is

$$\begin{aligned} s\mathbf{x}(s) - \mathbf{x}(0) &= \mathbf{A}\mathbf{x}(s) + \mathbf{b}u(s) \\ s\mathbf{I}\mathbf{x}(s) - \mathbf{A}\mathbf{x}(s) &= \mathbf{x}(0) + \mathbf{b}u(s) \\ \mathbf{x}(s) &= (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}(0) + (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}u(s) \end{aligned}$$

Define the state transition matrix, $\mathbf{\Phi}(t)$ as

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

The inverse Laplace transform of $\mathbf{x}(s)$ yields time domain solution of the state vector $\mathbf{x}(t)$.

$$\mathbf{x}(t) = \mathbf{\Phi}(t)\mathbf{x}(0) + \int_0^t \mathbf{\Phi}(t-\tau)\mathbf{b}u(\tau)\, d\tau$$

The output is

$$y(t) = \mathbf{cx}(t)$$

Example: Find the state transition matrix for the next state equation,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -a \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

recalling the matrix inversion for 2×2 matrices,

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -c \\ -b & a \end{bmatrix}^{T} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

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

$$
\begin{aligned}
&= \mathcal{L}^{-1}\left\{\left(s\mathbf{I} - \begin{bmatrix} 0 & 1 \\ 0 & -a \end{bmatrix}\right)^{-1}\right\} \\
&= \mathcal{L}^{-1}\left\{\begin{bmatrix} s & -1 \\ 0 & s+a \end{bmatrix}^{-1}\right\} \\
&= \mathcal{L}^{-1}\left\{\frac{1}{s(s+a)}\begin{bmatrix} s+a & 1 \\ 0 & s \end{bmatrix}\right\} \\
&= \mathcal{L}^{-1}\left\{\begin{bmatrix} \frac{1}{s} & \frac{1}{s(s+a)} \\ 0 & \frac{1}{s+a} \end{bmatrix}\right\} \\
&= \begin{bmatrix} 1 & \frac{1}{a}(1-e^{-at}) \\ 0 & e^{-at} \end{bmatrix}
\end{aligned}
$$

- Exercise: Based on the state equation for

$$E(s) = \frac{s+3}{(s+1)(s+2)}$$

developed in the previous exercise, find the state transition matrix, $\mathbf{\Phi}(t)$.

3 Discrete Time State Equation

Consider transition of the continuous state $\mathbf{x}(t)$ from $t = Tk$ to $t = T(k+1)$ in the state equation,

$$\mathbf{x}(t) = \mathbf{\Phi}(t - t_0)\mathbf{x}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}(t - \tau)\mathbf{b}u(\tau)\, d\tau.$$

$$\mathbf{x}(T(k+1)) = \mathbf{\Phi}(T)\mathbf{x}(Tk) + \int_{Tk}^{T(k+1)} \mathbf{\Phi}(T(k+1) - \tau)\mathbf{b}u(\tau)\, d\tau.$$

Let us simplify this equation.

$$\begin{aligned}
\mathbf{x}(T(k+1)) &= \mathbf{\Phi}(T)\mathbf{x}(Tk) + \int_{Tk}^{T(k+1)} \mathbf{\Phi}(T(k+1) - \tau)\mathbf{b}u(\tau)\, d\tau \\
&= \mathbf{\Phi}(T)\mathbf{x}(Tk) + u(Tk)\int_{Tk}^{T(k+1)} \mathbf{\Phi}(T(k+1) - \tau)\mathbf{b}\, d\tau \\
& \quad T(k+1) > \tau > Tk
\end{aligned}$$

$$
\begin{aligned}
&\Rightarrow \quad -T(k+1) < -\tau < -Tk \\
&\Rightarrow \quad 0 < T(k+1) - \tau < T \\
&\Rightarrow \quad \text{Let} \;\; \tau' = T(k+1) - \tau \\
&\Rightarrow \quad \text{then} \;\; d\tau = -d\tau' \\
&= \quad \mathbf{\Phi}(T)\mathbf{x}(Tk) + u(Tk) \int_T^0 -\mathbf{\Phi}(\tau')\mathbf{b}\, d\tau' \\
&= \quad \mathbf{\Phi}(T)\mathbf{x}(Tk) + u(Tk) \int_0^T \mathbf{\Phi}(\tau)\mathbf{b}\, d\tau \\
&= \quad \mathbf{P}\mathbf{x}(Tk) + \mathbf{q}u(Tk)
\end{aligned}
$$

Thus,

$$
\mathbf{P} = \mathbf{\Phi}(T) \;\; \text{and} \;\; \mathbf{q} = \int_0^T \mathbf{\Phi}(\tau)\mathbf{b}\, d\tau
$$

It is important to notice that $u(Tk)$ was held constant over the integration period of $T(k+1) > \tau > Tk$. This means that ZOH was taken into consideration for input $u(Tk)$.

