EE5103 Computer Control

W.K. Ho

References

1. KJ Åström and Bjorn Wittenmark, *Computer Control Systems, Theory and Design*, 3nd Edition, Prentice Hall, 1997.

2. Liuping Wang *Model Predictive Control System Design and Implementation Using MATLAB*, Springer-Verlag, 2009

3. J.M. Maciejowski, *Predictive Control with Constraints*, Prentice Hall, 2002

Chapter 1

KALMAN FILTER

1.1 Thomas Algorithm

The Thomas algorithm is a simplified form of Gaussian elimination that can be used to solve tridiagonal system of equations. Consider the system of equations.

$$\begin{bmatrix} b_1 & c_1 & 0 & 0 \\ a_2 & b_2 & c_2 & 0 \\ 0 & a_3 & b_3 & c_3 \\ 0 & 0 & a_4 & b_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

Row 1:

$$b_1 x_1 + c_1 x_2 = z_1$$

Divide by b_1

$$x_1 + \frac{c_1}{b_1} x_2 = \frac{z_1}{b_1}$$

Rewrite the system of equations.

$$x_1 + \gamma_1 x_2 = \rho_1 \tag{1.1}$$

$$\begin{bmatrix} 1 & \gamma_1 & 0 & 0 \\ a_2 & b_2 & c_2 & 0 \\ 0 & a_3 & b_3 & c_3 \\ 0 & 0 & a_4 & b_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \rho_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

where

$$\begin{aligned} \gamma_1 &= \frac{c_1}{b_1} \\ \rho_1 &= \frac{z_1}{b_1} \end{aligned}$$

Row 2:

$$a_2 x_1 + b_2 x_2 + c_2 x_3 = z_2 \tag{1.2}$$

Equation (1.2) $-\ a_2 \times$ Equation (1.1) gives

$$(b_2 - a_2\gamma_1)x_2 + c_2x_3 = z_2 - a_2\rho_1$$

Divide by $(b_2 - a_2\gamma_1)$ gives

$$x_2 + \frac{c_2}{b_2 - a_2\gamma_1}x_3 = \frac{z_2 - a_2\rho_1}{b_2 - a_2\gamma_1}$$

Rewrite the system of equations.

$$x_2 + \gamma_2 x_3 \ = \ \rho_2 \tag{1.3}$$

$$\begin{bmatrix} 1 & \gamma_1 & 0 & 0 \\ 0 & 1 & \gamma_2 & 0 \\ 0 & a_3 & b_3 & c_3 \\ 0 & 0 & a_4 & b_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ z_3 \\ z_4 \end{bmatrix}$$

where

$$\gamma_2 \ = \ \frac{c_2}{b_2 - a_2\gamma_1}$$
$$\rho_2 \ = \ \frac{z_2 - a_2\rho_1}{b_2 - a_2\rho_1}$$

Row 3:

$$a_3x_2 + b_3x_3 + c_3x_4 = z_3 \tag{1.4}$$

Equation (1.4) $-\ a_3 \times$ Equation (1.3) gives

$$(b_3 - a_3\gamma_2)x_3 + c_3x_4 = z_3 - a_3\rho_2$$

Divide by $(b_3 - a_3\gamma_2)$ gives

$$x_3 + \frac{c_3}{b_3 - a_3\gamma_2}x_4 = \frac{z_3 - a_3\rho_2}{b_3 - a_3\gamma_2}$$

Rewrite the system of equations.

$$x_3 + \gamma_3 x_4 = \rho_3 \tag{1.5}$$

$$\begin{bmatrix} 1 & \gamma_1 & 0 & 0 \\ 0 & 1 & \gamma_2 & 0 \\ 0 & 0 & 1 & \gamma_3 \\ 0 & 0 & a_4 & b_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ z_4 \end{bmatrix}$$

where

$$\begin{aligned} \gamma_3 &= \frac{c_3}{b_3 - a_3\gamma_2} \\ \rho_3 &= \frac{z_3 - a_3\rho_2}{b_3 - a_3\gamma_2} \end{aligned}$$

Row 4:

$$a_4x_3 + b_4x_4 = z_4 \tag{1.6}$$

Equation (1.6) $- a_4 \times$ Equation (1.5) gives

$$(b_4 - a_4\gamma_3)x_4 = z_4 - a_4\rho_3$$

Divide by $(b_4 - a_4\gamma_3)$ gives

$$x_4 = \frac{z_4 - a_4\rho_3}{b_4 - a_4\gamma_3}$$

In general, for n equations,

$$\begin{bmatrix} b_1 & c_1 & 0 & \dots & & \\ a_2 & b_2 & c_2 & 0 & \dots & \\ 0 & a_3 & b_3 & c_3 & 0 & \dots \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & \dots & & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & \dots & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ \\ x_n \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ \\ \\ z_n \end{bmatrix}$$

the last state x_n

$$x_n = \frac{z_n - a_n\rho_{n-1}}{b_n - a_n\gamma_{n-1}} \tag{1.7}$$

$$\gamma_{n-1} = \frac{c_{n-1}}{b_{n-1} - a_{n-1}\gamma_{n-2}} \tag{1.8}$$

$$\rho_{n-1} = \frac{z_{n-1} - a_{n-1}\rho_{n-2}}{b_{n-1} - a_{n-1}\gamma_{n-2}} \tag{1.9}$$

1.2 Least-Squares Estimation

Solve simultaneous equations "approximately" for the unknowns when there are more equations than unknowns. For example 3 equations 2 unknowns θ_1 and θ_2:

$$\begin{aligned} \phi_{11}\theta_1 + \phi_{12}\theta_2 &= y_1 \\ \phi_{21}\theta_1 + \phi_{22}\theta_2 &= y_2 \\ \phi_{31}\theta_1 + \phi_{32}\theta_2 &= y_3 \end{aligned}$$

Organised in matrices give

$$\Phi\theta = Y \tag{1.10}$$

where

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \\ \phi_{31} & \phi_{32} \end{bmatrix}; \qquad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}; \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Choose the θ_1 and θ_2 that minimizes the error in the 3 equations. There many ways to define such an error, but the most convenient is the sum of squares:

$$\begin{aligned} V &= (y_1 - \phi_{11}\theta_1 - \phi_{12}\theta_2)^2 \\ &\quad +(y_2 - \phi_{21}\theta_1 - \phi_{22}\theta_2)^2 \\ &\quad +(y_3 - \phi_{31}\theta_1 - \phi_{32}\theta_2)^2 \\ &= (Y - \Phi\theta)^T(Y - \Phi\theta) \end{aligned} \tag{1.11}$$

If there is an exact solution to $\Phi\theta = Y$, the minimum error is $V = 0$ else the minimum is given by

$$\begin{aligned} \frac{dV}{d\hat{\theta}_1} &= -(\phi_{11}y_1 + \phi_{21}y_2 + \phi_{31}y_3) \\ &\quad +(\phi_{11}^2 + \phi_{21}^2 + \phi_{31}^2)\hat{\theta}_1 + (\phi_{11}\phi_{12} + \phi_{21}\phi_{22} + \phi_{31}\phi_{32})\hat{\theta}_2 = 0 \\ \frac{dV}{d\hat{\theta}_2} &= -(\phi_{12}y_1 + \phi_{22}y_2 + \phi_{32}y_3) \\ &\quad +(\phi_{11}\phi_{12} + \phi_{21}\phi_{22} + \phi_{31}\phi_{32})\ \hat{\theta}_1 + (\phi_{12}^2 + \phi_{22}^2 + \phi_{32}^2)\hat{\theta}_2 = 0 \end{aligned}$$

$$\begin{aligned} \frac{dV}{d\theta} &= -\Phi^T Y + \Phi^T\Phi\hat{\theta} = 0 \\ \hat{\theta} &= (\Phi^T\Phi)^{-1}\Phi^T Y \end{aligned} \tag{1.12}$$

if $\Phi^T\Phi$ is invertible.

Example 1

Consider the following simultaneous equation

$$\begin{aligned} x_1 &= x_0 \\ 0 &= x_0 \\ 1 &= x_1 \end{aligned}$$

Rewriting gives

$$\begin{aligned} 0 &= x_0 - x_1 \\ 0 &= x_0 \\ 1 &= x_1 \end{aligned}$$

Formulate as a least-squares problem gives

$$\Phi = \begin{bmatrix} 1 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \hat{\theta} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \qquad Y = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\begin{aligned}
\hat{\theta} &= \left(\begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\
&= \frac{1}{6} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{3} \\ \frac{2}{2} \end{bmatrix}
\end{aligned}$$

1.3 Estimation of State Variables

Consider the first-order process model

$$\begin{aligned} x(k+1) &= Ax(k) + w(k) \qquad (1.13) \\ y(k) &= Cx(k) + v(k) \qquad (1.14) \end{aligned}$$

which is also the model used by the Kalman filter where A, B, C and measurement $y(k)$ are known scalars; $w(k)$ and $v(k)$ are zero-mean independent Gaussian noises with variances R_1 and R_2 respectively. See Figure 1.1.

Consider $k = 0$, 1, 2, 3. Process of Equation (1.13) and (1.14) give

$$\begin{aligned} x(1) &= Ax(0) + w(0) \\ y(0) &= Cx(0) + v(0) \\ x(2) &= Ax(1) + w(1) \\ y(1) &= Cx(1) + v(1) \\ x(3) &= Ax(2) + w(2) \\ y(2) &= Cx(2) + v(2) \\ y(3) &= Cx(3) + v(3) \end{aligned}$$

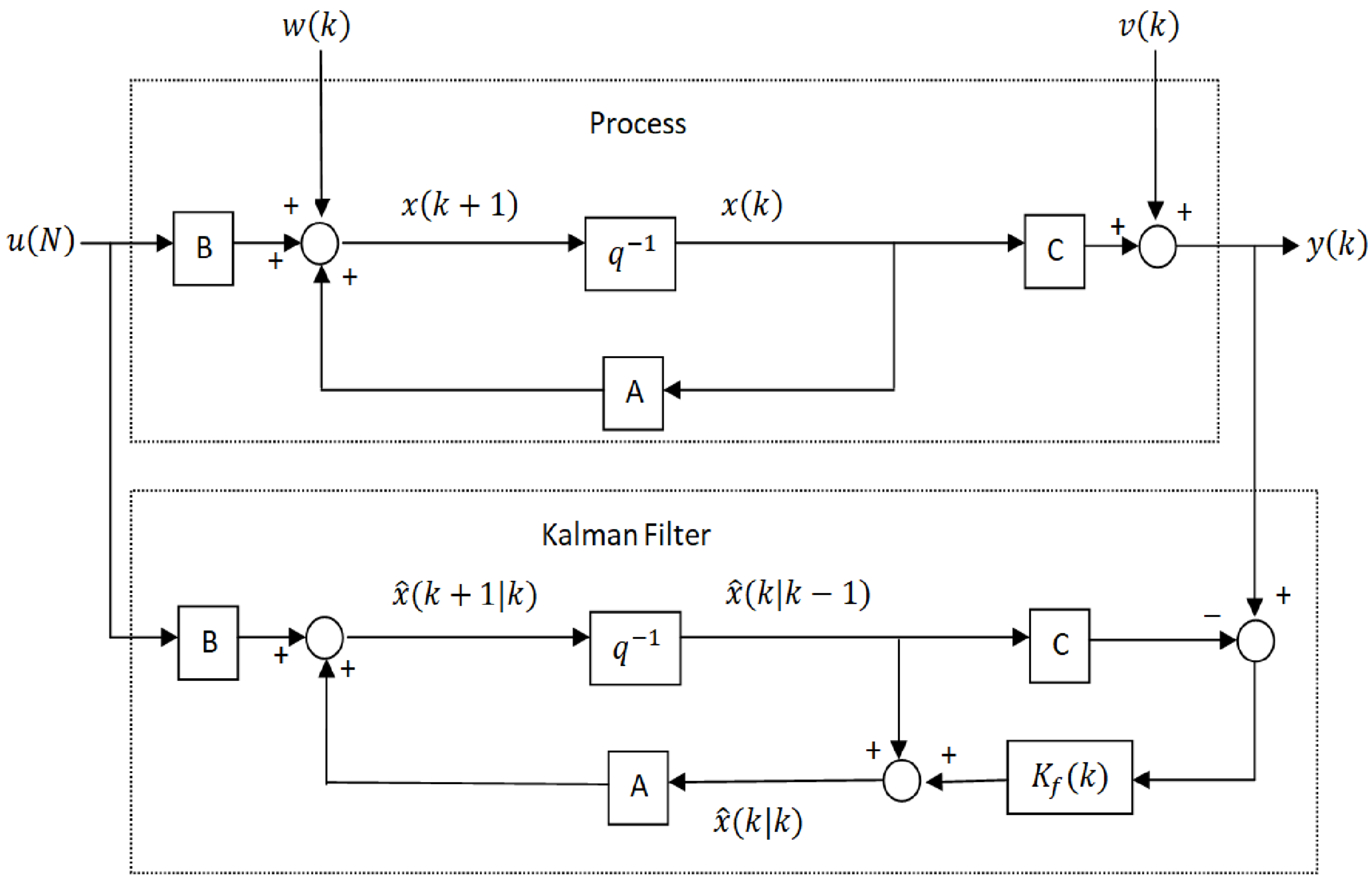

Figure 1.1: The Kalman gain $K_f(k)$ is a function of R_1 and R_2.

If we ignore the noise terms then there are 7 equations with 4 unknowns $x(k)$ for $k = 0, 1, 2, 3$:

$$\begin{aligned}
x(1) &= Ax(0) \\
y(0) &= Cx(0) \\
x(2) &= Ax(1) \\
y(1) &= Cx(1) \\
x(3) &= Ax(2) \\
y(2) &= Cx(2) \\
y(3) &= Cx(3)
\end{aligned}$$

and the least-squares estimates $\hat{x}(k)$ can be obtained by minimizing the following objective function

$$\begin{aligned}
J &= \frac{1}{2}\left[(\hat{x}(1) - A\hat{x}(0))^2 + (\hat{x}(2) - A\hat{x}(1))^2 + (\hat{x}(3) - A\hat{x}(2))^2\right] \\
&+ \frac{1}{2}\left[(y(0) - C\hat{x}(0))^2 + (y(1) - C\hat{x}(1))^2 + (y(2) - C\hat{x}(2))^2 + (y(3) - C\hat{x}(3))^2\right]
\end{aligned}$$

Modify the objective function as follows

$$\begin{aligned}
J &= \frac{1}{2R_1}\left[(\hat{x}(1) - A\hat{x}(0))^2 + (\hat{x}(2) - A\hat{x}(1))^2 + (\hat{x}(3) - A\hat{x}(2))^2\right] \\
&+ \frac{1}{2R_2}\left[(y(0) - C\hat{x}(0))^2 + (y(1) - C\hat{x}(1))^2 + (y(2) - C\hat{x}(2))^2 + (y(3) - C\hat{x}(3))^2\right] \quad (1.15)
\end{aligned}$$

to take the noise terms into consideration i.e. large variance is given less consideration.

