Chapter 9
State Feedback Control

Consider first continuous systems. The state space representation of a continuous, linear, time invariant single input–single output system can be given by parameter matrices A, b, c, d in the following form:

$$\dot{x} = Ax + bu$$
$$y = c^T x + du$$

(The upper index T indicates transpose, i.e. c^T is a row vector.) The equations above (the state equation and the output equation) determine the transfer function between the u input signal and the y output signal, which is calculated by

$$P(s) = \frac{Y(s)}{U(s)} = c^T(sI - A)^{-1}b + d$$

The system model characterized by the four parameters $\{A, b, c, d\}$ is called the state model.

The poles of the model are the roots of the characteristic equation

$$\det(sI - A) = 0.$$

In most practical cases, $d = 0$.

By state feedback, the control signal is obtained from the state variables feeding them back to the input through the constant elements of the vector k^T:

$$u = k_r r - k^T x.$$

The state feedback control shown in Fig. 9.1 modifies both the static and the dynamic response of the system between the reference signal r and the output signal y.

Fig. 9.1 State feedback control
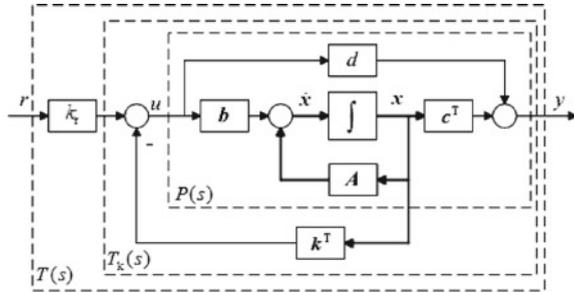
In the feedback, let us consider the feedback (row) vector k^{T}, and in the forward path suppose a compensation factor k_r. The control signal is obtained as $u = k_r r - k^{\mathrm{T}} x$. The equations of the closed loop control system are as follows (here d is also considered, its value is generally zero):

$$\dot{x} = \left(A - bk^{\mathrm{T}}\right)x + k_r b r$$
$$y = \left(c^{\mathrm{T}} - dk^{\mathrm{T}}\right)x + dk_r r$$

By introducing the notation $A_k = A - bk^{\mathrm{T}}$, $b_k = k_r b$, $c_k = c - dk$, $d_k = dk_r$, we have

$$\dot{x} = A_k x + b_k r$$
$$y = c_k^{\mathrm{T}} x + d_k r$$

and the characteristic equation is

$$\det(sI - A_k) = \det\left(sI - A + bk^{\mathrm{T}}\right) = 0.$$

Comparing the characteristic equations of the open and of the closed loops, it can be seen that the poles of the open loop depend only on A, while the poles of the closed loop depend on three parameters $\{A,\ b,\ k\}$. The performance of the closed loop is prescribed by the required location of its poles in the complex plane. What has to be found is a state feedback vector k that ensures that the roots of the characteristic equation are in the required locations.

9.1 State Feedback with Pole Placement

The design of state feedback is executed in three steps:

- choose the desired location of the poles of the closed loop system;
- for *SISO* systems the state feedback vector k can be determined by the Ackermann formula (textbook [1], Sect. 9.1), in MATLAB™ by using the command acker.
- determine the compensation factor k_r to fulfill the static requirements.

Example 9.1 Consider the system given by the following transfer function:

$$P(s) = \frac{6}{(s+1)(s+2)(s+3)}$$

The static gain of the system is 1, its poles are −1, −2, −3. Give the system in MATLAB™ with its poles, then transform it to state space form.

```
num=6;
den=poly([-1,-2,-3])
P=tf(num,den)
[A,b,c,d]=tf2ss(num,den)
```

The command tf2ss gives the controllability canonical form of the state equation.

$$A = \begin{bmatrix} -6 & -11 & -6 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \ b = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \ c^{\mathrm{T}} = [0 \ \ 0 \ \ 6], \ d = 0$$

Choose the p_k poles of the closed loop by

```
pk=[-6;-3+i*4;-3-i*4]
```

(The conjugate complex poles can be considered as the poles of a second order oscillating system. The damping factor ξ is calculated from the angle φ of the vector of the poles, $\xi = \cos \varphi = 3/\sqrt{9 + 16} = 0.6$).

Let us remark that the system can be accelerated by shifting its poles to the left in the complex plane. Analyse the required behaviour of the step response with these prescribed poles.