For the previous example:

$$\mathbf{P} = \mathbf{\Phi}(T) = \begin{bmatrix} 1 & \frac{1}{a}(1 - e^{-aT}) \\ 0 & e^{-aT} \end{bmatrix}$$

$$\begin{aligned} \mathbf{q} &= \int_0^T \begin{bmatrix} 1 & \frac{1}{a}(1 - e^{-a\tau}) \\ 0 & e^{-at}\tau \end{bmatrix} \begin{bmatrix} 0 \\ K \end{bmatrix} d\tau \\ &= \begin{bmatrix} \frac{K}{a}(T + \frac{1}{a}e^{-aT} - \frac{1}{a}) \\ \frac{K}{a}(1 - e^{-aT}) \end{bmatrix} \end{aligned}$$

Properties of the state transition matrix $\mathbf{\Phi}(t)$

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) \quad \Leftrightarrow \quad \dot{x}(t) = ax(t)$$

$$\text{Characteristic Equation} \qquad \lambda - a = 0$$

$$\mathbf{x}(t) = \mathbf{\Phi}(t)\mathbf{x}(0) \quad \Leftrightarrow \quad x(t) = e^{at}x(0)$$

Because of the similarity, we write

$$\mathbf{\Phi}(t) = e^{\mathbf{A}t}.$$

The state transition matrix $\mathbf{\Phi}(t)$ has similar properties as the scaler function e^{at}.

1. $\mathbf{\Phi}(t_1 + t_2) = \mathbf{\Phi}(t_1)\mathbf{\Phi}(t_2) \quad \Leftrightarrow \quad e^{a(t_1+t_2)} = e^{at_1}e^{at_2}$
2. $\frac{d}{dt}\mathbf{\Phi}(t) = \mathbf{A}\mathbf{\Phi}(t) \quad \Leftrightarrow \quad \frac{d}{dt}e^{\mathbf{A}t} = \mathbf{A}e^{\mathbf{A}t} \quad \Leftrightarrow \quad \frac{d}{dt}e^{at} = ae^{at}$

A simple way to Calculate $\mathbf{P}$ and $\mathbf{q}$

Taylor's expansion also applies to $e^{\mathbf{A}t}$ in the similar manner as it applies to e^{at}.

$$f(t) = f(0) + \frac{1}{1!}f'(0)t + \frac{1}{2!}f''(0)t^2 + \cdots$$

$$e^{at} = 1 + \frac{1}{1!}at + \frac{1}{2!}a^2t^2 + \cdots$$

$$e^{\mathbf{A}t} = \mathbf{I} + \frac{1}{1!}\mathbf{A}t + \frac{1}{2!}\mathbf{A}^2t^2 + \cdots$$

$$\begin{aligned} \mathbf{P} &= \mathbf{\Phi}(T) = e^{\mathbf{A}T} \\ &= \mathbf{I} + \frac{1}{1!}\mathbf{A}T + \frac{1}{2!}\mathbf{A}^2T^2 + \frac{1}{3!}\mathbf{A}^3T^3 + \cdots \end{aligned}$$