Differentiate and equate to 0 to find the minimun J

$$
\begin{aligned}
\frac{\partial J}{\partial \hat{x}(0)} &= -\frac{A(\hat{x}(1) - A\hat{x}(0))}{R_1} - \frac{C(y(0) - C\hat{x}(0))}{R_2} = 0 \\
\frac{\partial J}{\partial \hat{x}(1)} &= \frac{\hat{x}(1) - A\hat{x}(0)}{R_1} - \frac{A(\hat{x}(2) - A\hat{x}(1))}{R_1} - \frac{C(y(1) - C\hat{x}(1))}{R_2} = 0 \\
\frac{\partial J}{\partial \hat{x}(2)} &= \frac{\hat{x}(2) - A\hat{x}(1)}{R_1} - \frac{A(\hat{x}(3) - A\hat{x}(2))}{R_1} - \frac{C(y(2) - C\hat{x}(2))}{R_2} = 0 \\
\frac{\partial J}{\partial \hat{x}(3)} &= \frac{\hat{x}(3) - A\hat{x}(2)}{R_1} - \frac{C(y(3) - C\hat{x}(3))}{R_2} = 0
\end{aligned}
$$

Write in matrix form

$$
\begin{bmatrix}
\frac{A^2}{R_1} + \frac{C^2}{R_2} & -\frac{A}{R_1} & 0 & 0 \\
-\frac{A}{R_1} & \frac{1}{R_1} + \frac{C^2}{R_2} + \frac{A^2}{R_1} & -\frac{A}{R_1} & 0 \\
0 & -\frac{A}{R_1} & \frac{1}{R_1} + \frac{C^2}{R_2} + \frac{A^2}{R_1} & -\frac{A}{R_1} \\
0 & 0 & -\frac{A}{R_1} & \frac{1}{R_1} + \frac{C^2}{R_2}
\end{bmatrix}
\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \\ \hat{x}(3) \end{bmatrix}
= \frac{C}{R_2}
\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix}
\tag{1.16}
$$

Let

$$\begin{aligned}\alpha &= -\frac{A}{R_1} &(1.17)\\ \beta &= \frac{1}{R_1}+\frac{C^2}{R_2}+\frac{A^2}{R_1} &(1.18)\\ z(k) &= \frac{C}{R_2}y(k) &(1.19)\end{aligned}$$

<u>Consider k samples</u>

$$\begin{bmatrix} \beta-\frac{1}{R_1} & \alpha & 0 & \dots & & & \\ \alpha & \beta & \alpha & 0 & \dots & & \\ 0 & \alpha & \beta & \alpha & 0 & & \dots \\ \vdots & & \ddots & \ddots & \ddots & & \\ 0 & \dots & & \alpha & \beta & & \alpha \\ 0 & \dots & & & \alpha & & \beta-\frac{A^2}{R_1} \end{bmatrix}\begin{bmatrix}\hat{x}(0)\\ \hat{x}(1)\\ \vdots\\ \\ \\ \hat{x}(k)\end{bmatrix}=\begin{bmatrix}z(0)\\ z(1)\\ \vdots\\ \\ \\ z(k)\end{bmatrix}$$

Use Thomas Algorithm (1.7), (1.8) and (1.9) to obtain expression for $\hat{x}(k)$.

$$\begin{aligned}\hat{x}(k) &= \left(\beta-\frac{A^2}{R_1}-\alpha\gamma(k-1)\right)^{-1}\{z(k)-\alpha\rho(k-1)\}\\ &= P(k)\{z(k)-\alpha\rho(k-1)\} &(1.20)\end{aligned}$$

where

$$\begin{aligned} P(k) &= \left(\beta - \frac{A^2}{R_1} - \alpha\gamma(k-1)\right)^{-1} && (1.21) \\ \gamma(k-1) &= \alpha(\beta - \alpha\gamma(k-2))^{-1} && (1.22) \\ \rho(k-1) &= (\beta - \alpha\gamma(k-2))^{-1}(z(k-1) - \alpha\rho(k-2)) && (1.23) \end{aligned}$$

See Example 4 for the statistical meaning of $P(k)$.

Consider $k-1$ samples

Replace k by $k-1$ in (1.20), (1.21), (1.22) and (1.23) to give

$$\begin{aligned} \hat{x}(k-1) &= \left(\beta - \frac{A^2}{R_1} - \alpha\gamma(k-2)\right)^{-1} (z(k-1) - \alpha\rho(k-2)) \\ &= P(k-1)\,(z(k-1) - \alpha\rho(k-2)) && (1.24) \end{aligned}$$

where

$$\begin{aligned} P(k-1) &= \left(\beta - \frac{A^2}{R_1} - \alpha\gamma(k-2)\right)^{-1} && (1.25) \\ \gamma(k-2) &= \alpha(\beta - \alpha\gamma(k-3))^{-1} && (1.26) \\ \rho(k-2) &= (\beta - \alpha\gamma(k-3))^{-1}(z(k-2) - \alpha\rho(k-3)) && (1.27) \end{aligned}$$

1.4 Recursive Algorithm

Equations (1.20), (1.21), (1.22), (1.23) for $\hat{x}(k)$ and (1.24), (1.25), (1.26), (1.27) for $\hat{x}(k-1)$ can be used to give a recursive algorithm for $\hat{x}(k)$ in terms of $\hat{x}(k-1)$

1.4.1 Recursion for $\hat{x}(k)$

From Equation (1.24)

$$z(k-1) - \alpha\rho(k-2) = P(k-1)^{-1}\hat{x}(k-1)$$

Substitute into Equation (1.23)

$$\rho(k-1) \;=\; (\beta - \alpha\gamma(k-2))^{-1}P(k-1)^{-1}\hat{x}(k-1) \tag{1.28}$$

From Equation (1.25)

$$\beta - \alpha\gamma(k-2) = \frac{A^2}{R_1} + P(k-1)^{-1}$$

Substutitue into Equation (1.28)

$$\rho(k-1) = \left(\frac{A^2}{R_1} + P(k-1)^{-1}\right)^{-1} P(k-1)^{-1}\hat{x}(k-1)$$

Substitute into Equation (1.20)

$$\hat{x}(k) = P(k)\left\{z(k) - \alpha\left(\frac{A^2}{R_1} + P(k-1)^{-1}\right)^{-1} P(k-1)^{-1}\hat{x}(k-1)\right\}$$

Add and substract $A\hat{x}(k-1)$

$$\begin{aligned} \hat{x}(k) &= A\hat{x}(k-1) + P(k) \\ &\quad \times \left\{z(k) - \alpha\left(\frac{A^2}{R_1} + P(k-1)^{-1}\right)^{-1} P(k-1)^{-1}\hat{x}(k-1) - P(k)^{-1}A\hat{x}(k-1)\right\} \end{aligned} \tag{1.29}$$

From Equation (1.21)

$$P(k)^{-1} = \beta - \frac{A^2}{R_1} - \alpha\gamma(k-1) \tag{1.30}$$

Substitute $\gamma(k-1)$ from Equaiton (1.22)

$$P(k)^{-1} = \beta - \frac{A^2}{R_1} - \alpha^2(\beta - \alpha\gamma(k-2))^{-1}$$

Substitute $\beta - a\gamma(k-2)$ from Equaiton (1.25)

$$P(k)^{-1} = \beta - \frac{A^2}{R_1} - \alpha^2 \left(\frac{A^2}{R_1} + P(k-1)^{-1} \right)^{-1} \tag{1.31}$$

Substitute into Equation (1.29)

$$\begin{aligned}
\hat{x}(k) &= A\hat{x}(k-1) + P(k) \\
&\times \left\{ z(k) - \alpha \left(\frac{A^2}{R_1} + P(k-1)^{-1} \right)^{-1} P(k-1)^{-1}\hat{x}(k-1) \right. \\
&\left. - \left(\beta - \frac{A^2}{R_1} - \alpha^2 \left(\frac{A^2}{R_1} + P(k-1)^{-1} \right)^{-1} \right) A\hat{x}(k-1) \right\} \\
\hat{x}(k) &= A\hat{x}(k-1) + P(k) \\
&\times \left\{ z(k) - \frac{C^2 A\hat{x}(k-1)}{R_2} \left[\frac{1}{C^2} \left(\frac{R_2}{a}\alpha \left(\frac{A^2}{R_1} + P(k-1)^{-1} \right)^{-1} P(k-1)^{-1} \right. \right. \right. \\
&\left. \left. \left. + \left(\beta - \frac{A^2}{R_1} - \alpha^2 \left(\frac{A^2}{R_1} + P(k-1)^{-1} \right)^{-1} \right) R_2 \right) \right] \right\}
\end{aligned}$$

The expression in the square bracket $= 1$. Hence

$$\hat{x}(k) = A\hat{x}(k-1) + P(k) \left(z(k) - \frac{C^2 A\hat{x}(k-1)}{R_2} \right) \tag{1.32}$$

Substitute $z(k)$ from Equation (1.19)

$$\hat{x}(k) \;=\; A\hat{x}(k-1) + K_f(k)(y(k) - CA\hat{x}(k-1)) \tag{1.33}$$

where

$$K_f(k) = \frac{P(k)C}{R_2} \tag{1.34}$$

In Kalman filter literature we denote the estimate $\hat{x}(k)$ at sample N obtained from $y(k)$ and $y(k-1)$ by $\hat{x}(k|k)$ (filter estimate) and $\hat{x}(k|k-1)$ (predicted estimate) respectively. Equation (1.33) can be written as

$$\hat{x}(k|k) \;=\; A\hat{x}(k-1|k-1) + K_f(k)(y(k) - CA\hat{x}(k-1|k-1)) \tag{1.35}$$

From Equation (1.13), since $w(k)$ is zero-mean

$$\hat{x}(k|k-1) = A\hat{x}(k-1|k-1) \tag{1.36}$$

and Equation (1.35) can be written as

$$\hat{x}(k|k) \;=\; \hat{x}(k|k-1) + K_f(k)(y(k) - C\hat{x}(k|k-1)) \tag{1.37}$$

Equation (1.36) gives

$$\hat{x}(k+1|k) = A\hat{x}(k|k) \tag{1.38}$$

Subtituting (1.37) into (1.38)

$$\hat{x}(k+1|k) \;=\; A\hat{x}(k|k-1) + K(k)(y(k) - C\hat{x}(k|k-1)) \tag{1.39}$$

where

$$K(k) = AK_f(k) \tag{1.40}$$

1.4.2 Recursion for $P(k)$

In (1.32), $\hat{x}(k)$ can be obtained from $\hat{x}(k-1)$. The recursive algorithm will be complete if $P(k)$ can also be obtained from $P(k-1)$.

Substitute for α and β from Equations (1.17) and (1.18) into (1.31)

$$\begin{aligned}
\frac{1}{P(k)} &= \frac{C^2}{R_2} + \frac{1}{R_1} - \frac{A^2}{R_1^2}\left(\frac{1}{P(k-1)} + \frac{A^2}{R_1}\right)^{-1} \\
&= \frac{C^2}{R_2} + \frac{1}{R_1} - \frac{A^2}{R_1^2}\left(\frac{A^2 P(k-1) + R_1}{P(k-1)R_1}\right)^{-1} \\
&= \frac{C^2}{R_2} + \frac{1}{R_1} - \frac{A^2}{R_1}\left(\frac{P(k-1)}{A^2 P(k-1) + R_1}\right) \\
&= \frac{C^2}{R_2} + \frac{1}{A^2 P(k-1) + R_1}
\end{aligned} \tag{1.41}$$

Equation (1.41) update the inverse of $P(k)$ from the inverse of $P(k-1)$. The non-inverse version can be obtained as follows. In Kalman Filter literature, $P(k)$ and $P(k-1)$ are written as $P(k|k)$ and $P(k-1|k-1)$ respectively to differentiate from a new variable $P(k|k-1)$ used to replace the denominator of the second term in (1.41) i.e.

$$P(k|k-1) = A^2 P(k-1|k-1) + R_1 \tag{1.42}$$

See Example 4 for the statistical meaning of $P(k|k-1)$.

Substitute Equation (1.42) into (1.41) give

$$\begin{aligned}\frac{1}{P(k|k)} &= \frac{C^2}{R_2} + \frac{1}{P(k|k-1)} \\ P(k|k) &= \frac{P(k|k-1)R_2}{P(k|k-1)C^2 + R_2} \qquad (1.43) \\ &= P(k|k-1) - \frac{P(k|k-1)^2 C^2}{P(k|k-1)C^2 + R_2} \qquad (1.44)\end{aligned}$$

Replace the index k by $k+1$ in Equation (1.42) gives

$$P(k+1|k) = A^2 P(k|k) + R_1 \qquad (1.45)$$

Substitute $P(k|k)$ from Equation (1.44) gives

$$P(k+1|k) = A^2 P(k|k-1) - \frac{(AP(k|k-1)C)^2}{P(k|k-1)C^2 + R_2} + R_1 \qquad (1.46)$$

From Equations (1.34), (1.40) and (1.43)

$$\begin{aligned}K_f(k) &= \frac{P(k|k-1)C}{P(k|k-1)C^2 + R_2} \qquad (1.47) \\ K(k) &= \frac{AP(k|k-1)C}{P(k|k-1)C^2 + R_2} \qquad (1.48)\end{aligned}$$

Substitute Equation (1.48) into (1.46) give

$$P(k+1|k) = A^2 P(k|k-1) - K(k)^2 (P(k|k-1)C^2 + R_2) + R_1 \qquad (1.49)$$

1.4.3 Process of Order n, $n > 1$

For a single-input-single-output process of order n where $n > 1$, we replace the scalars by vectors and matrices as follows

Variable	Dimension
$x(k)$	$n \times 1$
A	$n \times n$
B	$n \times 1$
C	$1 \times n$
P	$n \times n$
$u(k)$	1×1
$y(k)$	1×1
$w(k)$	$n \times 1$
$v(k)$	1×1
$R_1 = \mathbb{E}\left[w(k)w(k)^T\right]$	$n \times n$
$R_2 = \mathbb{E}\left[v(k)^2\right]$	1×1
$K_f(k)$	$n \times 1$
$K(k)$	$n \times 1$

1.4.4 The Kalman Filter

Consider the model

$$x(k+1) = Ax(k) + Bu(k) + w(k) \tag{1.50}$$
$$y(k) = Cx(k) + v(k) \tag{1.51}$$

where $w(k)$ and $v(k)$ are zero-mean independent Gaussian noise with covariance matrix R_1 and R_2 respectively. The Kalman Filter is given as

$$K_f(k) = P(k|k-1)C^T \left(CP(k|k-1)C^T + R_2\right)^{-1} \tag{1.52}$$
$$K(k) = \left(AP(k|k-1)C^T\right)\left(CP(k|k-1)C^T + R_2\right)^{-1} \tag{1.53}$$
$$\hat{x}(k|k) = \hat{x}(k|k-1) + K_f(k)(y(k) - C\hat{x}(k|k-1)) \tag{1.54}$$
$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + Bu(k) + K(k)(y(k) - C\hat{x}(k|k-1)) \tag{1.55}$$
$$P(k|k) = P(k|k-1) - P(k|k-1)C^T \left(CP(k|k-1)C^T + R_2\right)^{-1} CP(k|k-1) \tag{1.56}$$
$$P(k+1|k) = AP(k|k-1)A^T - K(k)\left(CP(k|k-1)C^T + R_2\right)K^T(k) + R_1 \tag{1.57}$$

The above Equations (1.52), (1.53), (1.54), (1.55), (1.56) and (1.57) correspond to Equations (1.47), (1.48), (1.37), (1.39), (1.44) and (1.49) of the scalar case respectively.