First let the numerator be the constant 1, and let the denominator be the characteristic polynomial.

```
numk=1
denk=poly(pk)
H=tf(numk,denk)
H=zpk(H)
g0=dcgain(H)
```

To get a system with unit gain, normalize the system by its static gain. Compare the step responses of the original and the prescribed system.

```
Hn=H/g0
step(P,'b',Hn,'r'); grid
```

In Fig. 9.2 it can be seen that with this pole prescription the system can be accelerated significantly.

Then using the ACKERMANN formula determine the state feedback vector that shifts the poles $p_o^{\mathrm{T}} = [-1 \quad -2 \quad -3]$ of the open loop to the required locations

Fig. 9.2 With pole
prescription the system can be
accelerated
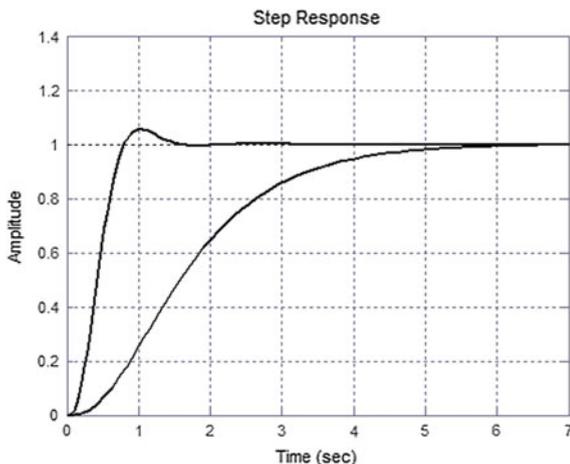
$p_k^T = [-6 \quad -3+4j \quad -3-4j]$ of the closed loop. The analytical form of the
ACKERMANN formula is

$$k^T = [1, 0, \ldots, 0] M_c^{-1} \mathcal{R}(A),$$

where M_c is the controllability matrix of the open loop, $\mathcal{R}(s)$ is the characteristic
equation of the closed loop (which is determined by its prescribed poles), and $\mathcal{R}(A)$
is the value of this polynomial at A. In MATLAB™ all this is executed by one
command:
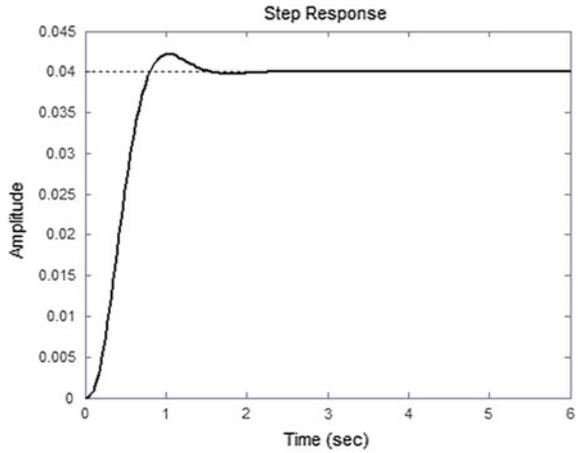
```
k=acker(A,b,pk)
     k = 6    50   144

Tk=ss(A-b*k,b,c,d)
Tk=zpk(Tk)
step(Tk,6)
```

In Fig. 9.3 it can be seen that by shifting the poles to the left, the transients of the
step response decay faster, but the static value is not satisfactory. To ensure a static
gain of value 1, a compensation factor k_r is calculated.

```
kr=1/dcgain(Tk)
     kr =
         25.0000
Tk1=kr*Tk
```

Fig. 9.3 Step response of
state feedback

or

```
Tk1=ss(A-b*k,kr*b,c,d),Tk=zpk(Tk)
               150
         ---------------------
         (s+6) (s^2  + 6s + 25)
step(Tk1,'b')
```
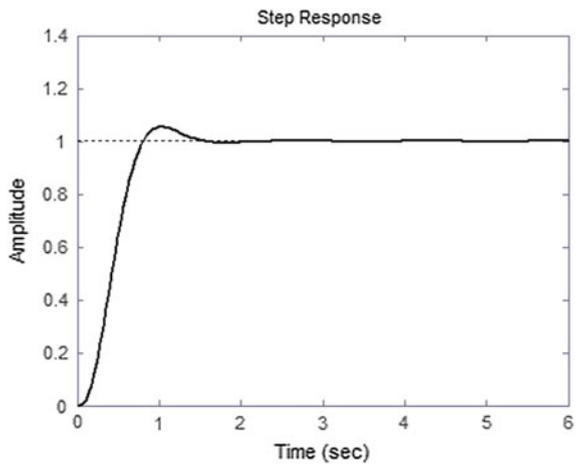
In Fig. 9.4 it can be seen that setting the state feedback vector k^T and the compensation factor k_r, the settling process is fast and there is no static error. The dynamic properties have also been improved by this pole placement.