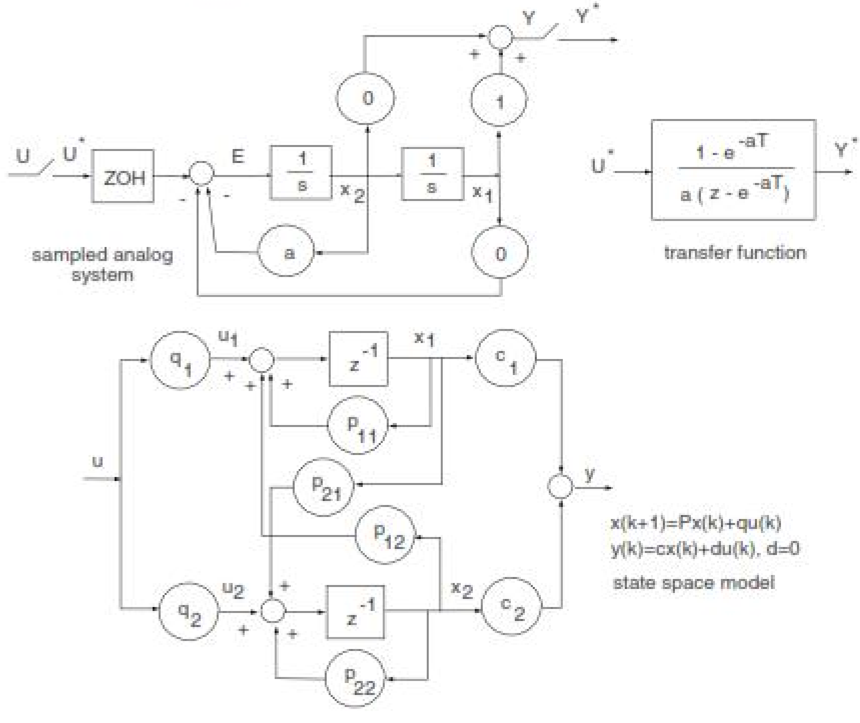
Since higher order terms exponentially get smaller, first 4 to 5 terms is practically sufficient to calculate $\mathbf{P}$.

$$
\begin{aligned}
\mathbf{q} &= \int_0^T \mathbf{\Phi}(\tau)\mathbf{b}\, d\tau \\
&= \int_0^T e^{\mathbf{A}t}\mathbf{b}\, d\tau \\
&= \frac{1}{\mathbf{A}} e^{\mathbf{A}t}\Big|_0^T \mathbf{b} \\
&= \mathbf{A}^{-1}(e^{\mathbf{A}T} - \mathbf{I})\mathbf{b} \\
&= \mathbf{A}^{-1}(\mathbf{I} + \frac{1}{1!}\mathbf{A}T + \frac{1}{2!}\mathbf{A}^2T^2 + \frac{1}{3!}\mathbf{A}^3T^3 + \cdots - \mathbf{I})\mathbf{b} \\
&= \mathbf{A}^{-1}(\frac{1}{1!}\mathbf{A}T + \frac{1}{2!}\mathbf{A}^2T^2 + \frac{1}{3!}\mathbf{A}^3T^3 + \cdots)\mathbf{b} \\
&= (\frac{1}{1!}T + \frac{1}{2!}\mathbf{A}T^2 + \frac{1}{3!}\mathbf{A}^2T^3 + \cdots)\mathbf{b}
\end{aligned}
$$

To maintain the degree of accuracy to be the same as $\mathbf{P}$, use one less terms for $\mathbf{q}$ than $\mathbf{P}$ if the cancelled $\mathbf{I}$ is taken into account.

4 The structure of the state space model

by the previous example



State space model to transfer function

$$\mathbf{x}(T(k+1)) = \mathbf{P}\mathbf{x}(Tk) + \mathbf{q}u(Tk), \quad \text{and} \quad \mathbf{y}(Tk) = \mathbf{c}\mathbf{x}(Tk)$$

$$\text{where,} \quad \mathbf{P} = \mathbf{\Phi}(T), \quad \text{and} \quad \mathbf{q} = \int_0^T \mathbf{\Phi}(\tau)\mathbf{b}\, d\tau$$

Note that the vector $\mathbf{c}$ is a row vector, whereas other vectors are a column vector. The z-transform of the state equation is given by

$$z\mathbf{I}\mathbf{x}(z) - z\mathbf{I}\mathbf{x}(0) = \mathbf{P}\mathbf{x}(z) + \mathbf{q}u(z)$$

$$\begin{aligned}
\mathbf{x}(z) &= (z\mathbf{I} - \mathbf{P})^{-1}\mathbf{q}u(z) \\
y(z) &= \mathbf{c}\mathbf{x}(z) = \mathbf{c}(z\mathbf{I} - \mathbf{P})^{-1}\mathbf{q}u(z) \\
\frac{y(z)}{u(z)} &= \mathbf{c}(z\mathbf{I} - \mathbf{P})^{-1}\mathbf{q}
\end{aligned}$$