We derived the Kalman Filter without the input $Bu(k)$. Superposition can be used to include the input $Bu(k)$ as shown in Equations (1.50) and (1.55). The covariance matrix $P(0|-1)$ is usually chosen as the identity matrix, I_n, multiplied by a large number. Note that we can iterate (1.52) (1.53) (1.56) and (1.57) for $k = 0, 1, 2, \ldots$, to pre-computed the Kalman filter parameters $K_f(k)$, $K(k)$, $P(k|k)$ and $P(k+1|k)$ off-line i.e. without even taking measurement $y(k)$. They can be saved and then used when measurement $y(k)$ becomes available.

Example 2

Consider the first-order process model

$$x(k+1) = ax(k) + w(k) \tag{1.58}$$

$$y(k) = cx(k) + v(k) \tag{1.59}$$

which is also the model used by the Kalman filter where $w(k)$ and $v(k)$ are independent Gaussian noise with variance $R_1 = 1$ and $R_2 = 1$ respectively and $a = c = 1$. The Kalman Filter is initialized with $\hat{x}(0|-1) = 0$, and $P(0|-1)$ is chosen as ∞.

(a) Given measurements $y(0) = 0$, $y(1) = 1$, $y(2) = 2$, fill up the table below.

k	$y(k)$	$K_f(k)$	$K(k)$	$\hat{x}(k\|k)$	$\hat{x}(k+1\|k)$	$P(k\|k)$	$P(k+1\|k)$
0	0						
1	1						
2	2						

□

We can compute $K_f(k)$, $K(k)$, $\hat{x}(k|k)$, $\hat{x}(k+1|k)$, $P(k|k)$ and $P(k+1|k)$ from (1.52), (1.53), (1.54), (1.55), (1.56) and (1.57).

$\underline{k=0}$

$$K_f(0)=1, \qquad K(0)=1, \qquad \hat{x}(0|0)=0, \qquad \hat{x}(1|0)=0$$
$$P(0|0)=\frac{R_2P(0|-1)}{P(0|-1)C^2+R_2}=1, \qquad P(1|0)=R_1+R_2=2$$

$\underline{k=1}$

$$K_f(1)=\frac{2}{3}, \qquad K(1)=\frac{2}{3}, \qquad \hat{x}(1|1)=0+\frac{2}{3}(1-0)=\frac{2}{3}, \qquad \hat{x}(2|1)=0+\frac{2}{3}(1-0)=\frac{2}{3}$$
$$P(1|1)=\frac{R_2P(1|0)}{P(1|0)C^2+R_2}=\frac{2}{3}, \qquad P(2|1)=2+1-\frac{2}{3}(2+1)\frac{2}{3}=1\frac{2}{3}$$

The results for $k=2$ is given in Table 1.

Table 1

k	$y(k)$	$K_f(k)$	$K(k)$	$\hat{x}(k\|k)$	$\hat{x}(k+1\|k)$	$P(k\|k)$	$P(k+1\|k)$
0	0	1	1	0	0	1	2
1	1	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$1\frac{2}{3}$
2	2	$\frac{5}{8}$	$\frac{5}{8}$	$1\frac{1}{2}$	$1\frac{1}{2}$	$\frac{5}{8}$	$1\frac{5}{8}$

(b) Use batch least-squares to find $\hat{x}(1|1)$. □

Equation (1.16) gives

$$
\begin{aligned}
\begin{bmatrix} \frac{a^2}{R_1}+\frac{c^2}{R_2} & -\frac{a}{R_1} \\ -\frac{a}{R1} & \frac{1}{R_1}+\frac{c^2}{R_2} \end{bmatrix} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \end{bmatrix} &= \frac{c}{R_2} \begin{bmatrix} y(0) \\ y(1) \end{bmatrix} \\
\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \end{bmatrix} &= \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{3} \\ \frac{2}{3} \end{bmatrix} \\
\hat{x}(1|1) &= \frac{2}{3}
\end{aligned}
$$

(c) Use batch least-squares to find $\hat{x}(2|2)$. $\square$

Equation (1.16) gives

$$\begin{aligned}
\begin{bmatrix} \frac{a^2}{R_1}+\frac{c^2}{R_2} & -\frac{a}{R_1} & 0 \\ -\frac{a}{R_1} & \frac{1}{R_1}+\frac{c^2}{R_2}+\frac{a^2}{R_1} & -\frac{a}{R_1} \\ 0 & -\frac{a}{R_1} & \frac{1}{R_1}+\frac{c^2}{R_2} \end{bmatrix} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} &= \frac{c}{R_2} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix} \\
\begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \\
\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} &= \begin{bmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{2} \\ 1 \\ 1\frac{1}{2} \end{bmatrix} \\
\hat{x}(2|2) &= 1\frac{1}{2}
\end{aligned}$$

(d) Find the Expectation and Variance of the estimation error $x(1) - \hat{x}(1|1)$. Compare your results with $P(1|1)$.

Use Equations (1.52), (1.53), (1.54), (1.55), (1.56) and (1.57).

Find $\hat{x}(1|1)$

$$\begin{aligned} K(0) &= \frac{a}{c} \\ \hat{x}(1|0) &= a\hat{x}(0|-1) + \frac{a}{c}\left(y(0) - c\hat{x}(0|-1)\right) = \frac{a}{c}y(0) \\ \hat{x}(1|1) &= \hat{x}(1|0) + K_f(1)\left(y(1) - c\hat{x}(1|0)\right) = \frac{a}{c}y(0) + K_f(1)(y(1) - ay(0)) \end{aligned}$$

Find $x(1)$

$$x(1) = ax(0) + w(0) = a\left(\frac{y(0) - v(0)}{c}\right) + w(0)$$

Find $\hat{x}(1|1) - x(1)$

$$\begin{aligned} \hat{x}(1|1) - x(1) &= K_f(1)(y(1) - ay(0)) + \frac{a}{c}v(0) - w(0) \\ &= K_f(1)\left\{c[ax(0) + w(0)] + v(1) - a[cx(0) + v(0)]\right\} + \frac{a}{c}v(0) - w(0) \\ &= K_f(1)\left\{cw(0) + v(1) - av(0)\right\} + \frac{a}{c}v(0) - w(0) \\ &= w(0)[cK_f(1) - 1] + v(0)[\frac{a}{c} - aK_f(1)] + v(1)K_f(1) \\ E[\hat{x}(1|1) - x(1)] &= 0 \\ E[(x(1) - \hat{x}(1|1))^2] &= R_1[cK_f(1) - 1]^2 + R_2[\frac{a}{c} - aK_f(1)]^2 + R_2K_f(1)^2 = \frac{2}{3} = P(1|1) \end{aligned}$$

Example 3

Consider the first-order process model

$$\begin{aligned} x(k+1) &= ax(k) + w(k) && (1.60) \\ y(k) &= cx(k) + v(k) && (1.61) \end{aligned}$$

where $w(k)$ and $v(k)$ are independent Gaussian noise with variance R_1 and R_2 respectively and $c = 1$.

(a) Given measurements $y(0)$, $y(1)$, $y(2)$, $R_1 = 1$, $R_2 = 0$, find the batch least-squares estimate of $\hat{x}(0)$, $\hat{x}(1)$, $\hat{x}(2)$.

Equation (1.16) gives

$$\begin{bmatrix} \frac{a^2}{R_1} + \frac{1}{R_2} & -\frac{a}{R_1} & 0 \\ -\frac{a}{R_1} & \frac{1}{R_1} + \frac{1}{R_2} + \frac{a^2}{R_1} & -\frac{a}{R_1} \\ 0 & -\frac{a}{R_1} & \frac{1}{R_1} + \frac{1}{R_2} \end{bmatrix} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \frac{1}{R_2} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix}$$

$$\begin{aligned} \begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} &= \frac{1}{a^2R_2(a^2R_2 + R_2 + 2R_1) + (R_1 + R_2)^2} \\ &\times \begin{bmatrix} a^2R_1R_2 + (R_1+R_2)^2 & aR_2(R_1+R_2) & a^2R_2^2 \\ aR_2(R_1+R_2) & a^2R_2(R_1+R_2) + R_1(R_1+R_2) & aR_2(R_1 + a^2R_2) \\ a^2R_2^2 & aR_2(R_1+a^2R_2) & a^2R_2(a^2R_2 + 2R_1) + R_1(R_1+R_2) \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix} \qquad (1.62) \end{aligned}$$

Let $R_1 = 1$, $R_2 = 0$ in Equation (1.62) gives

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix}$$

Note: $R_1 = 1$, $R_2 = 0$ means there is model uncertainty but no measurement noise. In other words, the measurement is exact and hence we take the measurement as the estimation result i.e. $\hat{x}(0) = y(0)$, $\hat{x}(1) = y(1)$, $\hat{x}(2) = y(2)$.

(b) Given measurements $y(0)$, $y(1)$, $y(2)$, $R_1 = 0$, $R_2 = 1$, find the batch least-squares estimate of $\hat{x}(0)$, $\hat{x}(1)$, $\hat{x}(2)$.

Let $R_1 = 0$, $R_2 = 1$ in Equation (1.62) gives

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \frac{1}{a^4 + a^2 + 1} \begin{bmatrix} 1 & a & a^2 \\ a & a^2 & a^3 \\ a^2 & a^3 & a^4 \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix} \tag{1.63}$$

Note:

(1) $\hat{x}(2) = a\hat{x}(1)$, $\hat{x}(1) = a\hat{x}(0)$ which is the process model in Equation (1.60) since $R_1 = 0$ means

there is no process model uncertainty. $R_2 = 1$ means there is measurement noise hence we take the process model as the estimation results.

(2) If $a = 1$ then $\hat{x}(0) = \hat{x}(1) = \hat{x}(2) = \frac{y(0)+y(1)+y(2)}{3}$. The estimation result is the least-squares estimate or average of the measurements.

(c) Given $y(0) = 3$, $y(1) = 1$, $y(2) = 0$, $a = 0.5$, plot the results in Part (a) and Part (b). Given $R_1 = R_2 = 1$ find and superimpose the least-squares estimate of $\hat{x}(0)$, $\hat{x}(1)$, $\hat{x}(2)$ on the same plot. $\square$

For Part (a)

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 0 \end{bmatrix}$$

For Part (b)

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \begin{bmatrix} 2.67 \\ 1.33 \\ 0.67 \end{bmatrix}$$

For $R_1 = R_2 = 1$ in Part (c)

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \hat{x}(2) \end{bmatrix} = \begin{bmatrix} 2.86 \\ 1.14 \\ 0.29 \end{bmatrix}$$

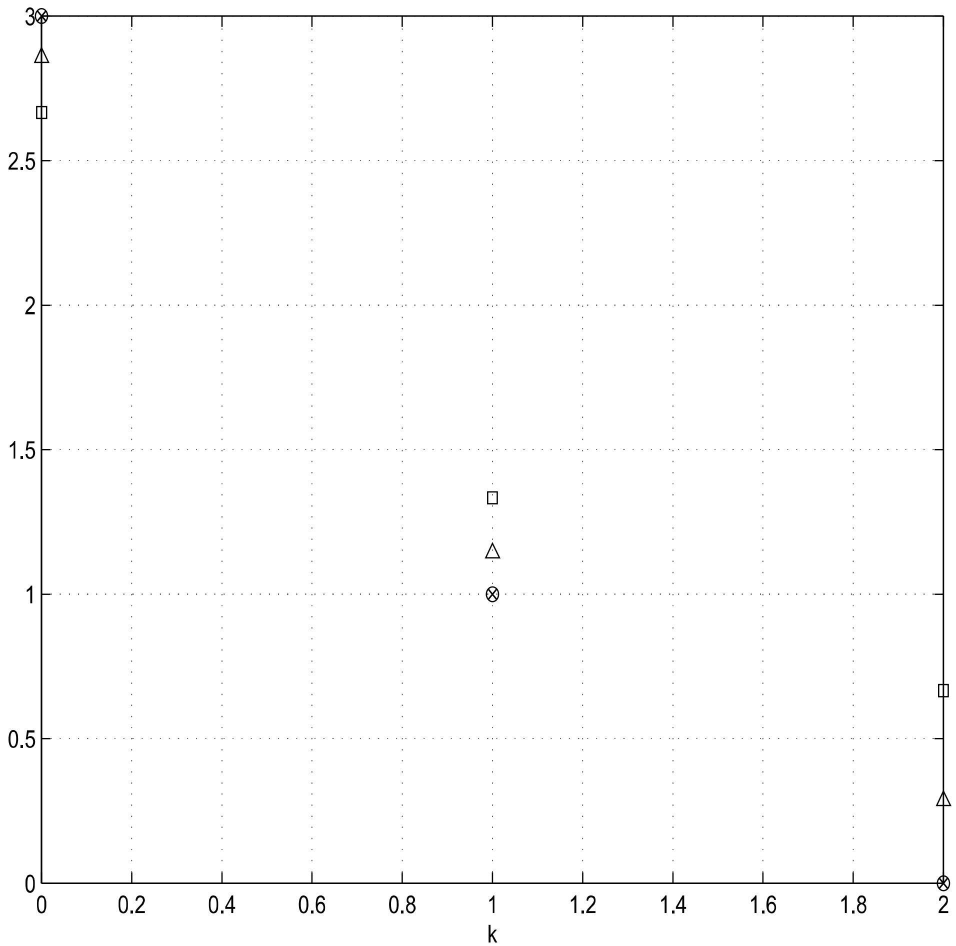

Figure 1.2: Cross: $y(k)$; circle: Part (a) $R_1 = 1, R_2 = 0$; square: Part (b) $R_1 = 0, R_2 = 1$, triangle: Part (c) $R_1 = R_2 = 1$.