Fig. 9.4 Static error can be
compensated

It should be mentioned that the choice of the state feedback vector k^T is not unique, it depends on the form of the state space representation. Let us check the value of the state feedback vector when the state space representation of the process is given in a different form.

```
s=zpk('s')
P=6/((s+1)*(s+2)*(s+3))
[A1,b1,c1,d1]=ssdata(P)
k1=acker(A1,b1,pk)
        k1 =  40.8248  13.0639  2.4495
```

A different state representation yields a different state feedback vector. But the transfer functions of the two different representations are the same, yielding the same step responses.

```
Tk1=ss(A1-b1*k1,b1,c1,d1)
kr1= 1/dcgain(Tk1)
T1=zpk(T1)*kr1
              150
        ------------------
        (s+6) (s^2 + 6s + 25)
```

Example 9.2 With state feedback, unstable processes can be stabilized easily. The state feedback constants are calculated by prescribing stable closed loop poles.

Consider the transfer function of an unstable process containing one pole in the right half-plane:
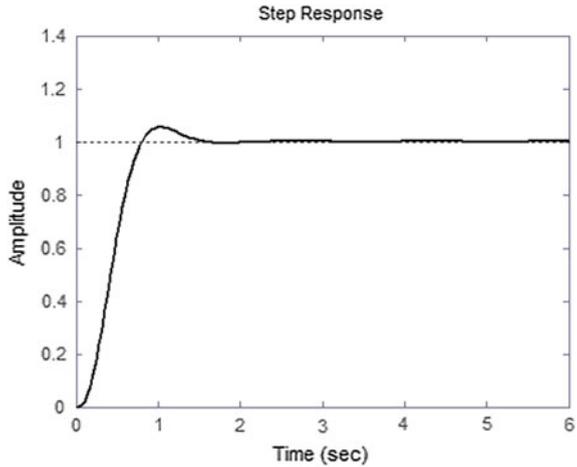
$$P(s) = \frac{-6}{(s-1)(s+2)(s+3)}$$

Suppose that the prescribed poles of the closed loop are

```
pk=[-6;-3+i*4;-3-i*4]
```

Determine the state feedback vector and plot the step response of the closed loop. The MATLAB™ commands to do this are

```
num=-6;
den=poly([1,-2,-3])
P=tf(num,den)
[A,b,c,d]=tf2ss(num,den)
pk=[-6;-3+i*4;-3-i*4]
k=acker(A,b,pk)
```

Fig. 9.5 Step response of a state feedback control system with an unstable process

```
Tk=ss(A-b*k,b,c,d)
kr=1/dcgain(Tk)
Tk1=ss(A-b*k,kr*b,c,d)
step(Tk1,6)
```

and the state feedback vector is then

```
k = 8   60   156
```

Figure 9.5 shows the step response which ensures a performance corresponding to the prescribed poles.

9.2 Introducing an Integrator into the Feedback Loop

The properties of state feedback control are analogous to the effect of serial *PD* compensation, resulting in acceleration of the control circuit. The static accuracy is ensured by a gain factor acting outside of the feedback circuit. This gain factor is determined by the knowledge of the system parameters. This means that this gain is sensitive to the accuracy of the knowledge of the parameters. Furthermore, the effect of the disturbances can not be compensated with elements outside of the feedback circuit. Therefore to ensure the static accuracy—similarly to design considerations in the frequency domain—it is expedient to introduce an integrator into the control circuit.

The state equation of the process is extended by the state variable x_i, which is the integral of the output signal *y* (Fig. 9.6).
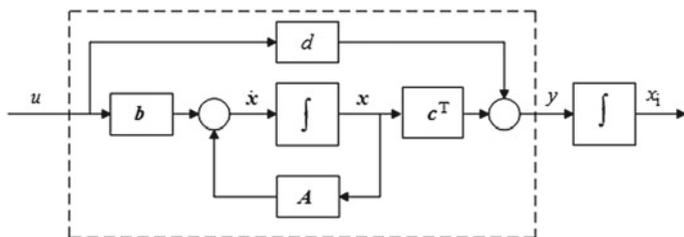
Fig. 9.6 An additional state variable is introduced as integral of the output variable

The state equation of the extended system is

$$\begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A & 0 \\ c^T & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} u = A_b x_b + b_b u$$

$$y = \begin{bmatrix} c^T & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + du = c_b^T x_b + du$$

So the number of the state variables is increased by 1. For state feedback design, the number of the prescribed poles should also increase by 1. The state feedback vector k_b^T is calculated now for the extended state equation with state matrices A_b and b_b, for the prescribed poles p_b, using the ACKERMANN formula. These poles will be the prescribed poles of the characteristic equation $\det(sI - A_b + b_b k_b^T) = 0$.