5 Calculating the closed-loop disrete time state equation

Method to use analog state equation and $\Phi(t)$

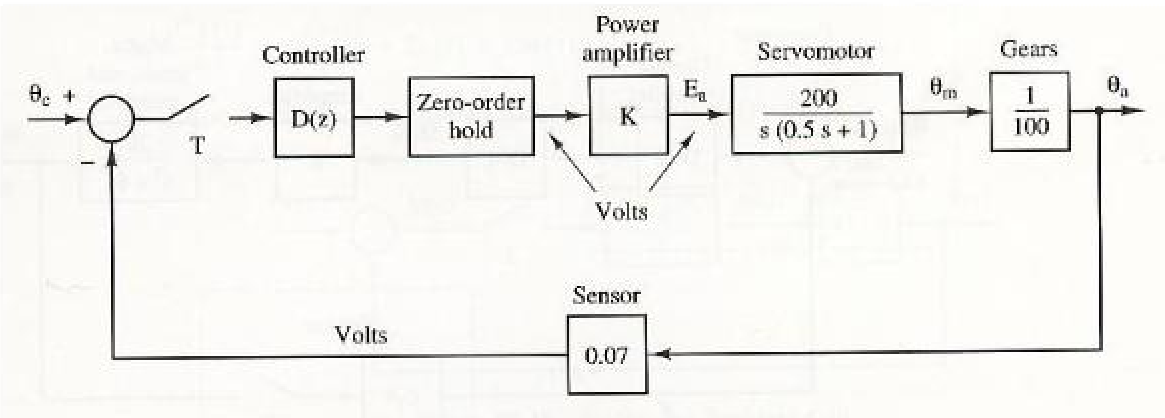


Figure P5-13 Robot arm joint control system.

$K = 2.4$ and $D(z) = 1$ in this block diagram.

The transfer function of this system is,

$$G_c(s) = \frac{9.6}{s(s+2)}$$

Analog state equation is,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 9.6 \end{bmatrix} e$$

For $T = 0.1$ and $a = 2$,

$$\begin{aligned} \mathbf{P} = \mathbf{\Phi}(T) &= \begin{bmatrix} 1 & \frac{1}{a}(1 - e^{-aT}) \\ 0 & e^{-aT} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0.0906 \\ 0 & 0.8187 \end{bmatrix} \end{aligned}$$

$$
\begin{aligned}
\mathbf{q} &= \int_0^T \begin{bmatrix} 1 & \frac{1}{a}(1-e^{-a\tau}) \\ 0 & e^{-at}\tau \end{bmatrix} \begin{bmatrix} 0 \\ K \end{bmatrix} d\tau \\
&= \begin{bmatrix} \frac{K}{a}(T + \frac{1}{a}e^{-aT} - \frac{1}{a}) \\ \frac{K}{a}(1-e^{-aT}) \end{bmatrix} \\
&= \begin{bmatrix} 0.0450 \\ 0.8701 \end{bmatrix}
\end{aligned}
$$

Using the expansion form to calculate $\mathbf{P}$ and $\mathbf{q}$ that includes up to T^2 term.

$$
\begin{aligned}
\mathbf{P} &= \mathbf{\Phi}(T) = e^{\mathbf{A}T} \\
&= \mathbf{I} + \frac{1}{1!}\mathbf{A}T + \frac{1}{2!}\mathbf{A}^2T^2 + \frac{1}{3!}\mathbf{A}^3T^3 + \cdots
\end{aligned}
$$

$$= \begin{bmatrix} 1 & 0.0900 \\ 0 & 0.8200 \end{bmatrix}$$

$$\begin{aligned} \mathbf{q} &= \int_0^T \mathbf{\Phi}(\tau)\mathbf{b}\, d\tau \\ &= (\frac{1}{1!}T + \frac{1}{2!}\mathbf{A}T^2 + \frac{1}{3!}\mathbf{A}^2T^3 + \cdots)\mathbf{b} \\ &= \begin{bmatrix} 0.0480 \\ 0.8640 \end{bmatrix} \end{aligned}$$

where, the output equation is,

$$y = [1, 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{cx}$$

$$e = u - y = u - 0.07\mathbf{cx}$$

Thus, the closed loop state equation is,

$$\mathbf{x}(k+1) = [\mathbf{P} - 0.07\mathbf{qc}]\mathbf{x} + \mathbf{q}u = \begin{bmatrix} 0.9969 & 0.0906 \\ -0.0609 & 0.8187 \end{bmatrix} \mathbf{x} + \mathbf{q}u$$