Example 4

Consider the first-order process model (1.58) and (1.59) in Example 2. Show that (a) if $P(k+1|k) = \mathbb{E}[x(k+1) - \hat{x}(k+1|k)]^2$ and $P(k|k) = \mathbb{E}[(x(k) - \hat{x}(k|k)]^2$ then the covariance update are given by Equations (1.44) and (1.45) and (b) $P(0|0) = \mathbb{E}[x(0) - \hat{x}(0|0)]^2$ for $P(0|-1) = \infty$. $\square$

(a) Using (1.58) and (1.36)

$$\begin{aligned}
P(k+1|k) &= \mathbb{E}[x(k+1) - \hat{x}(k+1|k)]^2 \\
&= \mathbb{E}[ax(k) + w(k) - a\hat{x}(k|k)]^2 \\
&= a^2\mathbb{E}[x(k) - \hat{x}(k|k)]^2 + \mathbb{E}w(k)^2 - 2a\mathbb{E}[w(k)x(k)] + 2a\mathbb{E}[w(k)\hat{x}(k|k)] \\
&= a^2 P(k|k) + R_1
\end{aligned}$$

which is (1.45) since $w(k)$, $x(k)$ and $\hat{x}(k|k)$ are independent and $w(k)$ is zero-mean, hence the expectations of their products are 0.

Let $\tilde{x}(k) = x(k) - \hat{x}(k|k)$ and using (1.58), (1.59) and (1.35)

$$\begin{aligned}
\tilde{x}(k) &= ax(k-1) - a\hat{x}(k-1|k-1) + w(k-1) \\
&\quad + K_f(k)ca\hat{x}(k-1|k-1) - K_f(k)[cx(k) + v(k)]
\end{aligned}$$

Using (1.58) to replace $x(k)$ gives

$$\begin{aligned}
\tilde{x}(k) &= ax(k-1) - a\hat{x}(k-1|k-1) + w(k-1) \\
&\quad +K_f(k)ca\hat{x}(k-1|k-1) - K_f(k)[cax(k-1) + cw(k-1) + v(k)] \\
&= [1 - K_f(k)c]a\tilde{x}(k-1) + [1 - K_f(k)c]w(k-1) - K_f(k)v(k)
\end{aligned}$$

Writing $F = 1 - K_f(k)c$ for ease of notation

$$\begin{aligned}
\tilde{x}(k) &= Fa\tilde{x}(k-1) + Fw(k-1) - K_f(k)v(k) \\
\tilde{x}(k)^2 &= (Fa\tilde{x}(k-1))^2 + (Fw(k-1))^2 + (K_f(k)v(k))^2 \\
&\quad F^2a\tilde{x}(k-1)w(k-1) - FaK_f(k)\tilde{x}(k-1)v(k) + F^2aw(k-1)\tilde{x}(k-1) \\
&\quad FKw(k-1)v(k) - K_f(k)aFv(k)\tilde{x}(k-1) - K_f(k)Fv(k)w(k-1)
\end{aligned}$$

Since $\tilde{x}(k-1)$, $w(k-1)$ and $v(k)$ are independent and $w(k-1)$ and $v(k)$ are zero-mean, the expected value of any product of 2 different random numbers will be 0. There are such products in the last 6 terms on the RHS of the above equation. Only the expected values of the first 3 terms on the RHS remains.

$$\begin{aligned}
P(k|k) = \mathbb{E}\tilde{x}(k)^2 &= F^2a^2\mathbb{E}\tilde{x}(k-1)^2 + F^2\mathbb{E}w(k-1)^2 + K_f(k)^2\mathbb{E}v(k)^2 \\
&= F^2a^2P(k-1|k-1) + F^2R_1 + K_f(k)^2R_2 \\
&= F^2(a^2P(k-1|k-1) + R_1) + K_f(k)^2R_2
\end{aligned}$$

Using (1.45) and reinstating $F = 1 - K_f(k)c$

$$\begin{aligned} P(k|k) &= F^2 P(k|k-1) + K_f(k)^2 R_2 \\ &= (1 - K_f(k)c)^2 P(k|k-1) + K_f(k)^2 R_2 \\ &= P(k|k-1) - \frac{P(k|k-1)^2 c^2}{P(k|k-1)c^2 + R_2} \end{aligned}$$

which is (1.44) after substituing for $K_f(k)$ from (1.53).

(b) For $P(0|-1) = \infty$, (1.43) gives

$$P(0|0) = \frac{R_2}{c^2}$$

Equation (1.52) gives

$$K_f(0) = \frac{1}{c}$$

Equation (1.54) gives

$$\hat{x}(0|0) = \frac{y(0)}{c}$$

Substitute for $y(0)$ from Equation (1.14) gives

$$
\begin{aligned}
x(0)-\hat{x}(0|0) &= x(0)-\frac{1}{c}(cx(0)+v(0)) = \frac{v(0)}{c} \\
\mathbb{E}[x(0)-\hat{x}(0|0)]^2 &= \mathbb{E}\frac{v(0)^2}{c^2} = \frac{R_2}{c^2} = P(0|0)
\end{aligned}
$$

Note:

(1) $\mathbb{E}[x(0)-\hat{x}(0|0)]^2 = P(0|0)$, $\mathbb{E}[x(1)-\hat{x}(1|0)]^2 = P(1|0)$ can be found from (1.45), $\mathbb{E}[x(1)-\hat{x}(1|1)]^2 = P(1|1)$, (1.44) and so on. Equation (1.57) was derived from (1.45) and can be used in place (1.45).

(2) For $n > 1$ states, the scalars are replaced by vectors and matrices and we can show that in general
$P(k|k) = \mathbb{E}\left\{[(x(k)-\hat{x}(k|k)][(x(k)-\hat{x}(k|k)]^T\right\}$ and
$P(k+1|k) = \mathbb{E}\left\{[(x(k+1)-\hat{x}(k+1|k)][(x(k+1)-\hat{x}(k+1|k)]^T\right\}$.

For example for a 2 states system

$$
\begin{aligned}
P(k|k) &= \mathbb{E}\left\{ \begin{bmatrix} x_1(k) - \hat{x}_1(k|k) \\ x_2(k) - \hat{x}_2(k|k) \end{bmatrix} \begin{bmatrix} x_1(k) - \hat{x}_1(k|k) & x_2(k) - \hat{x}_2(k|k) \end{bmatrix} \right\} \\
&= \begin{bmatrix} \mathbb{E}\{(x_1(k) - \hat{x}_1(k|k))^2\} & \mathbb{E}\{(x_1(k) - \hat{x}_1(k|k))(x_2(k) - \hat{x}_2(k|k))\} \\ \mathbb{E}\{(x_2(k) - \hat{x}_2(k|k))(x_1(k) - \hat{x}_1(k|k))\} & \mathbb{E}\{(x_2(k) - \hat{x}_2(k|k))^2\} \end{bmatrix}
\end{aligned}
$$

Example 5

The true state position $x_1(k)$ and velocity $x_2(k)$ of a moving target are given by the following equations

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix} w(k) \tag{1.64}$$

$$y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + v(k) \tag{1.65}$$

where $w(k)$ and $v(k)$ are zero-mean independent Gaussian random variables with variances $\sigma_w = 1$ and $\sigma_v = 1$ respectively. The sampling period is $T = 1$ and the initial conditions are $x_1(0) = x_2(0) = 0$. Design a Kalman filter using the model of (1.64) and (1.65) and initial conditions $\hat{x}_1(0|-1) = \hat{x}_2(0|-1) = 0$ and $P(0|-1) = 1 \times 10^5 I_2$. Simulate one run of $k = 0$, 1, and 2.

(a) Repeat the simulation for 100 runs and plot $x_1(k) - \hat{x}_1(k)$. □

(b) Repeat the simulation for 10,000 runs and show that

$$P(2|2) = \mathbb{E}\left\{[x(2) - \hat{x}(2|2)]\,[x(2) - \hat{x}(2|2)]^T\right\}$$

(a)

$$
\begin{aligned}
R_1 &= \mathbb{E}\left\{\begin{bmatrix} 0.5T^2w(k) \\ Tw(k) \end{bmatrix}\begin{bmatrix} 0.5T^2w(k) & Tw(k) \end{bmatrix}\right\} \\
&= \mathbb{E}\left\{\begin{bmatrix} 0.25T^4w(k)^2 & 0.5T^3w(k)^2 \\ 0.5T^3w(k)^2 & T^2w(k)^2 \end{bmatrix}\right\} \\
&= \begin{bmatrix} 0.25T^4\mathbb{E}w(k)^2 & 0.5T^3\mathbb{E}w(k)^2 \\ 0.5T^3\mathbb{E}w(k)^2 & T^2\mathbb{E}w(k)^2 \end{bmatrix} \\
&= \begin{bmatrix} 0.25 & 0.5 \\ 0.5 & 1 \end{bmatrix} \\
R_2 &= \mathbb{E}v(k)^2 = 1
\end{aligned}
$$

The matlab program is given below. The plot is given in Figure 1.3.

(b) In the matlab program, change “Nrun=100” to “Nrun=10000” and “plot([0:N-1],x(1,1:N) . . .” to “%plot([0:N-1],x(1,1:N) . . .” The simulation gives

$$P(2|2) = \mathbb{E}\left\{[x(2) - \hat{x}(2|2)]\,[x(2) - \hat{x}(2|2)]^T\right\} = \begin{bmatrix} 0.8 & 0.6 \\ 0.6 & 1.1 \end{bmatrix}$$

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Kalman Filter Solution
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear
N=3;      %number of samples
Nrun=100; %number of runs
T=1;      %sampling interval
sw=1;     % sigma_w
sv=1;     % sigma_v
%%%%%%%%%
% Process
%%%%%%%%%
A=[1 T; 0 1];                      %process matrix
C=[1 0];                           %process matrix
R1=sw^2*[T^4/4 T^3/2; T^3/2 T^2]; %covariance of process noise
R2=sv^2;                           %covariance of measurement noise
Pm(:,:,1)=1e5*eye(2);              %P(0|-1)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculation of Kalman Gains
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for k=1:N
Kf(:,k)=Pm(:,:,k)*C'*(C*Pm(:,:,k)*C'+R2)^(-1); %Kf(:,k)=Kf(k)
K(:,k)=(A*Pm(:,:,k)*C')*(C*Pm(:,:,k)*C'+R2)^(-1); %K(:,k)=K(k)
P(:,:,k)=Pm(:,:,k)-(Pm(:,:,k)*C')*(C*Pm(:,:,k)*C'+R2)^(-1)*C*Pm(:,:,k); %P(:,:,k)=P(N|N)
Pm(:,:,k+1)=A*Pm(:,:,k)*A'+R1-K(:,k)*(C*Pm(:,:,k)*C'+R2)*K(:,k)'; %Pm(:,:,k+1)=P(N+1|N)
end
plot(0,0) %plot a point at (0,0)
hold on   %hold the plot for further plotting
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Initializaton
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
w=random('normal',0,sw,Nrun,N); %process noise
v=random('normal',0,sv,Nrun,N); %measurement noise
for run=1:Nrun
xhm(:,1)=[0 0]'; %xhm(:,1)=x(0|-1)
x(:,1)=[0 0]';   %x(:,1)=x(0)
for k=1:N
%%%%%%%%%
% Process
%%%%%%%%%
x(:,k+1)=A*x(:,k)+[T^2/2 T]'*w(run,k); %x(:,k+1)=x(k+1)
y(k)=C*x(:,k)+v(run,k);       %y(k)=y(k)
%%%%%%%%%%%%%%%
% Kalman Filter
%%%%%%%%%%%%%%%
xh(:,k)=xhm(:,k)+Kf(:,k)*(y(k)-C*xhm(:,k));     %xh(:,k)=x^hat(k|k)
xhm(:,k+1)=A*xhm(:,k)+K(:,k)*(y(k)-C*xhm(:,k)); %xhm(:,k+1)=x^hat(k+1|k)
end
plot([0:N-1],x(1,1:N)-xh(1,1:N),'x'); %plot x(k)-x^hat(k|k)
xs(:,run)=x(:,N);   %x(N) for each run is saved in xs(:,run)
xhs(:,run)=xh(:,N); %x^hat(N|N) for each run is save in xhs(:,run)
end
hold off    %release the plot
grid        %draw grid on the plot
xlabel('k') %x-axis label
Pf=P(:,:,N) %print P(N|N)
%%%%%%%%%%%%%%%%%%%%%%%%
% Compute E{x(k)^T*x(k)}
%%%%%%%%%%%%%%%%%%%%%%%%
```

