The solution of digital time-varying and nonlinear state-equations also can be obtained using the techniques presented in Chapter 4.

8.10 Design of Multivariable, Digital Control Systems Using Pole-Placement: Regulators, Observers, and Compensators

In the previous section, we noted how a digital state-space representation is equivalent to an analog state-space representation, with the crucial difference that while the latter's characteristics are analyzed in the s-plane, the former is characterized in the z-plane. Therefore, whereas the techniques of Chapter 5 were employed to design an analog control system by placing its poles in the *s-plane*, a digital control system can be similarly designed by placing its poles at desired locations in the *z-plane*. Of course, if a digital system is *controllable*, its poles can be placed anywhere in the *z-plane* using full-state feedback. It can be shown that the conditions and tests for controllability for a linear, time-invariant digital system are the *same* as those presented in Chapter 5 for the analog systems, with the coefficient pair $(\mathbf{A}, \mathbf{B})$ *replaced* by the pair $(\mathbf{A_d}, \mathbf{B_d})$. Hence, one can directly apply the MATLAB (CST) command *ctrb* $(\mathbf{A_d}, \mathbf{B_d})$ to construct the controllability test matrix. Furthermore, using a full-state feedback, $\mathbf{u}(k) = -\mathbf{Kx}(k)$, results in a closed-loop system with poles at the eigenvalues of the closed-loop state-dynamics matrix, $\mathbf{A_{dc}} = (\mathbf{A_d} - \mathbf{B_d K})$. For single-input systems, one can derive the equivalent expression for the full-state feedback gain matrix, $\mathbf{K}$, as the *Ackermann's pole-placement formula* of Eq. (5.52), namely $\mathbf{K} = (\boldsymbol{\alpha} - \mathbf{a})\mathbf{P}'\mathbf{P}^{-1}$, where $\mathbf{P}'$ is the controllability test matrix of the plant in *controller companion form*, α is the vector containing coefficients of the closed-loop characteristic polynomial (except the highest power of z) (i.e. the coefficients of z in $|z\mathbf{I} - \mathbf{A_{dc}}|$), and $\mathbf{a}$ is the vector containing coefficients of the digital plant's characteristic polynomial (except the highest power of z) (i.e. the coefficients of z in $|z\mathbf{I} - \mathbf{A_d}|$). Such a direct equivalence between analog and digital systems allows us to use the MATLAB (CST) commands *place* and *acker* to design full-state feedback regulators for digital systems in the same manner as presented in Chapter 5.

Example 8.18

It is desired to use a digital computer based controller with a sampling interval of 0.2 seconds, to control the inverted pendulum on a moving cart of Example 5.9. We begin by converting the plant to an equivalent digital plant using the command *c2dm* as follows:

```
>>A = [0 0 1 0; 0 0 0 1; 10.78 0 0 0; -0.98 0 0 0];B = [0 0 -1 1]';
  C=[1 0 0 0;0 1 0 0];D=[0; 0]; <enter>

>>[Ad,Bd,Cd,Dd] = c2dm(A,B,C,D,0.2,'zoh') <enter>
```

```
Ad =
        1.2235    0        0.2147    0
       -0.0203    1.0000  -0.0013    0.2000
        2.3143    0        1.2235    0
       -0.2104    0       -0.0203    1.0000


Bd =
       -0.0207
        0.0201
       -0.2147
        0.2013

Cd =
    1   0   0   0
    0   1   0   0

Dd =
   0
   0
```

The plant's pole locations in the z-plane are as follows:

```
>>ddamp(Ad,0.2) <enter>
Eigenvalue   Magnitude   Equiv. Damping   Equiv. Freq. (rad/sec)
  1.9283     1.9283      -1.0000          3.2833
  1.0000     1.0000      -1.0000          0
  1.0000     1.0000      -1.0000          0
  0.5186     0.5186       1.0000          3.2833
```