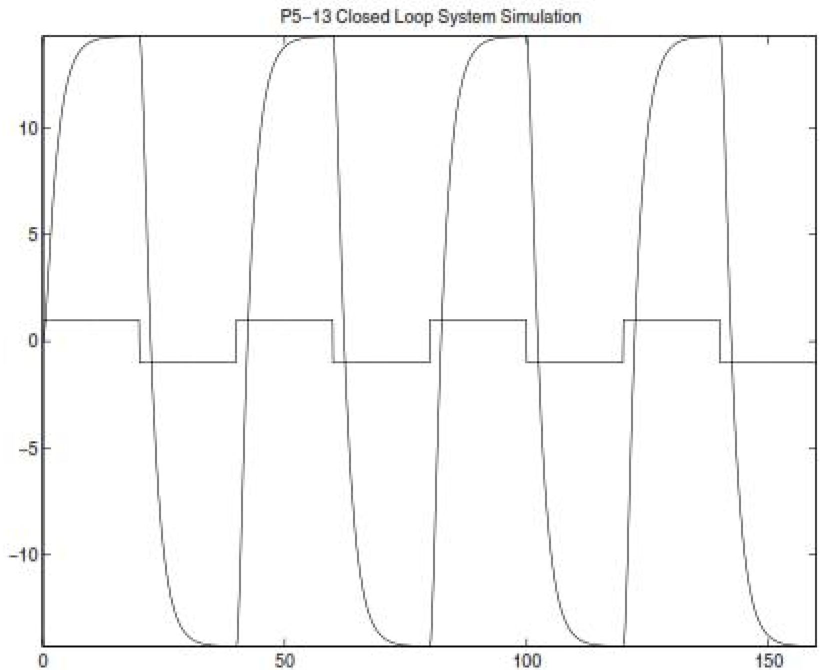
How to program state space model in MATLAB

```
% ------------------------------------------
% Problem 5-13 p. 197 by state space model 2008
% ------------------------------------------
T=0.1
a=2;
Sys=ss([0, 1; 0, -a],[0; 9.6],[1, 0], 0)
dSys=c2d(Sys,T,'zoh')
[A,b,c,d]=ssdata(dSys)
dSys.A
dSys.b
dSys.c
dSys.d

sensor=0.07
P=dSys.A
q=dSys.b
c=Sys.c
Pc=P-q*c*sensor
dSys.A=Pc
```

```
%--
u=ones(1,200); u=[u,-u]; u=[u,u,u,u];
%--
N=size(u,2);
t=[0:T:(N-1)*T];
[angle,x]=dlsim(dSys.a,dSys.b,dSys.c,dSys.d,u);
plot(t,u,'r',t,angle,'b'); axis tight;
title('P5-13 Closed Loop System Simulation')
```