```
Covariance=[mean(((xs(1,:)-xhs(1,:))).^2) mean((xs(1,:)-xhs(1,:)).*(xs(2,:)-xhs(2,:)));
   mean((xs(2,:)-xhs(2,:)).*(xs(1,:)-xhs(1,:))) mean(((xs(2,:)-xhs(2,:))).^2)]
%print E{[x(2)-xh(2)][x(2)-xh(2)]^T}
```

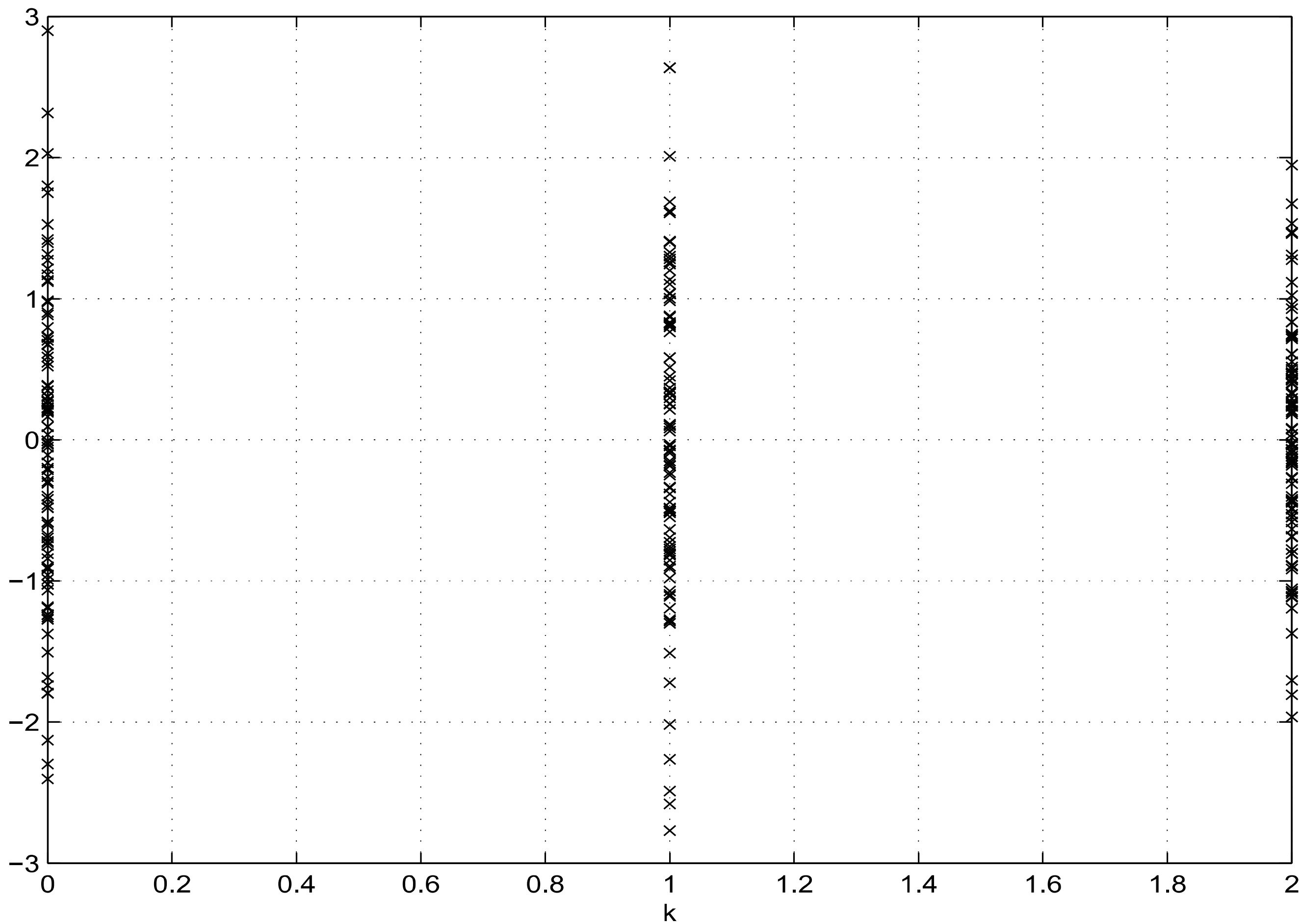

Figure 1.3: $x_1(k) - \hat{x}_1(k)$ for $k = 0$, 1 and 2.

Chapter 2

Model Predictive Control

Model Predictive Control (MPC) is the only advanced control technique — that is, more advanced than standard Proportional-Integral-Derivative (PID) control — to have had a significant and widespread impact on industrial process control.

2.1 State-Space Model Augmented with an Integrator

Consider the single-input and single-out process

$$
\begin{aligned}
x_p(k+1) &= A_p x_p(k) + B_p u(k) && (2.1)\\
y(k) &= C_p x_p(k) && (2.2)
\end{aligned}
$$

where u is the input, y is the output, and x_p is the state variable vector with dimension n_p.

To introduce an integrator into the controller, a standard practice is to augment the process with an integrator by making $\Delta u(k)$ the imaginary input and output of the process and controller respectively where

$$
u(k) = u(k-1) + \Delta u(k).
$$

Take z-transform

$$
\begin{aligned}
U(z) &= z^{-1}U(z) + \Delta U(z)\\
&= \frac{1}{1-z^{-1}}\Delta U(z)
\end{aligned}
$$

Note that in real implementation, $u(k)$ is the real input and output of the process and controller respectively (see Figure 2.2). Since $u(k)$ is the accumulation of $\Delta u(k)$, it is the discrete-time integration of $\Delta u(k)$.

Let $\Delta x_p(k) = x_p(k) - x_p(k-1)$ and using (2.1) gives

$$\begin{aligned} x_p(k+1) - x_p(k) &= A_p(x_p(k) - x_p(k-1)) + B_p(u(k) - u(k-1)) \\ \Delta x_p(k+1) &= A_p \Delta x_p(k) + B_p \Delta u(k) \end{aligned} \tag{2.3}$$

Equation (2.2) gives

$$\begin{aligned} y(k+1) - y(k) &= C_p(x_p(k+1) - x_p(k)) \\ &= C_p \Delta x_p(k+1) \end{aligned}$$

Using (2.3) gives

$$y(k+1) = y(k) + C_p A_p \Delta x_p(k) + C_p B_p \Delta u(k) \tag{2.4}$$

Putting together (2.3) and (2.4) leads to an augmented state-space model

$$\overbrace{\begin{bmatrix} \Delta x_p(k+1) \\ y(k+1) \end{bmatrix}}^{x(k+1)} = \overbrace{\begin{bmatrix} A_p & o_p^T \\ C_pA_p & 1 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} \Delta x_p(k) \\ y(k) \end{bmatrix}}^{x(k)} + \overbrace{\begin{bmatrix} B_p \\ C_pB_p \end{bmatrix}}^{B} \Delta u(k) \tag{2.5}$$

$$y(k) = \overbrace{\begin{bmatrix} o_p & 1 \end{bmatrix}}^{C} \overbrace{\begin{bmatrix} \Delta x_p(k) \\ y(k) \end{bmatrix}}^{x(k)} \tag{2.6}$$

where

$$x(k) = \begin{bmatrix} \Delta x_p(k) \\ y(k) \end{bmatrix} \tag{2.7}$$

is the new state vector and and $o_p = \overbrace{\begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix}}^{n_p}$. The triplet (A, B, C) is called the augmented model which will be used in the design of MPC.

Example 6

Suppose that a first-order process is described by

$$\begin{aligned} x_p(k+1) &= a_p x_p(k) + b_p u(k) \\ y(k) &= c_p x_p(k) \end{aligned}$$

(a) Find the transfer function $\frac{Y(z)}{U(z)}$. $\square$

$$\begin{aligned} zX_p(z) &= a_p X_p(z) + b_p U(z) \\ Y(z) &= c_p X_p(z) \\ \frac{Y(z)}{U(z)} &= \frac{c_p b_p}{z - a_p} \\ Y(z) &= \frac{c_p b_p}{z - a_p} U(z) \end{aligned}$$

(b) Find the augmented state-space model and the transfer function $\frac{Y(z)}{\Delta u(z)}$ of the augmented model. $\square$

Equation (2.5) gives

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ y(k) &= Cx(k) \end{aligned}$$

where

$$\begin{aligned}
A &= \begin{bmatrix} a_p & 0 \\ c_p a_p & 1 \end{bmatrix} \\
B &= \begin{bmatrix} b_p \\ c_p b_p \end{bmatrix} \\
C &= \begin{bmatrix} 0 & 1 \end{bmatrix} \\
x(k) &= \begin{bmatrix} \Delta x_p(k) \\ y(k) \end{bmatrix}
\end{aligned}$$

Using the formula $C(zI - A)^{-1}B$ to convert from state-space to transfer function

$$\begin{aligned}
\frac{Y(z)}{\Delta U(z)} &= \frac{z}{z-1}\frac{c_p b_p}{z-a_p} \\
&= \frac{c_p b_p}{z-a_p}\frac{1}{1-z^{-1}} \\
Y(z) &= \frac{c_p b_p}{z-a_p}\frac{1}{1-z^{-1}}\Delta U(z)
\end{aligned}$$

2.2 Prediction of State and Output Variables

Assume that at sampling instant k, the state variable vector $x(k)$ is available through measurement. The current and future control trajectory is denoted by $\Delta u(k)$, $\Delta u(k+1)$, ..., $\Delta u(k+N_c-1)$ where N_c is called the control horizon. The future state variables are predicted for N_p number of samples, where N_p is called the prediction horizon. We donote the future state as $x(k+1)$, $x(k+2)$, ..., $x(k+N_p)$ where $x(k+m)$ is the predicted state at $k+m$ given $x(k)$. $N_c \leq N_p$.

Based on the state-space model of (2.5), the future state variables are calculated sequentially as follows.

$$
\begin{aligned}
x(k+1) &= Ax(k) + B\Delta u(k) && (2.8)\\
x(k+2) &= Ax(k+1) + B\Delta u(k+1) \\
&= A^2x(k) + AB\Delta u(k) + B\Delta u(k+1) && (2.9)\\
x(k+3) &= Ax(k+2) + B\Delta u(k+2) \\
&= A^3x(k) + A^2B\Delta u(k) + AB\Delta u(k+1) + B\Delta u(k+2) && (2.10)\\
&\vdots \\
x(k+N_p) &= A^{N_p}x(k) + A^{N_p-1}B\Delta u(k) + A^{N_p-2}B\Delta u(k+1) + \ldots + A^{N_p-N_c}B\Delta u(k+N_c-1) && (2.11)
\end{aligned}
$$

Substituting (2.8) to (2.11) into (2.6) gives

$$\begin{aligned}
y(k+1) &= CAx(k) + CB\Delta u(k) && (2.12)\\
y(k+2) &= CA^2x(k) + CAB\Delta u(k) + CB\Delta u(k+1) && (2.13)\\
y(k+3) &= CA^3x(k) + CA^2B\Delta u(k) + CAB\Delta u(k+1) + CB\Delta u(k+2) && (2.14)\\
&\vdots \\
y(k+N_p) &= CA^{N_p}x(k) + CA^{N_p-1}B\Delta u(k) + CA^{N_p-2}B\Delta u(k+1) \\
&+ \ldots + CA^{N_p-N_c}B\Delta u(k+N_c-1) && (2.15)
\end{aligned}$$

Note that all predicted variables are formulated in terms of current state $x(k)$ and current and future control movement

$$\Delta U = \begin{bmatrix} \Delta u(k) & \Delta u(k+1) & \Delta u(k+2) & \ldots & \Delta u(k+N_c-1) \end{bmatrix}^T$$

Define

$$Y = \begin{bmatrix} y(k+1) & y(k+2) & y(k+3) & \ldots & y(k+N_p) \end{bmatrix}^T$$

The dimension of Δu and Y are N_c and N_p respectively.

We collect (2.12) to (2.15) together in a compact matrix form as

$$Y = Fx(k) + \Phi\Delta U \tag{2.16}$$

where

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix} \tag{2.17}$$

$$\Phi = \begin{bmatrix} CB & 0 & 0 & \ldots & 0 \\ CAB & CB & 0 & \ldots & 0 \\ CA^2B & CAB & CB & \ldots & 0 \\ \ldots & & & & \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \ldots & CA^{N_p-N_c}B \end{bmatrix} \tag{2.18}$$

2.3 Optimization

For a given set-point signal $r(k)$ at sample time k, the objective is then translated into a design to find the 'best' control parameter vector ΔU such that an error funciton between the set-point and the predicted output is minimized. Assuming that the signal vector that contains the set-point information is

$$R_s^T = \overbrace{\begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}}^{N_p} r(k) \tag{2.19}$$

we define the cost function J that reflects the control objective as

$$J = \frac{1}{2}\left\{(R_s - Y)^T(R_s - Y) + \Delta U^T \bar{R} \Delta U\right\} \tag{2.20}$$

where the first term is linked to the objective of minimizing the errors between the predicted output and the set-point signal while the second term reflects the consideration given to the size of ΔU when the objective function J is made to be as small as possible. $\bar{R}$ is a diagonal matrix of weights

$$\bar{R} = r_w I_{N_c \times N_c} \tag{2.21}$$

where $r_w \geq 0$ is used as a tuning parameter for the deisred closed-loop performance.

Substitute (2.16) into (2.20) gives

$$J = \frac{1}{2}\Delta U^T(\Phi^T\Phi + \bar{R})\Delta U - \Delta U^T\Phi^T(R_s - Fx(k)) + \frac{1}{2}(R_s - Fx(k))^T(R_s - Fx(k)) \tag{2.22}$$

Since the last term do not depend on ΔU, minimizing

$$J = \frac{1}{2}\Delta U^T(\Phi^T\Phi + \bar{R})\Delta U - \Delta U^T\Phi^T(R_s - Fx(k)) \tag{2.23}$$

wrt ΔU gives the same results. From the first derivative of the cost function J

$$\frac{\partial J}{\partial \Delta U} = (\Phi^T\Phi + \bar{R})\Delta U - \Phi^T(R_s - Fx(k)) \tag{2.24}$$

the necessary condition of the minimum J is obtained as

$$\frac{\partial J}{\partial \Delta U} = 0$$

from which we find the optimal solution for the control signal as

$$\Delta U = (\Phi^T\Phi + \bar{R})^{-1}\Phi^T(R_s - Fx(k)) \tag{2.25}$$

with the assumption that $(\Phi^T\Phi + \bar{R})^{-1}$ exists. Note that R_s is a signal vector that contains the set-point information expressed as

$$R_s = \overbrace{\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \end{bmatrix}^T}^{N_p} r(k) = \bar{R}_s r(k) \tag{2.26}$$

where

$$\bar{R}_s = \overbrace{\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \end{bmatrix}^T}^{N_p} \tag{2.27}$$