Note that the plant is *unstable*. Let us choose to place *all* the four closed-loop poles at $z = 0$. Then the full-state feedback gain matrix is obtained by using the command *acker* as follows (we could not have used *place* because the multiplicity of poles to be placed is greater than rank(**B**), i.e. 1):

```
>>P=zeros(1,4); K = acker(Ad,Bd,P) <enter>

K =
-142.7071 -61.5324 -41.5386 -30.7662
```

Let us check the closed-loop pole locations, and calculate the closed-loop natural frequencies and damping ratios as follows:

```
>>ddamp(Ad-Bd*K, 0.2) <enter>

Eigenvalue               Magnitude  Equiv. Damping  Equiv. Freq. (rad/s)
 3.64e-004               3.64e-004   1.00e+000      3.96e+001
 6.03e-008+3.64e-004i    3.64e-004   9.81e-001      4.04e+001
 6.03e-008-3.64e-004i    3.64e-004   9.81e-001      4.04e+001
-3.64e-004               3.64e-004   9.30e-001      4.26e+001
```

Note that the closed-loop poles have not been placed with a great precision, but the error is acceptable for our purposes. The closed-loop digital system is now stable,

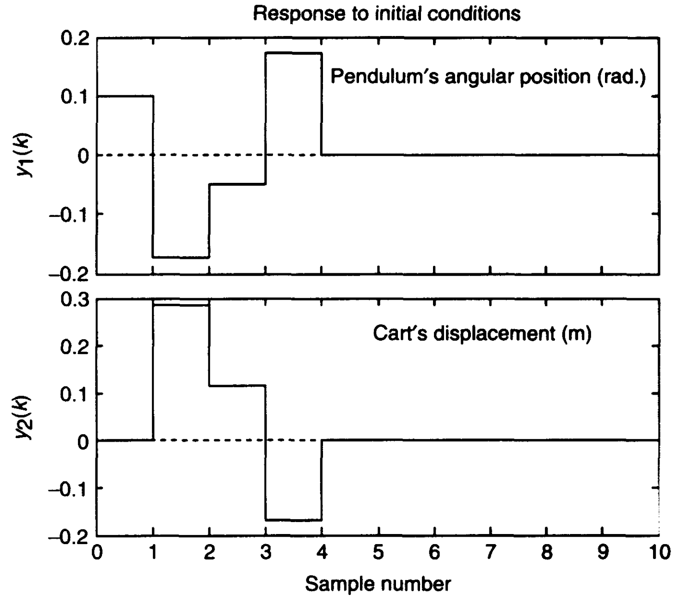

Figure 8.10 Initial response of a control system for inverted pendulum on a moving cart with all closed-loop poles placed at $z = 0$, illustrating the *deadbeat response*

with all damping ratios of $\zeta \approx 1$. Let us see how a closed-loop system with all poles at $z \approx 0$ behaves. Figure 8.10 shows an initial response of the closed-loop system obtained using *dinitial* as follows:

```
>>dinitial(Ad-Bd*K, zeros(4,1),Cd,Dd, [0.1;0;0;0]); <enter>
```

Note the interesting feature of the closed-loop response seen in Figure 8.10: both the outputs settle to a steady-state in *exactly* four time steps. This phenomenon of a sampled data analog system settling to a steady-state in a finite number of sampling intervals is called a *deadbeat response*, and is a property of a stable system with all poles at $z = 0$.

In addition to the full-state feedback regulator problem, all other concepts presented in Chapter 5 for analog control system design – tracking systems, observability, observers, and compensators – can be similarly extended to digital systems with the modification that the time derivative, $dx(t)/dt$, is replaced by the corresponding value of $x(t)$ at the next sampling instant, i.e. $x(k+1)$, and the analog state coefficient matrices, **A**, **B**, **C**, **D**, are replaced by the digital coefficient matrices, **Ad**, **Bd**, **Cd**, **Dd**. For instance, let us consider a full-order observer for a digital system described by Eqs. (8.71) and (8.72). As an extension of the analog observer described by Eq. (5.97), the state-equation for the digital observer can be written as follows:

$$\mathbf{x_o}(k+1) = \mathbf{A_o x_o}(k) + \mathbf{B_o u}(k) + \mathbf{Ly}(k) \tag{8.74}$$

where $\mathbf{x_o}(k)$ is the *estimated state-vector*, $\mathbf{u}(k)$ is the input vector, $\mathbf{y}(k)$ is the output vector (all at the kth sampling instant), $\mathbf{A_o}$, $\mathbf{B_o}$ are the *digital* state-dynamics and control coefficient matrices of the observer, and $\mathbf{L}$ is the *digital observer gain matrix*. The matrices $\mathbf{A_o}$, $\mathbf{B_o}$, and $\mathbf{L}$ must be selected in a design process such that the *estimation error*, $\mathbf{e_o}(k) = \mathbf{x}(k) - \mathbf{x_o}(k)$, is brought to zero in the steady state. On subtracting Eq. (8.74) from Eq. (8.71), we get the following *error dynamics* state-equation:

$$\mathbf{e_o}(k+1) = \mathbf{A_o}\mathbf{e_o}(k) + (\mathbf{A_d} - \mathbf{A_o})\mathbf{x}(k) + (\mathbf{B_d} - \mathbf{B_o})\mathbf{u}(k) - \mathbf{L}\mathbf{y}(k) \tag{8.75}$$

Substitution of Eq. (8.72) into Eq. (8.75) yields

$$\mathbf{e_o}(k+1) = \mathbf{A_o}\mathbf{e_o}(k) + (\mathbf{A_d} - \mathbf{A_o})\mathbf{x}(k) + (\mathbf{B_d} - \mathbf{B_o})\mathbf{u}(k) - \mathbf{L}[\mathbf{C_d}\mathbf{x}(k) + \mathbf{D_d}\mathbf{u}(k)] \tag{8.76}$$

or

$$\mathbf{e_o}(k+1) = \mathbf{A_o}\mathbf{e_o}(k) + (\mathbf{A_d} - \mathbf{A_o} - \mathbf{L}\mathbf{C_d})\mathbf{x}(k) + (\mathbf{B_d} - \mathbf{B_o} - \mathbf{L}\mathbf{D_d})\mathbf{u}(k) \tag{8.77}$$

From Eq. (8.77), it is clear that estimation error, $\mathbf{e_o}(k)$, will go to zero in the steady state irrespective of $\mathbf{x}(k)$ and $\mathbf{u}(k)$, if all the *eigenvalues* of $\mathbf{A_o}$ are inside the *unit circle*, and the coefficient matrices of $\mathbf{x}(k)$ and $\mathbf{u}(k)$ are *zeros*, i.e. $(\mathbf{A_d} - \mathbf{A_o} - \mathbf{L}\mathbf{C_d}) = \mathbf{0}$, $(\mathbf{B_d} - \mathbf{B_o} - \mathbf{L}\mathbf{D_d}) = \mathbf{0}$. The latter requirement leads to the following expressions for $\mathbf{A_o}$ and $\mathbf{B_o}$:

$$\mathbf{A_o} = \mathbf{A_d} - \mathbf{L}\mathbf{C_d}; \quad \mathbf{B_o} = \mathbf{B_d} - \mathbf{L}\mathbf{d_d} \tag{8.78}$$

The error dynamics state-equation is thus the following:

$$\mathbf{e_o}(k+1) = (\mathbf{A_d} - \mathbf{L}\mathbf{C_d})\mathbf{e_o}(k) \tag{8.79}$$

The observer gain matrix, $\mathbf{L}$, must be selected to place all the eigenvalues of $\mathbf{A_o}$ (which are also the poles of the observer) at desired locations inside the unit circle in the z-plane, which implies that the estimation error dynamics given by Eq. (8.79) is *asymptotically stable* (i.e. $\mathbf{e_o}(k) \to \mathbf{0}$ as $k \to \infty$).