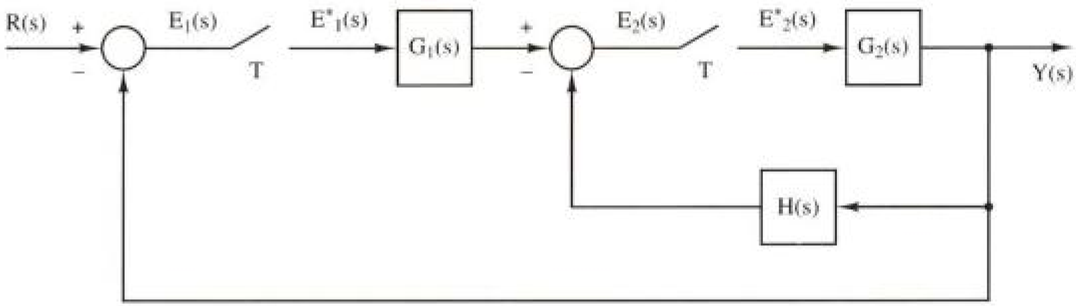
Calculate **P** and **q** from **A**

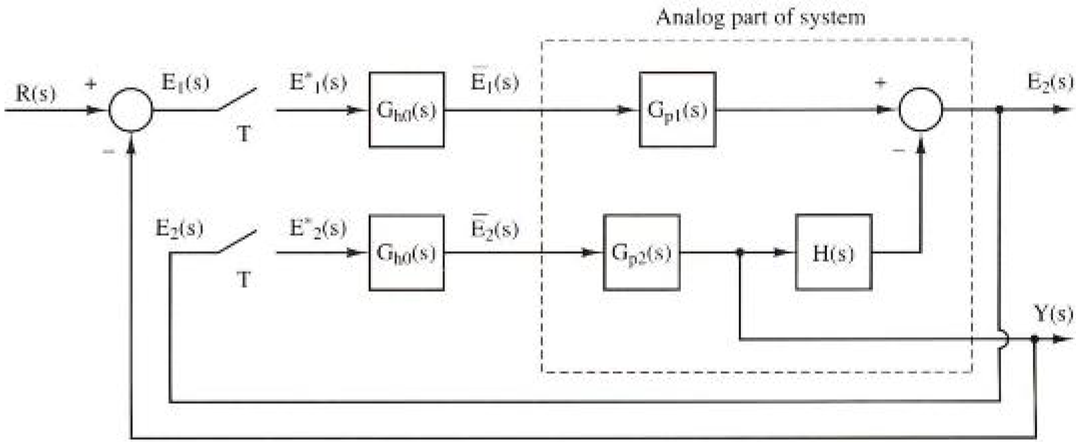
```
% P5_13calc.m
T=0.1
a=2
a12=(1/a)*(1-exp(-a*T))
a22=exp(-a*T)
q1=(1/a)*(T+(1/a)*exp(-a*T)-(1/a))
q2=(1/a)*(1-exp(-a*T))
P=[1, a12; 0, a22]
q=9.6*[q1, q2]'

A=[0, 1; 0, -2]
b=[0; 9.6]
P=[1,0; 0,1]+A*T+(1/2)*A*A*T^2
q=([1,0; 0,1]*T+(1/2)*A*T^2)*b
```


P5−13 Closed Loop System Simulation
10
5
0
−5
−10
0
50
100
150

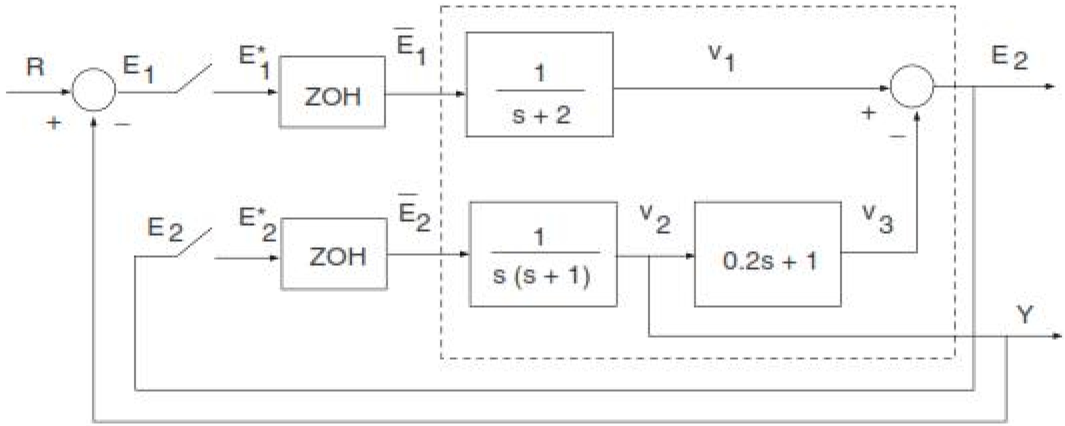
6 Example of more complex state space model

Example 5.5 textbook p. 184





Specifying blocks $G_{1p}(s)$, $G_{2p}(s)$, $H(s)$ as shown in the figure below, derive the state equations in discrete time domain. There is no need to include ZOH as the integrations to find the state equations' matrices take ZOH into account. Also, modeling needs not consider interconnections from outut to input as this can be done by introducing a connection matrix.