is a column vector of weights of 1. Substitute R_s in (2.25) gives

$$\Delta U = (\Phi^T\Phi + \bar{R})^{-1}\Phi^T(\bar{R}_s r(k) - Fx(k)) \tag{2.28}$$

2.4 Receding Horizon Control

At each sampling instance k, although the optimal parameter vector ΔU contains the controls $\Delta u(k)$, $\Delta u(k+1)$, $\Delta u(k+2)$, ..., $\Delta u(k+N_c-1)$, with the receding horizon control principle, we only implement the first sample of the sequence, i.e. $\Delta u(k)$ while ignoring the rest of the sequence.

2.5 Closed-Loop Control System

Because of the receding horizon control principle, we only take the first element of ΔU in (2.28) at time k. Thus

$$\begin{aligned}\Delta u(k) &= \overbrace{\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}}^{N_c}(\Phi^T\Phi+\bar{R})^{-1}(\Phi^T\bar{R}_s r(k) - \Phi^T F x(k)) \\ &= K_r r(k) - K_{mpc} x(k) \end{aligned} \tag{2.29}$$

where

$$K_r = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}(\Phi^T\Phi+\bar{R})^{-1}\Phi^T\bar{R}_s \tag{2.30}$$

$$K_{mpc} = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}(\Phi^T\Phi+\bar{R})^{-1}\Phi^T F \tag{2.31}$$

Note that K_r is the first element of $(\Phi^T\Phi+\bar{R})^{-1}\Phi^T\bar{R}_s$ and K_{mpc} is the first row of $(\Phi^T\Phi+\bar{R})^{-1}\Phi^T F$. Equation (2.29) is in a standard form of linear time-invariant state feedback control.

Substitute (2.29) into (2.5) gives

$$
\begin{aligned}
x(k+1) &= Ax(k) - BK_{mpc}x(k) + BK_r r(k) &(2.32)\\
&= (A - BK_{mpc})x(k) + BK_r r(k) &(2.33)
\end{aligned}
$$

Using (2.6), and taking z-transform, the closed-loop transfer function is given as

$$
\frac{Y(z)}{R(z)} = C\left[zI - (A - BK_{mpc})\right]^{-1} BK_r \tag{2.34}
$$

The closed-loop poles can be evaluated through the closed-loop characteristic equation

$$
\det[zI - (A - BK_{mpc})] = 0 \tag{2.35}
$$

We obtain the closed-loop block diagram for the predictive control system in Figure 2.2 where q^{-1} denotes the backward shift operator. The diagram shows the state feedback structure for the MPC with integral action in which the block $\frac{1}{1-q^{-1}}$ denotes the discrete-time integrator.

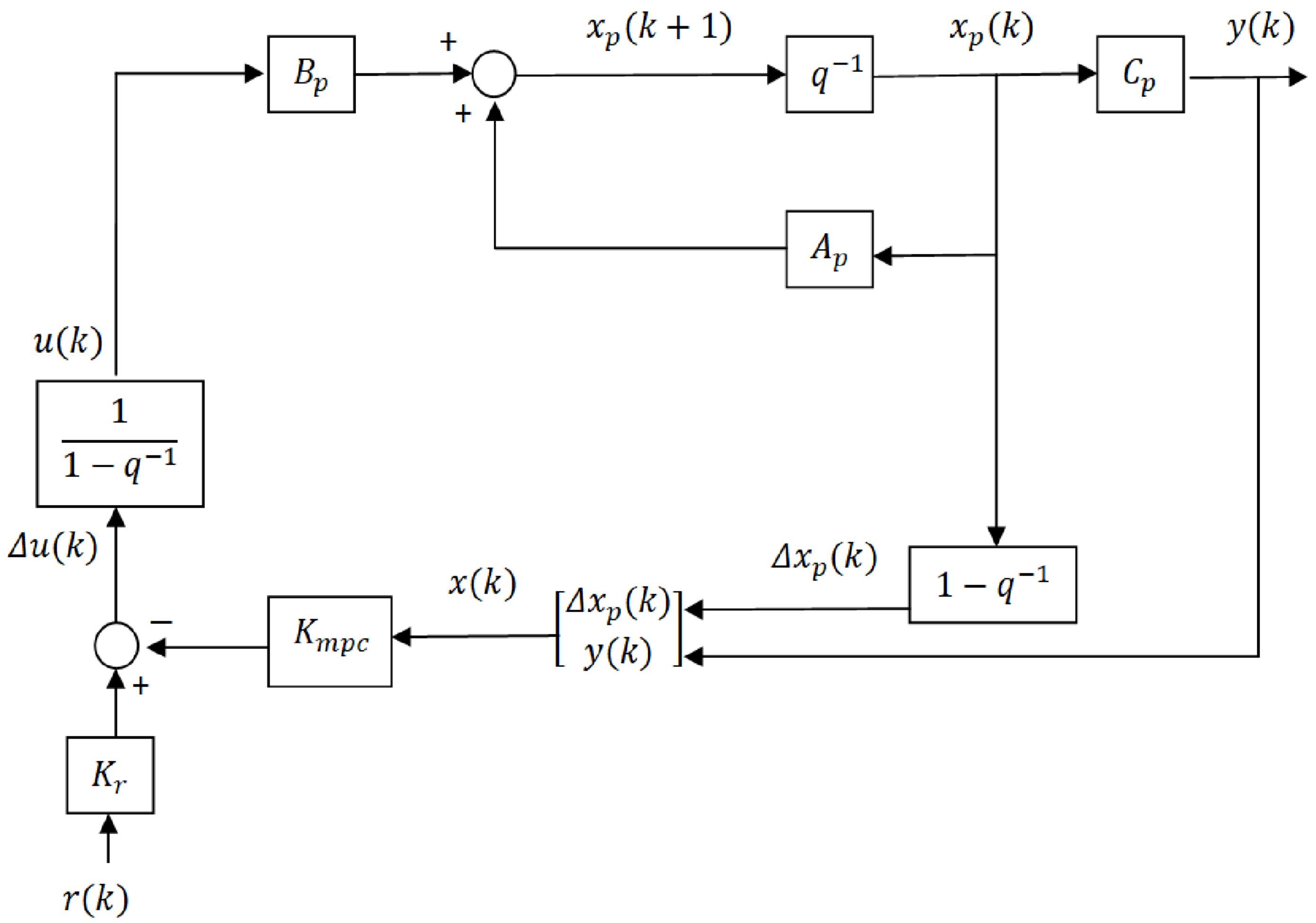

Figure 2.1: Full-State Feedback Control

Example 7

Consider the tank shown in the figure where $u(t)$, $q(t)$ and $h(t)$ are the inflow rate in m^3/s, the

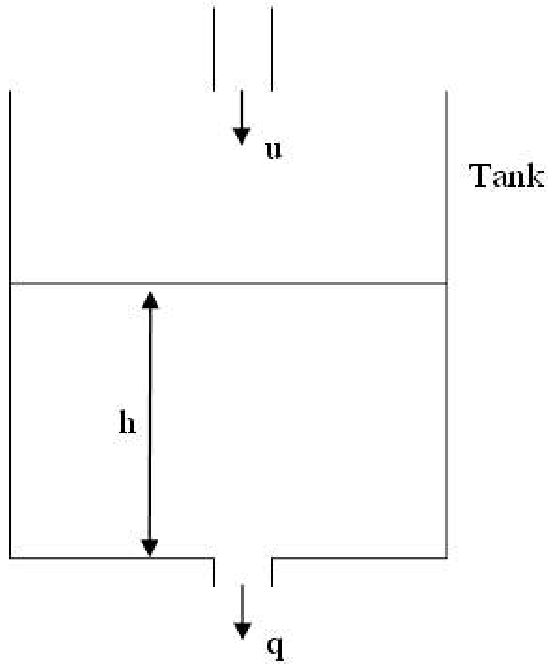

Figure 2.2: Full-State Feedback Control

outflow rate in m^3/s, and height in m. The differential equation for the height is given as

$$\begin{aligned} A\dot{h}(t) &= -q(t) + u(t) \\ &= -\frac{1}{R}h(t) + u(t) \end{aligned}$$

where the area $A = 10\ \mathrm{m}^2$ and constant $R = 0.5\ \mathrm{s/m}^2$. □

(a) Define the height $h(t)$ as the state $x_p(t)$ and output $y(t)$, obtain the discretized state-space model with sampling interval of 1 s. □

The state-space model is then

$$\begin{aligned} \dot{x}_p(t) &= -\frac{1}{AR}x_p(t) + \frac{1}{A}u(t) \\ y(t) &= x_p(t) \end{aligned}$$

Discretize by approximating the derivative using the forward difference with sampling interval h gives

$$\begin{aligned} \frac{x_p(k+1) - x_p(k)}{h} &= -\frac{1}{AR}x_p(k) + \frac{1}{A}u(k) \\ y(k) &= x_p(k) \end{aligned}$$

Substituting the values of A, R and h gives

$$\begin{aligned} x_p(k+1) &= a_p x_p(k) + b_p u(k) \\ y(k) &= c_p x_p(k) \end{aligned}$$

where $a_p = 0.8$, $b_p = 0.1$, $c_p = 1$.

(b) Find $\Delta U(k)$, $\Delta u(k)$, $u(k)$ and $y(k)$ of the MPC with integrator control system for $k = 0, 1, 2$. It is given that $r_w = 0.01$, $N_p = 3$ and $N_c = 2$, $x_p(k) = y(k) = 0$, for $k \leq 0$, $u(k) = 0$ for $k < 0$, and

$$r(k) = \begin{cases} 0 & \text{for } k < 0 \\ 1 & \text{for } k \geq 0 \end{cases}$$

□

Equations (2.5) and (2.6) give

$$\begin{aligned} A &= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \\ B &= \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} \\ C &= \begin{bmatrix} 0 & 1 \end{bmatrix} \end{aligned}$$

Equations (2.17), (2.18), (2.21) and (2.26) gives

$$
\begin{aligned}
R_s &= \bar{R}_s r(k) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T \times 1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T \\
F &= \begin{bmatrix} CA \\ CA^2 \\ CA^3 \end{bmatrix} = \begin{bmatrix} 0.8 & 1 \\ 1.44 & 1 \\ 1.952 & 1 \end{bmatrix} \\
\Phi &= \begin{bmatrix} CB & 0 \\ CAB & CB \\ CA^2B & CAB \end{bmatrix} = \begin{bmatrix} 0.1 & 0 \\ 0.18 & 0.1 \\ 0.244 & 0.18 \end{bmatrix} \\
\bar{R} &= r_w I_{N_c \times N_c} = 0.01 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\end{aligned}
$$

Equation (2.25) gives $\Delta U(k)$ and $\Delta u(k)$. Equations (2.5) and (2.6) give $x(k)$ and $y(k)$.

$$\begin{aligned}
\Delta U(0) &= \begin{bmatrix} 4.9819 & -0.5435 \end{bmatrix}^T, \quad \Delta u(0) = 4.9819 \\
u(0) &= u(-1) + \Delta u(0) = 4.9819, \quad x(1) = \begin{bmatrix} 0.4982 & 0.4982 \end{bmatrix}^T, \quad y(1) = 0.4982 \\
\Delta U(1) &= \begin{bmatrix} -0.4575 & -1.4876 \end{bmatrix}^T, \quad \Delta u(1) = -0.4575 \\
u(1) &= u(0) + \Delta u(1) = 4.5244, \quad x(2) = \begin{bmatrix} 0.3528 & 0.8510 \end{bmatrix}^T, \quad y(2) = 0.8510 \\
\Delta U(2) &= \begin{bmatrix} -1.3520 & -0.9413 \end{bmatrix}^T, \quad \Delta u(2) = -1.3520 \\
u(2) &= u(1) + \Delta u(2) = 3.1724, \quad x(3) = \begin{bmatrix} 0.1470 & 0.9980 \end{bmatrix}^T, \quad y(3) = 0.9980
\end{aligned}$$

k	0	1	2	3
$\Delta u(k)$	4.9819	−0.4575	−1.3520	
$u(k)$	4.9819	4.5244	3.1724	
$y(k)$	0	0.4982	0.8510	0.9980

(c) Find K_r and K_{mpc}. □

Equations (2.30) and (2.31) give

$$\begin{aligned} K_r &= 4.9819 \\ K_{mpc} &= \begin{bmatrix} 5.9364 & 4.9819 \end{bmatrix} \end{aligned}$$

(d) Find $\Delta u(k)$ and $u(k)$ using K_r and K_{mpc} and compare with the results in (a). $\square$

Using (2.29)

$$\begin{aligned} \Delta u(0) &= K_r r(0) - K_{mpc} x(0) = 4.9819 \\ u(0) &= u(-1) + \Delta u(0) = 4.9819 \end{aligned}$$

Using (2.33)

$$\begin{aligned} x(1) &= (A - BK_{mpc})x(0) + BK_r r(0) = \begin{bmatrix} 0.4982 \\ 0.4982 \end{bmatrix} \\ \Delta u(1) &= K_r r(1) - K_{mpc} x(1) = -0.4575 \\ u(1) &= u(0) + \Delta u(1) = 4.5244 \\ x(2) &= (A - BK_{mpc})x(1) + BK_r r(1) = \begin{bmatrix} 0.3528 \\ 0.8510 \end{bmatrix} \\ \Delta u(2) &= K_r r(2) - K_{mpc} x(2) = -1.352 \\ u(2) &= u(1) + \Delta u(2) = 3.1724 \end{aligned}$$

which is the same as the results in part (a).

(e) Find the closed-loop transfer function $\frac{Y(z)}{R(z)}$. □

Use (2.34)

$$
\begin{aligned}
\frac{Y(z)}{R(z)} &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} z-0.2064 & 0.4982 \\ -0.2064 & z-0.5018 \end{bmatrix}^{-1} \begin{bmatrix} 0.4982 \\ 0.4982 \end{bmatrix} \\
&= \frac{0.4982z}{z^2-0.7082z+0.2064}
\end{aligned}
$$

(f) Find $y(k)$ from the closed-loop transfer function and compare with the results in (a). □

$$
\begin{aligned}
\frac{Y(z)}{R(z)} &= \frac{0.4982z^{-1}}{1-0.7082z^{-1}+0.2064z^{-2}} \\
y(k) &= 0.7082y(k-1)-0.2064y(k-2)+0.4982r(k-1) \\
y(0) &= 0 \text{ (given in the question)} \\
y(1) &= 0.4982 \\
y(2) &= 0.7082 \times 0.4982 + 0.4982 = 0.851 \\
y(3) &= 0.7082 \times 0.851 - 0.2064 \times 0.4982 + 0.4982 = 0.998
\end{aligned}
$$

which is the same as the $y(k)$ in part (a).

2.5.1 Tuning

The prediction horizon, control horizon and weight N_p, N_c and r_w respectively are used as tuning parameters. As a starting point we can choose $N_c = 1$ and N_p as the number of samples in the open-loop rise-time which is typically between 4 and 15. The closed-loop system is simulated to determine whether the control signal $u(k)$ is too aggressive or sluggish by changing r_w. For the case of $r_w = 0$, the cost function (2.20) is interpreted as the situation where we would not want to pay any attention to how large the Δu might be and our goal would be solely to make the error $(R_s - Y)^T(R_s - Y)$ as small as possible.

2.6 State Estimation

In the design of model predictive controllers, we assumed that the information $x(k)$ is available at the time k. This assumes that all the state variables are measurable. In some applications, not all state variables are measured (or available). Some of them may be impossible to measure. In a deterministic system such as (2.5) and (2.6) i.e. a no noise system, an observer can be used to obtain the states.

2.6.1 Kalman Filter Approach

One approach is to estimate the state variable $\hat{x}(k+1)$ from the steady-state Kalman filter (1.55). For the sake of computing the observer gain $K_{ob} = K(k)$ in (1.55) for $k = \infty$, we pretend that there are zero-mean independent Gaussian process noise $w(k)$ and meaurement noise $v(k)$ as follows.