Figure 9.7 shows the extended state feedback system. The integrator is located after the error signal.

Supposing a single input–single output *SISO* system and $d = 0$ the state equation of the closed loop system is written as

$$\dot{x}_z = \begin{bmatrix} \dot{x} \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A - bk^T & bk_i \\ -c^T & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r = A_z x_z + b_z r$$

$$y = \begin{bmatrix} c^T & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + 0 \cdot r = c_z^T x_z + 0 \cdot r$$
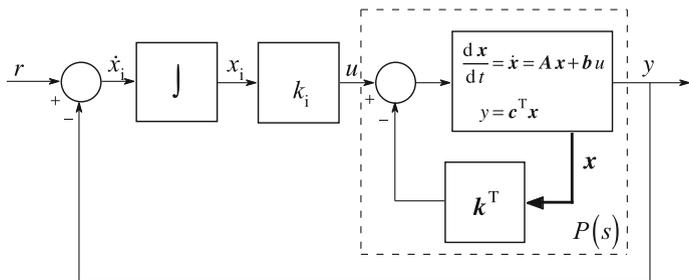

Fig. 9.7 Block diagram of the extended state feedback system

Example 9.3 Extend the process given in Example 9.1 with an integrating state variable.

```
num =6;
den =poly([-1,-2,-3])
P=tf(num,den)
[A,b,c,d]=tf2ss(num,den)
```

The parameter matrices of the extended system are

```
nulvec=[0;0;0];
Ab=[A nulvec;c 0]
bb=[b;0]
        Ab =
             -6      -11      -6       0
              1        0       0       0
              0        1       0       0
              0        0       6       0
        bb =
              1
              0
              0
              0
```

Let the poles of the closed loop system be

```
pb=[-9 -6 -3+i*4 -3-i*4];
```

Determine the state feedback vector:

```
kb=acker(Ab,bb,pb)
        kb =
            15    158    693    225
```

The first three elements of the extended state feedback vector realize the state feedback from the original state variables, while the fourth element, k_i, belongs to the artificially introduced integrator:

```
    k=kb(1:3)
    ki=kb(4)
        k =
```

```
      15     158      693
```

The state matrices of the closed loop system are

```
Az=[A-b*k b*ki;-c 0]
bz=[nulvec;1]
cz=[c 0]
dz=0;
      Az =
            -21 -169     -699      225
              1    0        0        0
              0    1        0        0
              0    0       -6        0
      bz =
              0
              0
              0
              1
      cz =
              0    0        6        0
      dz=0
```

The step response of the closed loop (Fig. 9.8) is found by

```
t=0:0.1:6;
step(Az,bz,cz,dz,1,t),grid
```


Fig. 9.8 Step response of the closed loop
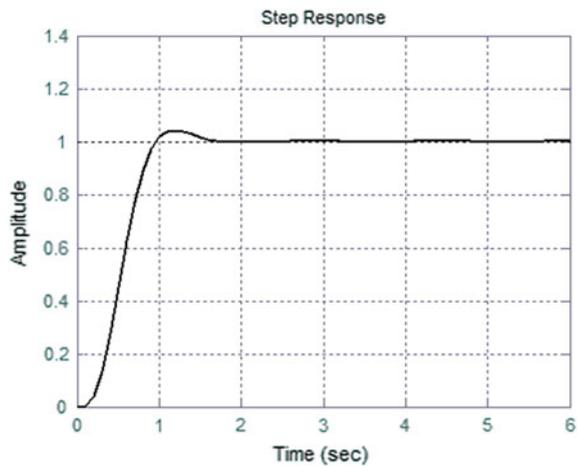
It can be seen that the dynamic and static behaviour of the closed loop system is appropriate.