For the state variables v_1, v_2 and v_3, we can write

$$
\begin{aligned}
v_1(s) &= \frac{1}{s+2}\,\bar{e}_1(s) \\
v_2(s) &= \frac{1}{s(s+1)}\,\bar{e}_2(s) \\
v_3(s) &= (0.2s+1)\,v_2(s)
\end{aligned}
$$

Augmenting another state variablt v_4 to break v_2 into two equations,

$$
\begin{aligned}
v_1(s) &= \frac{1}{s+2}\,\bar{e}_1(s) \\
v_4(s) &= \frac{1}{s+1}\,\bar{e}_2(s) \\
v_2(s) &= \frac{1}{s}\,v_4(s) \\
v_3(s) &= (0.2s+1)\,v_2(s)
\end{aligned}
$$

$$
\begin{aligned}
v_1(s) &= \frac{1}{s+2}\,\bar{e}_1(s) \\
v_4(s) &= \frac{1}{s+1}\,\bar{e}_2(s) \\
v_2(s) &= \frac{1}{s}\,v_4(s) \\
v_3(s) &= (0.25s+1)\,v_2(s)
\end{aligned}
$$

$$
\begin{aligned}
sv_1(s) = -2v_1(s) + \bar{e}_1(s) \quad &\Rightarrow \quad \dot{v}_1 = -2v_1 + \bar{e}_1 \\
sv_4(s) = -v_4(s) + \bar{e}_2(s) \quad &\Rightarrow \quad \dot{v}_4 = -v_4 + \bar{e}_2 \\
sv_2(s) = v_4(s) \quad &\Rightarrow \quad \dot{v}_2 = v_4 \\
v_3(s) = (0.25s + 1)\, v_2(s) = 0.25v_4(s) + v_2(s) \quad &\Rightarrow \quad v_3 = 0.25v_4 + v_2
\end{aligned}
$$

We can write the state equation,

$$
\begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{v}_4 \end{bmatrix} = \begin{bmatrix} -2 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{e}_1 \\ \bar{e}_2 \end{bmatrix}
$$

A dependent state variable is $v_3 = 0.25v_4 + v_2$ and the output equation is $y = v_2$. Now, consider interconnections between outputs and inputs. From the block diagram,

$$
\begin{aligned}
e_1 &= r - y = r - v_2 \\
e_2 &= v_1 - v_3 = v_1 - v_2 - 0.25v_4
\end{aligned}
$$

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & -1 & -0.25 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_4 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} r$$

Let this equation be represented by a matrix equation,

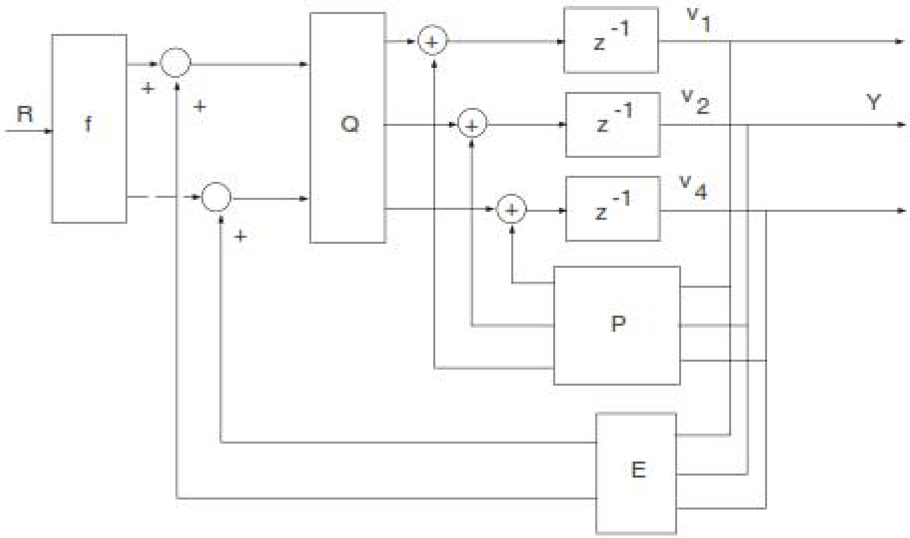
$$\mathbf{e} = \mathbf{E}\mathbf{v} + \mathbf{f}r$$