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

where the covariance matrix of $w(k)$ and $v(k)$ are given by R_1 and R_2 respectively. Note that $w(k)$ and $v(k)$ are needed only for computing the observer gain; it is not needed for the rest of the control system design which should be (2.5) and (2.6). Using (1.55), the observer can be written as

$$\hat{x}(k+1) = A\hat{x}(k) + B\Delta u(k) + K_{ob}(y(k) - C\hat{x}(k)) \tag{2.36}$$

Note that $u(k)$ in (1.55) is replaced by $\Delta u(k)$ because (2.5) is in terms of $\Delta u(k)$. For simplicity, we use (1.55) which give the predicted estimate $\hat{x}(k+1|k)$ for the observer. Since we will not be using the filter estimate $\hat{x}(k|k)$ for the observer, there is no need to make a distinction between $\hat{x}(k|k)$ and $\hat{x}(k|k-1)$, we can write $\hat{x}(k+1|k)$ and $\hat{x}(k|k-1)$ simply as $\hat{x}(k+1)$ and $\hat{x}(k)$ respectively.

2.6.2 Pole-Placement Approach

Instead of calculating observer gain K_{ob} from the Kalman filter algorithm, another approach is to choose the observer gain K_{ob} to give desired closed-loop poles. We examine the closed-loop error equation. Substituting (2.6) into (2.36) and substracting (2.5) gives

$$\begin{aligned}
\hat{x}(k+1) - x(k+1) &= A\hat{x}(k) - Ax(k) + K_{ob}(Cx(k) - C\hat{x}(k)) \\
\tilde{x}(k+1) &= A\tilde{x}(k) - K_{ob}C\tilde{x}(k) \\
&= (A - K_{ob}C)\tilde{x}(k)
\end{aligned} \tag{2.37}$$

where state $\tilde{x}(k) = x(k) - \hat{x}(k)$. Now, with given initial error $\tilde{x}(0)$, we have

$$\tilde{x}(k) = (A - K_{ob}C)^k \tilde{x}(0)$$

It is apparent that the observer gain K_{ob} can be used to manipulate the convergence rate of the error. A commonly used approach is to place the closed-loop eigenvalues of the error system matrix $A - K_{ob}C$ at a desired location of the complex plane. Taking the z-transform of (2.37) gives

$$\begin{aligned} z\tilde{X}(z) - z\tilde{x}(0) &= (A - K_{ob}C)\tilde{X}(z) \\ [zI - (A - K_{ob}C)]\,\tilde{X}(z) &= z\tilde{x}(0) \\ \tilde{X}(z) &= [zI - (A - K_{ob}C)]^{-1} z\tilde{x}(0) \end{aligned}$$

and the characteristic equation

$$\det(zI - (A - K_{ob}C)) = 0 \tag{2.38}$$

Choose K_{ob} to give desired poles.

2.7 State Estimate Predictive Control

Essentially, the state variable $x(k)$ is estimated via an observer in (2.36) and in all the MPC equations, $x(k)$ is replaced by $\hat{x}(k)$.

Equation (2.29) is now given as

$$\Delta u(k) = K_r r(k) - K_{mpc}\hat{x}(k) \tag{2.39}$$

The block diagram of the control system is given in Figure 2.3. Substitute (2.39) into the plant in (2.5) gives

$$x(k+1) = Ax(k) + BK_r r(k) - BK_{mpc}\hat{x}(k) \tag{2.40}$$

Replacing $\hat{x}(k)$ by $x(k) - \tilde{x}(k)$ gives

$$x(k+1) = (A - BK_{mpc})x(k) + BK_{mpc}\tilde{x}(k) + BK_r r(k) \tag{2.41}$$

Note that the closed-loop observer error equation is (2.37). Combination of (2.37) and (2.41) gives

$$\begin{aligned} \begin{bmatrix} \tilde{x}(k+1) \\ x(k+1) \end{bmatrix} &= \begin{bmatrix} A - K_{ob}C & o_{n\times n} \\ BK_{mpc} & A - BK_{mpc} \end{bmatrix} \begin{bmatrix} \tilde{x}(k) \\ x(k) \end{bmatrix} + \begin{bmatrix} o_{n\times m} \\ BK_r \end{bmatrix} r(k) \\ y(k) &= \begin{bmatrix} 0_{1\times n} & C \end{bmatrix} \begin{bmatrix} \tilde{x}(k) \\ x(k) \end{bmatrix} \end{aligned} \tag{2.42}$$

where $o_{n\times m}$ is a $n \times m$ zero matrix. The closed-loop transfer function is given by

$$\frac{Y(z)}{R(z)} = \begin{bmatrix} 0_{1\times n} & C \end{bmatrix} \left(zI - \begin{bmatrix} A - K_{ob}C & o_{n\times n} \\ BK_{mpc} & A - BK_{mpc} \end{bmatrix} \right)^{-1} \begin{bmatrix} o_{n\times m} \\ BK_r \end{bmatrix}$$

and the characteristic equation

$$\det\left(zI - \begin{bmatrix} A - K_{ob}C & o_{n\times n} \\ BK_{mpc} & A - BK_{mpc} \end{bmatrix}\right) = 0 \tag{2.43}$$

which is equivalent to

$$\det(zI - (A - K_{ob}C))\det(zI - (A - BK_{mpc})) = 0$$

because the system matrix in (2.42) has a lower block triangular structure. This effectively means that the closed-loop model predictive control system with state estimate has two independent characteristic equations

$$\det(zI - (A - K_{ob}C)) = 0 \tag{2.44}$$

$$\det(zI - (A - BK_{mpc})) = 0 \tag{2.45}$$

Since the closed-loop eigenvalues are the solutions of the characteristic equations, (2.44) and (2.45) indicate that the set of eigenvalues of the combined closed-loop system consists of predictive control-loop eigenvalues (2.35) and observer-loop eigenvalues (2.38). This means that the design of the predictive control law and the observer can be carried out independently (or separately), yet when they are put together in this way, the eigenvalues remain unchanged.

Keep the estimator poles faster than the control poles in order that the total system response is dominated by the control poles. Typically, we select well-damped estimator poles that are 2 to 6 times faster than the control poles in order to provide a response dominated by the control poles.

In the Kalman filter approach, we choose the covariance matrices R_1 and R_2 to calculate the observer gain K_{ob}. The closed-loop system obtained is analyzed wrt the location of the eigenvalues of $A - K_{ob}C$, the transient response of the observer. The elements of R_1 and R_2 are modified by trial-and-error until a desired result is obtained. The advantage of using Kalman filter is that the tuning results can be interpreted from the variances of the process and measurement noises as well as from the closed-loop poles.

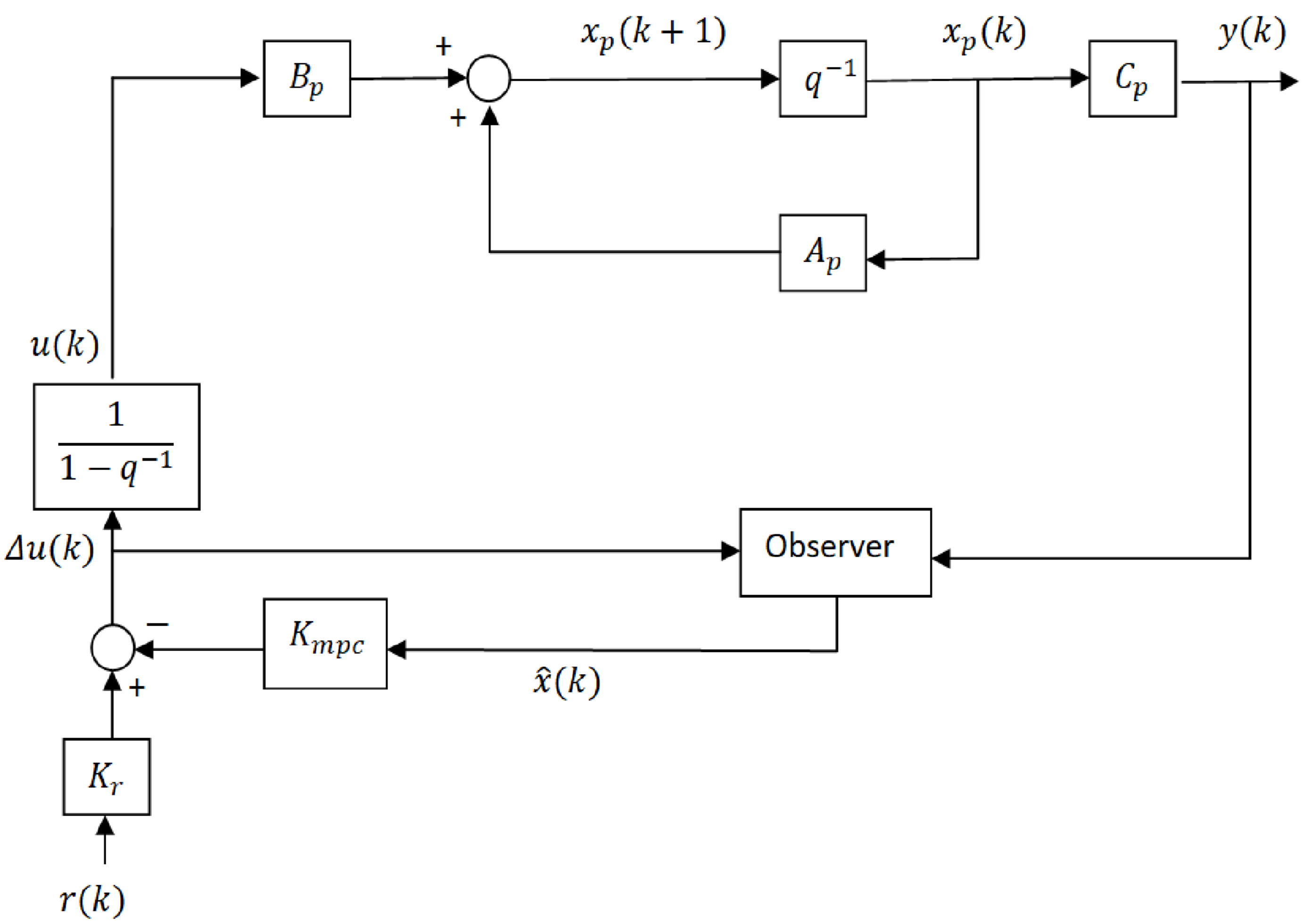

Figure 2.3: State Estimate Predictive Control

Chapter 3

Model Predictive Control with Constraints

MPC is the only generic control technology which can deal routinely with equipment and safety constraints. Operation at or near such constraints is necessary for the most profitable or most efficient operation in many cases.

As the optimal solutions will be obtained using quadratic programming, the constraints need to be decomposed into two parts to reflect the lower limit, and the upper limit with opposite sign.

3.1 Constraints on the Control Variable Incremental Variation

$$\Delta U^{min} \leq \Delta U \leq \Delta U^{max}$$

will be expressed by two inequalities

$$-\Delta U \leq -\Delta U^{min} \quad (3.1)$$

$$\Delta U \leq \Delta U^{max} \quad (3.2)$$

In a matrix form, this becomes

$$\begin{bmatrix} -I \\ I \end{bmatrix} \Delta U \leq \begin{bmatrix} -\Delta U^{min} \\ \Delta U^{max} \end{bmatrix} \quad (3.3)$$

3.2 Constraints on the Amplitude of the Control Variable

$$U^{min} \leq U \leq U^{max}$$

will be expressed by two inequalities

$$
\begin{aligned}
-U &\leq -U^{min} && (3.4)\\
U &\leq U^{max} && (3.5)
\end{aligned}
$$

Express U in terms of ΔU gives

$$
\overbrace{\begin{bmatrix} u(k) \\ u(k+1) \\ u(k+2) \\ \vdots \\ u(k+N_c-1) \end{bmatrix}}^{U} = \overbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}}^{C_1} u(k-1) + \overbrace{\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \vdots & & & & \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix}}^{C_2} \overbrace{\begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix}}^{\Delta U} \quad (3.6)
$$

Substitute (3.6) into (3.4) and (3.5) gives

$$
\begin{aligned}
-(C_1 u(k-1) + C_2 \Delta U) &\leq -U^{min} && (3.7)\\
C_1 u(k-1) + C_2 \Delta U &\leq U^{max} && (3.8)
\end{aligned}
$$

In matrix form

$$
\begin{bmatrix} -C_2 \\ C_2 \end{bmatrix} \Delta U \leq \begin{bmatrix} -U^{min} + C_1 u(k-1) \\ U^{max} - C_1 u(k-1) \end{bmatrix} \quad (3.9)
$$

3.3 Output Constraints

$$Y^{min} \leq Y \leq Y^{max} \tag{3.10}$$

Substitute (2.16) into (3.10) gives

$$Y^{min} \leq Fx(k) + \Phi\Delta U \leq Y^{max}$$

Expressed by two inequalities

$$\begin{aligned} -(Fx(k) + \Phi\Delta U) &\leq -Y^{min} \\ Fx(k) + \Phi\Delta U &\leq Y^{max} \end{aligned}$$

In matrix form

$$\begin{bmatrix} -\Phi \\ \Phi \end{bmatrix} \Delta U \leq \begin{bmatrix} -Y^{min} + Fx(k) \\ Y^{max} - Fx(k) \end{bmatrix} \tag{3.11}$$

3.4 Control with Constraints

The MPC in the presence of constraints is to find ΔU that minimizes (2.23)

$$J = \frac{1}{2}\Delta U^T(\Phi^T\Phi + \bar{R})\Delta U - \Delta U^T\Phi^T(R_s - Fx(k))$$

subject to the inequality constraints

$$\begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} \Delta U \leq \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix} \tag{3.12}$$

where (3.3), (3.9) and (3.11) gives

$$\begin{aligned}