9.3 State Estimation

In practical applications, the instrumentation of the processes includes possibilities for measurement of several variables. Sensors measure the output signal, but generally not all the state variables are available for measurement. In this case the control with state feedback has to be supplemented with the estimation of the non-measurable state variables. The block scheme of a *state estimator* is shown in Fig. 9.9. The estimator contains the model of the system. It is assumed that $d = 0$. If the system is known, the parameter matrices of the model are the same as the parameter matrices of the system. The difference between the output of the system and the model constitutes an error signal. This error signal is fed back to the summing point at the derivatives of the estimated variables to modify their values. The aim is to ensure that the estimated state variables move quickly to follow the movement of the real state variables. The state estimation circuit forms a closed loop whose input signal is y, the output signal of the process. The poles of the estimation circuit can be prescribed. An important requirement is that the dynamics of the estimation circuit should be much faster than the dynamics of the process. The gain l of the estimation circuit can be calculated by the ACKERMANN formula. It can be seen in the figure that the behaviour of the estimation circuit is influenced by
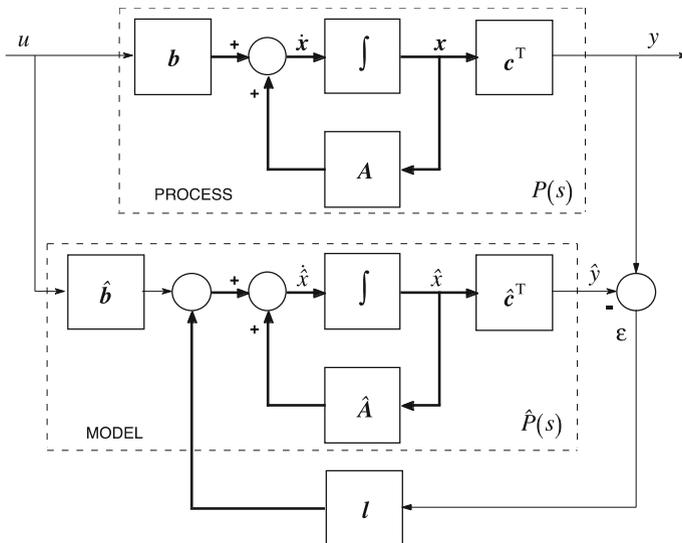

Fig. 9.9 Block diagram of state estimation

the parameter matrices $\hat{A}$ and $\hat{c}^T$. (For simplicity, A, b and c^T are used in the formulas.)

Let us suppose that the parameter matrices of the process and of the model are the same ($\varepsilon = 0$). The free motion of the system states is to be estimated, i.e. the motion of the state variables starting from their initial values supposing a zero input signal. The output disturbance is zero. Based on Fig. 9.9, the estimated state variables can be calculated according to the following relation:

$$\dot{\hat{x}} = A\,\hat{x} + b\,u + l\,c^T(x - \hat{x}) = \left(A - l\,c^T\right)\hat{x} + l\,y.$$

Let us introduce the error signal $\varepsilon = x - \hat{x}$. The derivative of the error signal is obtained if the equation given for the estimated state variables is subtracted from the equation of the original state variables.

$$\dot{x} - \dot{\hat{x}} = \dot{\varepsilon} = A\,\varepsilon - l\,c^T\varepsilon = \left(A - l\,c^T\right)\varepsilon = A_\varepsilon\varepsilon = A\,\varepsilon - l(y - \hat{y})$$

The estimation circuit can be redrawn as Fig. 9.10.

The parameters of the estimation circuit (the elements of the vector l) can be calculated by the ACKERMANN formula prescribing the roots of the characteristic equation of the closed estimation circuit.

```
L=acker(A',c',Pe)'
```

Here `Pe` is the vector of the prescribed poles of the estimation circuit. The estimation circuit has to be faster than the process, and faster than the control system with state feedback. (Transposition is required to reconcile the dimensions of the matrices and the vectors.)

Example 9.4 The process is the third order proportional system investigated also in Example 9.1 (without the extension by the integrating state variable). Let the initial conditions of all the three state variables have the value 1. The reference signal and the disturbance signal are zero. Give the poles of the estimation circuit as
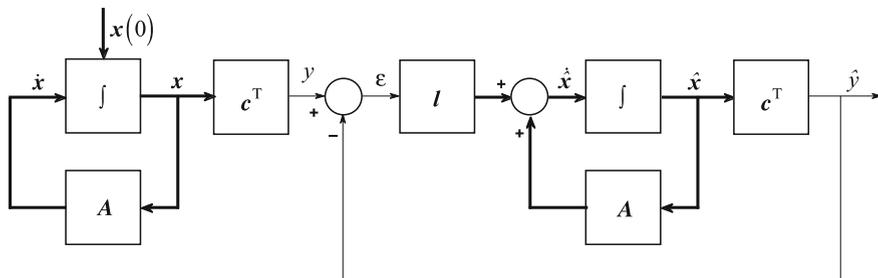