The digital observer described by Eq. (8.74) estimates the state-vector at a given sampling instant, $\mathbf{x_o}(k+1)$, based upon the measurement of the output, $\mathbf{y}(k)$, which is *one sampling instant old*. Such an estimate is likely to be *less accurate* than that based on the current value of the measured output, $\mathbf{y}(k+1)$. Thus, a *more accurate* linear, digital observer is described by the following state-equation:

$$\mathbf{x_o}(k+1) = \mathbf{A_o}\mathbf{x_o}(k) + \mathbf{B_o}\mathbf{u}(k) + \mathbf{L}^*\mathbf{y}(k+1) \tag{8.80}$$

where $\mathbf{L}^*$ is the new observer gain matrix. The observer described by Eq. (8.80) is referred to as the *current observer*, because it employs the *current* value of the output, $\mathbf{y}(k+1)$. Equation (8.80) yields the following state-equation for the estimation error:

$$\begin{aligned}\mathbf{e_o}(k+1) = {} & \mathbf{A_o}\mathbf{e_o}(k) + (\mathbf{A_d} - \mathbf{A_o})\mathbf{x}(k) + (\mathbf{B_d} - \mathbf{B_o})\mathbf{u}(k) \\ & - \mathbf{L}^*[\mathbf{C_d}\mathbf{x}(k+1) + \mathbf{D_d}\mathbf{u}(k+1)]\end{aligned} \tag{8.81}$$

Substituting for $\mathbf{x}(k+1)$ from Eq. (8.71), we get

$$\mathbf{e_o}(k+1) = \mathbf{A_o e_o}(k) + (\mathbf{A_d} - \mathbf{A_o} - \mathbf{L^* C_d A_d})\mathbf{x}(k) + (\mathbf{B_d} - \mathbf{B_o} \\ -\mathbf{L^* C_d B_d})\mathbf{u}(k) - \mathbf{L^* D_d u}(k+1) \tag{8.82}$$

For $\mathbf{e_o}(k)$ to go to zero in the steady state, the following conditions must be satisfied in addition to $\mathbf{A_o}$ having all eigenvalues inside the unit circle:

$$\mathbf{A_o} = \mathbf{A_d} - \mathbf{L^* C_d A_d} \tag{8.83}$$

$$(\mathbf{B_d} - \mathbf{B_o} - \mathbf{L^* C_d Bd})\mathbf{u}(k) = \mathbf{L^* D_d u}(k+1) \tag{8.84}$$

which result in the following state equations for the estimation error and the observer:

$$\mathbf{e_o}(k+1) = (\mathbf{A_d} - \mathbf{L^* C_d A_d})\mathbf{e_o}(k) \tag{8.85}$$

$$\mathbf{x_o}(k+1) = (\mathbf{A_d} - \mathbf{L^* C_d A_d})\mathbf{x_o}(k) + (\mathbf{B_d} - \mathbf{L^* C_d B_d})\mathbf{u}(k) \\ - \mathbf{L^* D_d u}(k+1) + \mathbf{L^* y}(k+1) \tag{8.86}$$

Equation (8.86) can be alternatively expressed as the following *two* difference equations:

$$\mathbf{x_1}(k+1) = \mathbf{A_d x_o}(k) + \mathbf{B_d u}(k) \tag{8.87}$$

$$\mathbf{x_o}(k+1) = \mathbf{x_1}(k+1) + \mathbf{L^*}[\mathbf{y}(k+1) - \mathbf{D_d u}(k+1) - \mathbf{C_d x_1}(k+1)] \tag{8.88}$$

where $\mathbf{x_1}(k+1)$ is the *first estimate* of the state (or *predicted state*) at the $(k+1)$th sampling instant based upon the quantities at the *previous* sampling instant, and $\mathbf{x_o}(k+1)$ is the *corrected* – or *final* – state estimate due to the measurement of the output at the current sampling instant, $\mathbf{y}(k+1)$. The *predictor-corrector* observer formulation given by Eqs. (8.87) and (8.88) is sometimes more useful when implemented in a computer program.