If we write the discrete state equation derived from the continuous time state equation above as

$$\mathbf{v}(k+1) = \mathbf{P}\mathbf{v}(k) + \mathbf{Q}\mathbf{e}(k),$$

The closed loop state equation is given by

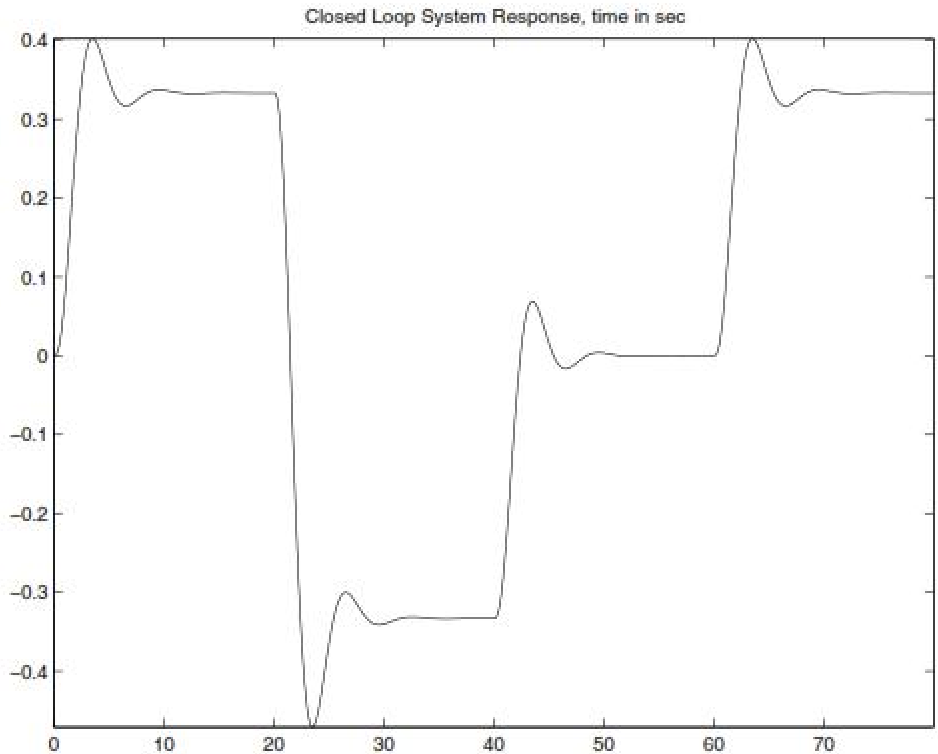
$$\begin{aligned} \mathbf{v}(k+1) &= \mathbf{P}\mathbf{v}(k) + \mathbf{Q}\mathbf{e}(k) \\ &= \mathbf{P}\mathbf{v}(k) + \mathbf{Q}[\mathbf{E}\mathbf{v} + \mathbf{f}r] \\ &= [\mathbf{P} + \mathbf{Q}\mathbf{E}]\mathbf{v} + \mathbf{Q}\mathbf{f}r \end{aligned}$$



The closed system is modelled according to this equation in the following MATLAB porgram.

```
% ---------------------------------------------
% Example 5.5 modified in H(s) p. 185
% ---------------------------------------------
T=0.01
Sys=ss([-2, 0, 0; 0, 0, 1; 0, 0, -1],[1, 0; 0, 0; 0, 1],[0, 1, 0], 0)
```

```
dSys=c2d(Sys,T,'zoh')
[P,Q,C,d]=ssdata(dSys)
E=[0, -1, 0; 1, -1, -0.25]
f=[1; 0]
Pc=P+Q*E
Qc=Q*f

u=ones(1,2000); u=[u,-u,zeros(1,2000),u];
N=size(u,2);
t=[0:T:(N-1)*T];
Cc=C; dc=[0];
sys=ss(Pc,Qc,Cc,dc,T)
[y,x]=dlsim(sys.a,sys.b,sys.c,sys.d,u);
plot(t,y); axis tight;
title('Closed Loop System Response, time in sec');
```


Closed Loop System Response, time in sec
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
0
10
20
30
40
50
60
70