Fig. 9.10 The redrawn estimation circuit

```
Pe=[-7    -7    -7]
```

The state estimation vector is obtained as $l^T = [-17.3333 \quad 7.6667 \quad 2.5000]$. The MATLAB™ program below gives the course in time of the real state variables of the system which have to be estimated, then calculates the vector l of the estimation circuit. Then according to Fig. 9.10 it simulates the evolution in time of the state estimation exciting the estimation circuit with the signal y as the input of the circuit. The program plots in one diagram the real state variables and their estimation, as well as the output signal and its estimated value.

```
clear
clc
num =6;
den =poly([-1,-2,-3])
P=tf(num,den)
[A,b,c,d]=tf2ss(num,den)
sys1=ss(A,b,c,d)
x0=[1;1;1]
t=0:0.05:6;
[y,t,x]=initial(sys1,x0,t);
figure(1)
plot(t,x),grid
Pe=[-7 -7 -7]
L=acker(A',c',Pe)
Aest=A-L'*c
sysest=ss(Aest,L',c,d)
x0est=[0;0;0]
[yest,t,xest]=lsim(sysest,y,t,x0est)
figure(2)
plot(t,x,t,xest),grid
figure(3)
plot(t,x(:,1),t,xest(:,1)),grid
figure(4)
plot(t,y,t,yest),grid
```

Plot the evolution in time of the first state variable and its estimated value (Fig. 9.11). The simulation shows that the state variables become settled quickly.

```
plot(t,[x(:,1),xest(:,1)]),grid
```

Prescribing appropriate poles of the estimation circuit, the settling process can be further accelerated and the transients of the estimation can be influenced.

Build the state estimation circuit also in SIMULINK™. The process and its model are built from the blocks *State-Space* of the *Continuous* library, and the *Matrix Gain* block of the library *Math Operations*. The separation of parameter c is needed because not only the output signal, but also the state variables have to be
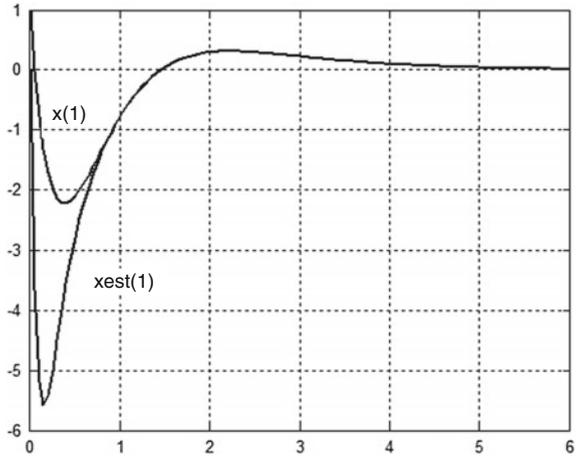
reached. The parameter b is also separated from the state model block, as the
derivatives of the state variables are modified, so the derivatives have to be also
available. (So in the *State-Space* blocks in the SIMULINK™ model (Fig. 9.12), the
parameters B and C are the identity matrices of the appropriate dimensions, and the
parameter d is a zero matrix. The process and its model can be the same, if the
process is known.) In the SIMULINK™ diagram shown in Fig. 9.12, the changes in
the real and the estimated state variables can be followed not only as the effect of
the unknown initial conditions, but also for the input and the disturbance signals. In
the example, after determining the state equation of the process and the calculation
of the vector l of the estimation circuit, the SIMULINK™ block can be run. In the
figure the parameters set for the *State-Space* blocks and the *Matrix Gain* blocks are
shown. Running the program, it can be observed in *Scope* that the estimated state
variables quickly follow the real state variables. As the variables are connected also
to *Workspace* blocks, the real and the estimated state variables can be plotted from
the MATLAB™ surface as well. For the course in time of the first state variable and
of its estimation, the result is the same as given in Fig. 9.11.

```
plot(t,x,t,xest),grid
```

Problem Set the values of the initial conditions to zero and the value of the output
disturbance to 1. Running the simulation, it can be seen that there is a static
deviation between the real and the estimated state variables. The input signal excites
the real and the estimation circuit the same way, therefore this excitation will not
distort the state estimation. But the output disturbance excites them differently, and
therefore a static error will appear in the estimation. To eliminate this deviation the
disturbance signal should be described by its state variables, then the state equation
should be enhanced by the state variables of the disturbance. Then the state esti-
mation could be executed for the extended system (but this extension will no be not
dealt with in more detail here).