For practical purposes, it is not the observer alone that we are interested in, but the implementation of the digital observer in a feedback *digital compensator*. Using a full-order current digital observer in a full-state feedback compensator results in a feedback control law of the form

$$\mathbf{u}(k) = -\mathbf{K x_o}(k) \tag{8.89}$$

Substituting Eq. (8.89) into Eq. (8.86), we can write the state-equation for the estimated state as follows:

$$\mathbf{x_o}(k+1) = (\mathbf{I} - \mathbf{L^* D_d K})^{-1}(\mathbf{A_d} - \mathbf{L^* C_d A_d} - \mathbf{B_d K} + \mathbf{L^* C_d B_d K})\mathbf{x_o}(k) \\ + (\mathbf{I} - \mathbf{L^* D_d K})^{-1}\mathbf{L^* y}(k+1) \tag{8.90}$$

Note that Eq. (8.90) requires that the matrix $(\mathbf{I} - \mathbf{L^* D_d K})$ must be *non-singular*. Substituting Eq. (8.89) into Eq. (8.71), we get the other state-equation for the closed-loop system as follows:

$$\mathbf{x}(k+1) = \mathbf{A_d x}(k) - \mathbf{B_d K x_o}(k) \tag{8.91}$$

Equations (8.90) and (8.91) must be solved simultaneously to get the composite state-vector for the closed-loop system, $\mathbf{x_c}(k) = [\mathbf{x^T}(k); \mathbf{x_o}^\mathbf{T}(k)]^T$. We can combine Eqs. (8.90) and (8.91) into the following state-equation:

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x_o}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A_d} & -\mathbf{B_d K} \\ \mathbf{0} & (\mathbf{I} - \mathbf{L^* D_d K})^{-1}(\mathbf{A_d} - \mathbf{L^* C_d A_d} - \mathbf{B_d K} + \mathbf{L^* C_d B_d K}) \end{bmatrix} \times \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{x_o}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ (\mathbf{I} - \mathbf{L^* D_d K})^{-1}\mathbf{L^*} \end{bmatrix} \mathbf{y}(k+1) \quad (8.92)$$

Substituting $\mathbf{y}(k+1) = \mathbf{C_d x}(k+1) + \mathbf{D_d u}(k+1) = \mathbf{C_d x}(k+1) - \mathbf{D_d K x_o}(k+1)$ into Eq. (8.92), and re-arranging, we can write

$$\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x_o}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A_d} & -\mathbf{B_d K} \\ (\mathbf{I} + \mathbf{F^* D_d K})^{-1}\mathbf{F^* C_d A_d} & (\mathbf{I} + \mathbf{F^* D_d K})^{-1}(\mathbf{I} - \mathbf{L^* D_d K})^{-1}(\mathbf{A_d} - \mathbf{L^* C_d A_d} - \mathbf{B_d K}) \end{bmatrix} \times \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{x_o}(k) \end{bmatrix} \quad (8.93)$$

where $\mathbf{F}^* = (\mathbf{I} - \mathbf{L^* D_d K})^{-1}\mathbf{L}^*$. Note that Eq. (8.93) is a homogeneous digital state-equation representing closed-loop dynamics of the compensated system.

Example 8.19

Let us design a full-order observer based digital compensator with a sampling interval of 0.2 seconds, to control the inverted pendulum on a moving cart of Example 8.18. The cart's displacement, $y_2(t)$, is the only measured output, which results in $\mathbf{C} = [0; 1; 0; 0]$, and $\mathbf{D} = 0$. We begin by converting the plant to an equivalent digital plant with a z.o.h using the command *c2dm* as follows:

```
>>A = [0 0 1 0; 0 0 0 1; 10.78 0 0 0; -0.98 0 0 0]; B = [0 0 -1 1]';
   C = [0 1 0 0]; D = 0; <enter>

>>[Ad,Bd,Cd,Dd] = c2dm(A,B,C,D,0.2,'zoh'); <enter>

Ad =
  1.2235      0  0.2147       0
 -0.0203  1.0000 -0.0013      0.2000
  2.3143      0  1.2235       0
 -0.2104      0 -0.0203    1.0000

Bd =
 -0.0207
  0.0201
 -0.2147
  0.2013

Cd =
  0  1  0  0
```

```
Dd =
  0
```

Note that since $\mathbf{A_o} = \mathbf{A_d} - \mathbf{L^* C_d A_d}$, the observability test matrix is now computed by *ctrb* $(\mathbf{A_d}^T, \mathbf{A_d}^T \mathbf{C_d}^T)$. Let us check whether the digital plant is observable as follows:

```
>>rank(ctrb(Ad',Ad'*Cd')) <enter>

ans =
4
```

Since the rank of the observability test matrix is equal to the order of the plant, the plant is observable. The observer gain matrix is calculated by placing the observer poles well inside the unit circle, such as at $z = \pm 0.7$, and $z = 0.5 \pm 0.5i$. Then the observer gain matrix, $\mathbf{L^*}$, is computed using *place* as follows:

```
>>P=[-0.7 0.7 0.5-0.5i 0.5+0.5i]; Lstar = place(Ad', Ad'*Cd', P)'
  <enter>

place: ndigits= 15
Lstar =
 -38.3080
   1.2450
 -102.1822
   6.4364
```

Let us check whether observer poles are placed as desired:

```
>>eig(Ad - Lstar*Cd*Ad) <enter>

ans =
        0.5000+0.5000i
        0.5000-0.5000i
        0.7000
        -0.7000
```

Note that the plant is strictly proper. Hence, the observer's state coefficient matrices are computed as follows:

```
>>Ao = Ad - Lstar*Cd*Ad, Bo = Bd - Lstar*Cd*Bd <enter>

Ao =
   0.4453   38.3080   0.1635    7.6616
   0.0050   -0.2450   0.0003   -0.0490
   0.2385  102.1822   1.0870   20.4364
  -0.0796   -6.4364  -0.0117   -0.2873

Bo =
   0.7480
  -0.0049
   1.8357
   0.0722
```

A full-state feedback regulator was designed for this plant in Example 8.18, to place all the regulated system's poles at $z = 0$. Using this regulator gain matrix, **K**, the digital closed-loop state-dynamics matrix (Eq. (8.93)) is calculated as follows:

```
>>Ac=[Ad -Bd*K; Lstar*Cd*Ad Ao-Bd*K]; <enter>
```

Finally, let us calculate the compensated digital system's initial response to $\mathbf{x_c}(0) = [\mathbf{x}^\mathbf{T}(0); \mathbf{x_o}^\mathbf{T}(0)]^T = [0.1; 0; 0; 0; 0; 0; 0; 0]^T$ (implying an initial pendulum angle of 0.1 rad.) using the command *dinitial* as follows:

```
>>dinitial(Ac,zeros(8,1),[Cd zeros(1,4)],0,[0.1 zeros(1,7)]') <enter>
```

The resulting digital initial response is plotted in Figure 8.11. The initial response settles to zero in about 25 sampling instants, with a maximum overshoot of 0.6 m. Note that the closed-loop system's performance has deteriorated when compared to the deadbeat response of Figure 8.10, due to the presence of dominant observer poles. Let us see what happens if we place all the observer poles also at $z = 0$, as follows:

```
>>P=zeros(1,4);Lstar = acker(Ad',Ad'*Cd',P)'; <enter>
```

and re-calculate the closed-loop initial response, which is plotted in Figure 8.12. Figure 8.12 shows that the deadbeat response of Figure 8.10 has been largely