Fig. 9.12 SIMULINK™ diagram of state estimation

9.4   State Feedback with State Estimation

State estimation (observer) and state feedback can be executed independently of each other (separation principle, textbook [1], Chap. 9). If the state variables are not available, then state feedback control can be realized by feeding back the estimated state variables with the state feedback vector **k** calculated for the original state variables. An important principle is that the dynamics of the state feedback circuit should be faster than the process dynamics, and the dynamics of the estimation circuit should be faster than the dynamics of the state feedback circuit to ensure that the state feedback would consider estimated state variables which approach quickly and well the state variables of the real system.

The block diagram of the state feedback system using an observer is given in Fig. 9.13. On the basis of the figure, the state equation of the system is

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & -bk^{\mathrm{T}} \\ lc^{\mathrm{T}} & A - lc^{\mathrm{T}} - bk^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} bk_{\mathrm{r}} \\ bk_{\mathrm{r}} \end{bmatrix} r$$

$$y = \begin{bmatrix} c^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + 0 \cdot r$$
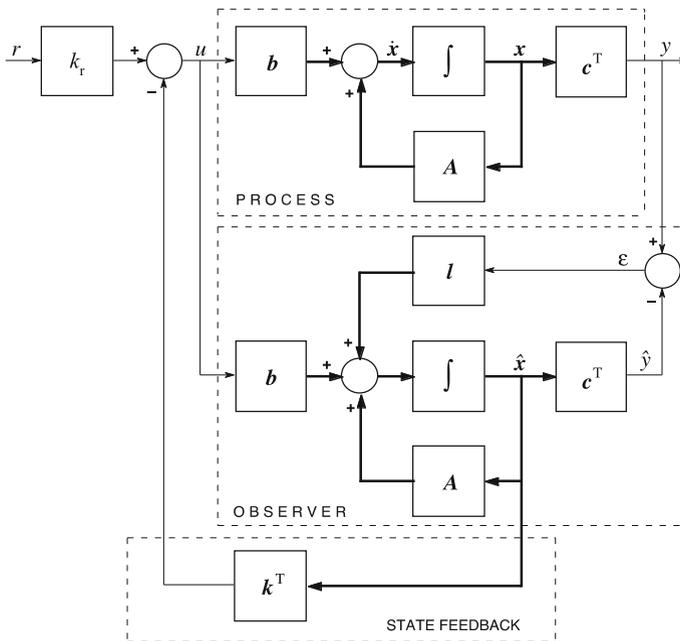
Fig. 9.13 State feedback from the estimated state variables

Example 9.5 Let us approximate the state variables of Example 9.1 by the estimated state variables calculated in Example 9.4, then feed back the approximate state variables by the state feedback constants calculated in Example 9.1. Plot in one diagram the step response of the output signals for the case when the feedback is taken from the original, and for the case where it is taken from the estimated state variables. The reference signal is a unit step and the initial conditions are [0.2 0.2 0.2]. The initial conditions of the observer states are supposed to be zeros.

The MATLAB™ program is

```
clear;clc
num =6;
den =poly([-1,-2,-3])
P=tf(num,den)
[A,b,c,d]=tf2ss(num,den)
sys1=ss(A,b,c,d)
%process
pk=[-6;-3+i*4;-3-i*4]
k=acker(A,b,pk)
```

```
sys2=ss(A-b*k,b,c,d)
kr=1/dcgain(sys2)
sys3=ss(A-b*k,b*kr,c,d)
%state feedback system
x0=[0.2;0.2;0.2]
t=0:0.05:3;
[y1,t,x] = initial(sys3,x0,t);
y2=step(sys3,t);
y=y1+y2;
%Output of the state feedback system
pe=[-7 -7 -7]
L=acker(A',c',pe)
Abvcs=[A -b*k;L'*c A-L'*c-b*k]
bbvcs=[b*kr;b*kr]
cbvcs=[c zeros(1,3)]
dbvcs=0
sys4=ss(Abvcs,bbvcs,cbvcs,dbvcs)
x0est=[0;0;0]
x0bvcs=[x0;x0est]
[y3,t,x3] = initial(sys4,x0bvcs,t);
y4=step(sys4,t);
y5=y3+y4;
%Output of the state feedback system from the estimated
%state variables
figure(1)
plot(t,y,t,y5),grid
figure(2)
plot(t,x3),grid
%The real and the estimated state variables
figure(3)
plot(t,x3(:,1),t,x3(:,4)),grid
figure(4)
plot(t,x3(:,2),t,x3(:,5)),grid
figure(5)
plot(t,x3(:,3),t,x3(:,6)),grid
```

Figure 9.14 gives the output signals of the state feedback systems when the feedback is taken from the original, and when it is taken from the estimated state variables. The overshoot is higher in the case when the feedback is taken from the estimated state variables. Figure 9.15 shows the first state variable and its estimation.