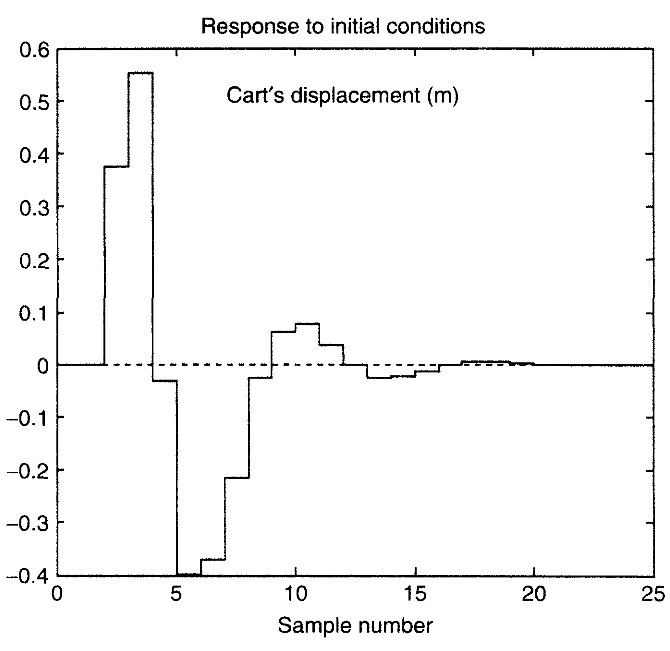

Figure 8.11 Closed-loop initial response (cart's displacement, $y_2(k)$, in m) of inverted pendulum on a moving cart with full-order compensator designed using observer poles at $z = \pm 0.7$, and $z = 0.5 \pm 0.5i$

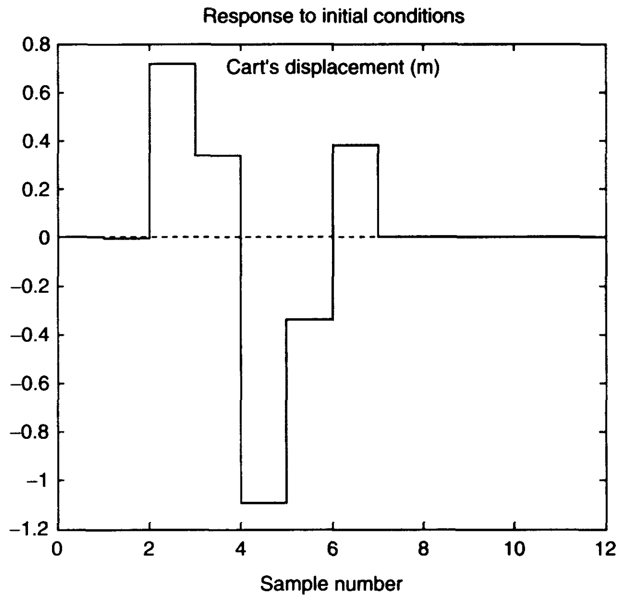

Figure 8.12 Closed-loop initial response (pendulum angle in radian) of inverted pendulum on a moving cart with full-order compensator designed using all observer poles at $z = 0$, and all regulator poles also at $z = 0$. Note the restoration of the *deadbeat response* of Figure 8.10, but with larger overshoots

restored, but with twice the overshoots of Figure 8.10, and with a larger settling time (seven steps) when compared to that of Figure 8.10 (four steps).

As an alternative to the calculations given above, MATLAB (CST) provides the command *dreg* for constructing the state-space representation of the full-order digital compensator of Eq. (8.93), and can be used to derive the full-order, full-state feedback closed-loop digital system as follows:

```
>>[ac,bc,cc,dc] = dreg(Ad,Bd,Cd,Dd,K,Lstar) <enter>
```

where (**ac**, **bc**, **cc**, **dc**) is the digital state-space representation of the digital compensator whose input is the plant output, $\mathbf{y}(k+1)$, and whose output is the plant input, $\mathbf{u}(k)$. Then the digital compensator is put in a closed-loop with the plant, to obtain the closed-loop state-space representation (**Ac**, **Bc**, **Cc**, **Dc**) as follows:

```
>>sysc=ss(ac,bc,cc,dc); sysp=ss(Ad,Bd,Cd,Dd); <enter>

>>syss=series(sysc,sysp);sysCL=feedback(syss,eye(size(syss))); <enter>

>>[Ac,Bc,Cc,Dc] = ssdata(sysCL); <enter>
```

The command *dreg* allows constructing a compensator in which only selected outputs are measured that are specified using an additional input argument, *SENSORS*. Also,