Fig. 9.14 Output signals
with feedback from the real
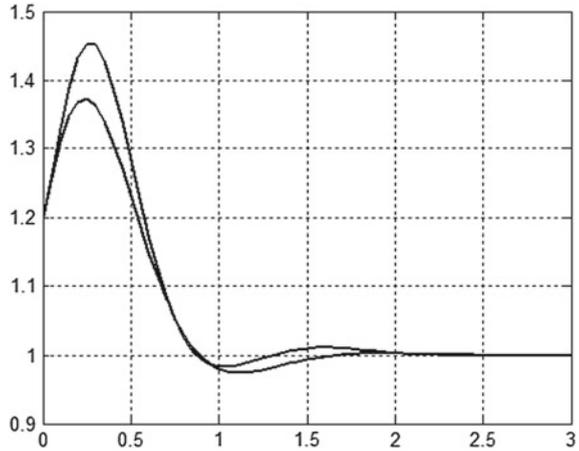and from the estimated state
variables

Fig. 9.15 The time course of
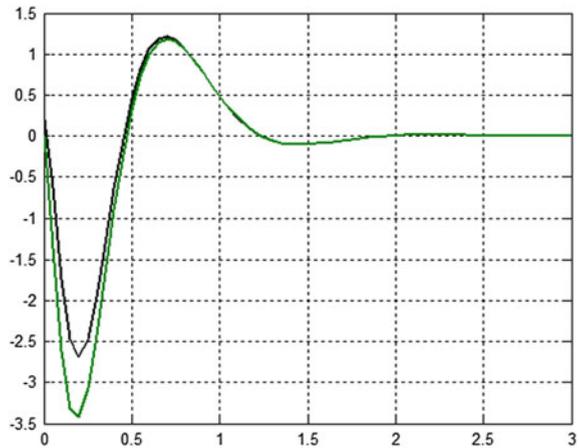the first state variable and its
estimation

Figure 9.16 gives the SIMULINK™ block diagram of the control system.
Running it for the given initial conditions and for unit step reference signal the
obtained results coincide with the results obtained in MATLAB™. With the output
disturbance, a static error does appear in the estimation of the state variables and
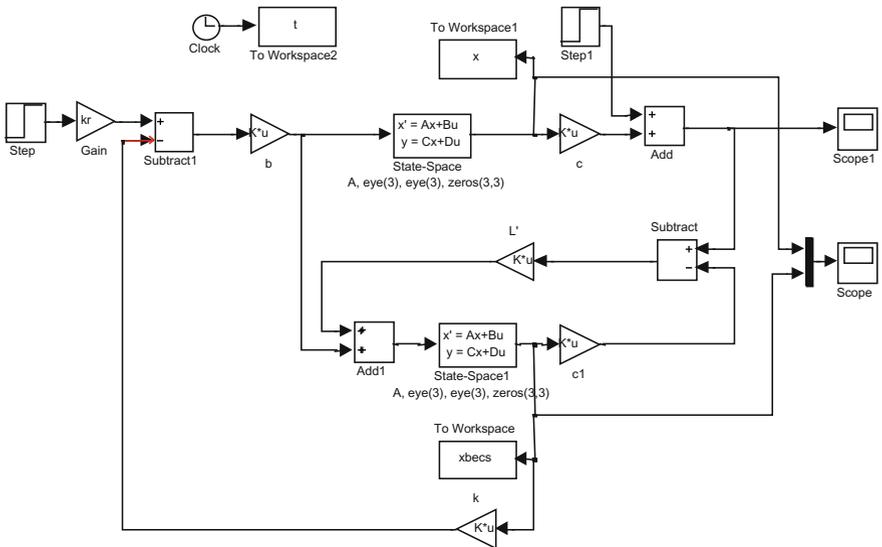also in the output signal.

Fig. 9.16 SIMULINK™ diagram of state feedback taken from the estimated state variables

Problem Build a SIMULINK™ block diagram when extending the system with the integrating state variable with state estimation and state feedback from the estimated state variables. Simulate the behaviour of the control system for unit step reference signal with the initial conditions given before. Analyse the disturbance rejection properties of the system in the case of an output disturbance.