M_1 &= \begin{bmatrix} -I \\ I \end{bmatrix}; \quad N_1 = \begin{bmatrix} -\Delta U^{min} \\ \Delta U^{max} \end{bmatrix}; \\
M_2 &= \begin{bmatrix} -C_2 \\ C_2 \end{bmatrix}; \quad N_2 = \begin{bmatrix} -U^{min} + C_1 u(k-1) \\ U^{max} - C_1 u(k-1) \end{bmatrix}; \\
M_3 &= \begin{bmatrix} -\Phi \\ \Phi \end{bmatrix}; \quad N_3 = \begin{bmatrix} -Y^{min} + Fx(k) \\ Y^{max} - Fx(k) \end{bmatrix}
\end{aligned}$$

Compare (2.23) and (3.12) with the standard quadratic programming problem

$$J = \frac{1}{2}\Delta U^T H \Delta U + \Delta U^T f \tag{3.13}$$

$$M\Delta U \leq \gamma \tag{3.14}$$

gives

$$
\begin{aligned}
H &= \Phi^T\Phi + \bar{R} && (3.15)\\
f &= -\Phi^T(R_s - Fx(k)) && (3.16)\\
M &= \begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} && (3.17)\\
\gamma &= \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix} && (3.18)
\end{aligned}
$$

There are many quadratic programming solver that can be used to solve (3.13) with constraints (3.14).

Example 8

Consider the plant

$$\begin{aligned} x_p(k+1) &= A_p x_p(k) + B_p u_p(k) \\ y(k) &= C_p x_p(k) \end{aligned}$$

where $A_p = 1$, $B_p = 1$, $C_p = 1$ and the augmented plant

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ y(k) &= Cx(k) \end{aligned} \tag{3.19}$$

which according to (2.7)

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} \Delta x_p(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} x_p(k) - x_p(k-1) \\ C_p x_p(k) \end{bmatrix}$$

and (2.5)

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad C = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

where $x_p(k) = y(k) = 0$ for $k \leq 0$ and $u(k) = 0$ for $k < 0$.

Consider the MPC with $r_w = 5$, $N_c = 1$, $N_p = 2$ and

$$r(k) = \begin{cases} 0 & \text{for } k < 0 \\ 1 & \text{for } k \geq 0 \end{cases}$$

(a) Formulate a quadratic programming problem with constraint $\begin{bmatrix} y(k+1) \\ y(k+2) \end{bmatrix} \leq Y^{max}$. $\square$

Equations (2.21), (2.26), (2.17) and (2.18) give

$$\begin{aligned} \bar{R} &= r_w I_{N_c} = 5 \\ R_s &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ F &= \begin{bmatrix} CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} \\ \Phi &= \begin{bmatrix} CB \\ CAB \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \end{aligned}$$

Equations (3.15), (3.16), (3.17) and (3.18) give

$$\begin{aligned}
H &= 10 \\
f &= \begin{bmatrix} 5x_1(k) + 3x_2(k) - 3 \end{bmatrix} \\
M &= \Phi = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\
\gamma &= Y^{max} - Fx(k) = \begin{bmatrix} Y^{max} - x_1(k) - x_2(k) \\ Y^{max} - 2x_1(k) - x_2(k) \end{bmatrix}
\end{aligned}$$

The quadratic programming problem (3.13) with constraint (3.14) can be formulated as minimizing

$$\begin{aligned}
J &= \frac{1}{2}\Delta U^T H \Delta U + \Delta U^T f \\
M\Delta U &\leq \gamma
\end{aligned}$$

Substituting for H, f, M, γ and ΔU gives

$$J = 5\Delta u(k)^2 + (5x_1(k) + 3x_2(k) - 3)\Delta u(k)$$

with constraints

$$\Delta u(k) \leq Y^{max} - x_1(k) - x_2(k) \tag{3.20}$$

$$\Delta u(k) \leq \frac{1}{2}\left(Y^{max} - 2x_1(k) - x_2(k)\right) \tag{3.21}$$

(b) Given $u(0) = y(0) = \Delta u(0) = 0$, $x(0) = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ and $Y^{max} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. Find $x(k)$, $\Delta u(k)$ and $y(k)$ for $k = 0, 1, \ldots 9$. $\square$

$$\begin{aligned} \frac{dJ}{d\Delta u(k)} &= 10\Delta u(k) + (5x_1(k) + 3x_2(k) - 3) = 0 \\ \Delta u(k) &= \frac{3 - 5x_1(k) - 3x_2(k)}{10} \end{aligned} \tag{3.22}$$

<u>$k = 0$</u>

Equation (3.22) gives $\Delta u(0) = 0.3$. Constaints (3.20) gives $\Delta u(0) \leq 1$ and (3.21) $\Delta u(0) \leq 0.5$. The minimum J is obtained from the minimum of $\Delta u(0) = 0.3, 1, 0.5$. Hence $\Delta u(0) = 0.3$. See Figure 3.1.

<u>$k = 1$</u>

$$x(1) = Ax(0) + B\Delta u(0) = \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}$$

Equation (3.22) gives $\Delta u(1) = 0.06$. Constraints (3.20) give $\Delta u(1) \leq 0.4$ and (3.21) $\Delta u(0) \leq 0.05$. The minimum J is obtained from the minimum of $\Delta u(1) = 0.06, 0.4, 0.05$. Hence $\Delta u(1) = 0.05$.

Table 3.1

k	$x_1(k)$ (3.19)	$x_2(k) = y(k)$ (3.19)	Constraint (3.20)	(3.21)	(3.22)	$\Delta u(k)$
0	0	0	≤ 1	≤ 0.5	0.3	0.3
1	0.3	0.3	≤ 0.4	≤ 0.05	0.06	0.05
2	0.35	0.65	≤ 0	≤ -0.175	-0.07	-0.175
3	0.175	0.825	≤ 0	≤ -0.0875	-0.035	-0.0875
4	0.0875	0.9125	≤ 0	≤ -0.0438	-0.0175	-0.0438
5	0.0437	0.9563	≤ 0	≤ -0.0219	-0.0087	-0.0219
6	0.0219	0.9781	≤ 0	≤ -0.0109	-0.0044	-0.0109
7	0.0109	0.9891	≤ 0	≤ -0.0055	-0.0022	-0.0055
8	0.0055	0.9945	≤ 0	≤ -0.0027	-0.0011	-0.0027
9	0.0027	0.9973	≤ 0	≤ -0.0014	-0.0005	-0.0014

See Figure 3.2.

The rest of the results are given in the Table 3.1 and Figure 3.3.

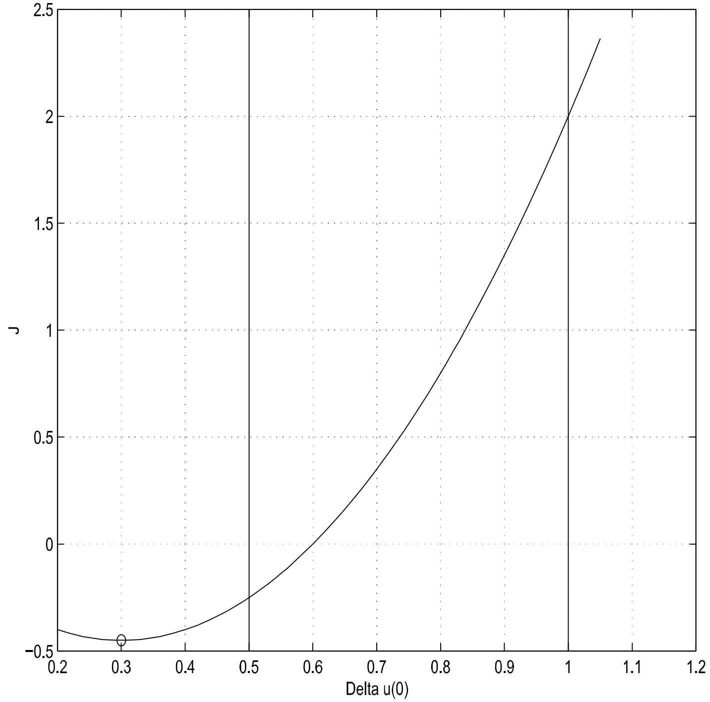

Figure 3.1

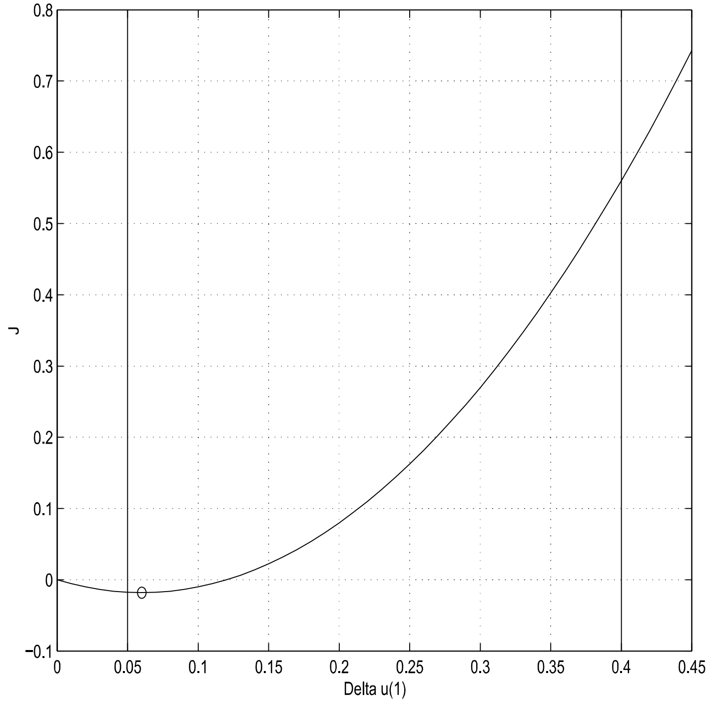
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
−0.1
J
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
Delta u(1)

Figure 3.2

(c) Find $y(k)$ if there are no constraints. $\square$

Equations (2.30) and (2.31) give

$$\begin{aligned} K_r &= 0.3 \\ K_{mpc} &= \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \end{aligned}$$

Equation (2.34) gives

$$\begin{aligned} Y(z) &= \frac{1}{(z-0.7)(z-0.5)+0.15} \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} z-0.7 & -0.3 \\ 0.5 & z-0.5 \end{bmatrix} \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix} R(z) \\ &= \frac{0.3z}{z^2-1.2z+0.5} R(z) \\ &= \frac{0.3z^{-1}}{1-1.2z^{-1}+0.5z^{-2}} R(z) \\ y(k) &= 1.2y(k-1) - 0.5y(k-2) + 0.3r(k-1) \end{aligned}$$

k	$y(k)$	
0	$0 - 0 + 0 =$	0
1	$0 - 0 + 0.3 =$	0.3
2	$(1.2)(0.3) - 0 + 0.3 =$	0.66
3	$(1.2)(0.66) - (0.5)(0.3) + 0.3 =$	0.9420
4	$(1.2)(0.942) - (0.5)(0.66) + 0.3 =$	1.1004
5	$\vdots$	1.1495
6	$\vdots$	1.1292
7		1.0803
8		1.0317
9		0.9979

Note that in Part (c), $y(k)$ overshot 1 unlike Part (b). See Figure 3.3.

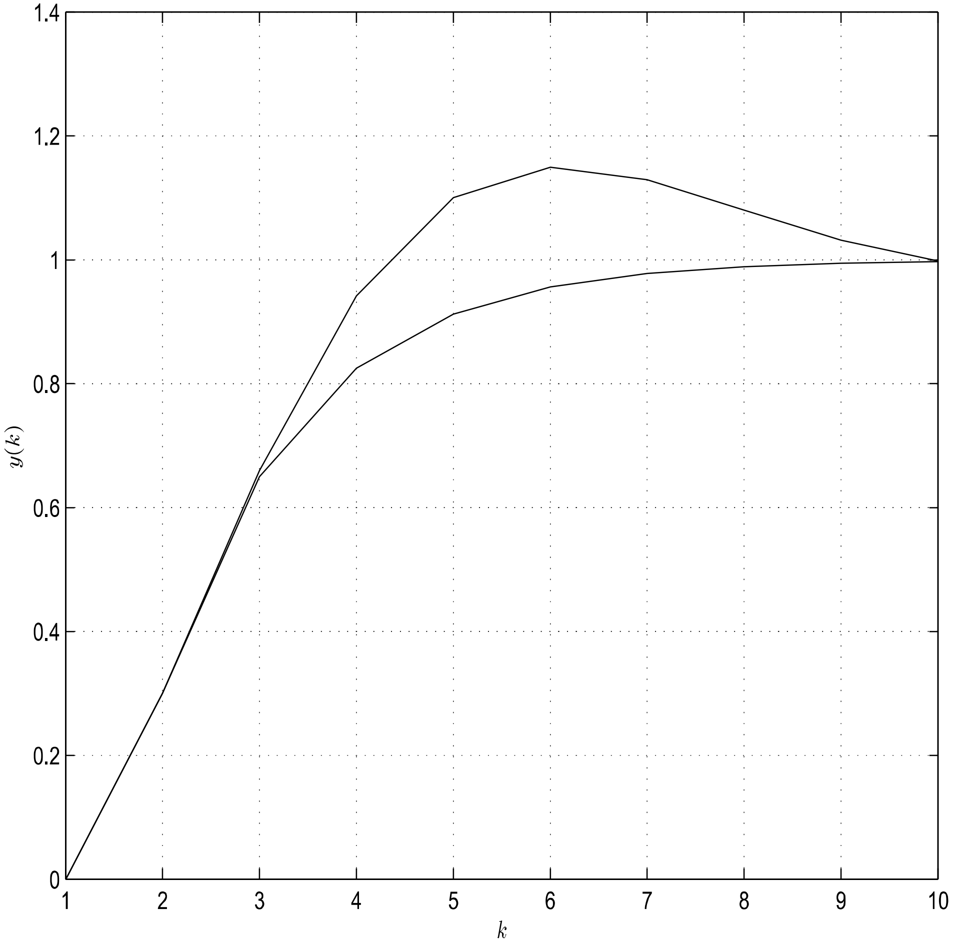

Figure 3.3

Example 9

A driverless car of mass 1000 kg moving at 10 m/s detected a boy running acrossing the road at the traffic light 15 m ahead although the signal was green for vehicle and red for pedestrian. A MPC with $r_w = 1$, $N_c = 2$, N_p =3 was used to compute the braking signal given that the maximun braking force is 5000 N. □

(a) Formulate a quadratic programming problem to compute the force.

Consider the Newton Law

$$\begin{aligned} \text{force} &= \text{mass} \times \ddot{y} \\ \frac{\text{force}}{\text{mass}} &= \ddot{y} \\ \Delta u &= \ddot{y} \end{aligned}$$

where y is distance and $\Delta u = \frac{\text{force}}{\text{mass}}$. We will compute Δu. The braking force can be obtained from force $=$ mass $\times$ Δu.

Note: There are already 2 integrators in the process. Therefore we will not introduce an integrator into the controller.

We introduce y as the state x_1 and $\dot{y}$ as the state x_2. The state-space representation is then

$$\begin{aligned}\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} &= \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}\Delta u\\ y &= \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}\end{aligned}$$

Discretize by approximating the derivative using the forward difference with sampling interval h gives

$$\begin{aligned}\begin{bmatrix}\frac{x_1(k+1)-x_1(k)}{h}\\ \frac{x_2(k+1)-x_2(k)}{h}\end{bmatrix} &= \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}\Delta u(k)\\ y(k) &= \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}\end{aligned}$$

Let $h = 1$ and rearranging gives

$$\begin{aligned}x(k+1) &= Ax(k) + B\Delta u\\ y(k) &= Cx(k)\end{aligned}$$

where

$$x(k) = \begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}, \qquad A = \begin{bmatrix}1 & 1\\ 0 & 1\end{bmatrix}, \qquad B = \begin{bmatrix}0\\ 1\end{bmatrix}, \qquad C = \begin{bmatrix}1 & 0\end{bmatrix}$$

Formulate a quadratic programming problem with constraints

$$\begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \end{bmatrix} \leq Y^{max} = \begin{bmatrix} 15 \\ 15 \\ 15 \end{bmatrix}$$

$$\begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} \geq \Delta U^{min} = \begin{bmatrix} -5 \\ -5 \end{bmatrix}$$

Equations (2.21), (2.26), (2.17) and (2.18) give

$$
\begin{aligned}
\bar{R} &= r_w I_{N_c \times N_c} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
R_s &= \overbrace{\begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \end{bmatrix}^T}^{N_p} r(k) = \begin{bmatrix} 15 \\ 15 \\ 15 \end{bmatrix} \\
F &= \begin{bmatrix} CA \\ CA^2 \\ CA^3 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \\
\Phi &= \begin{bmatrix} CB & 0 \\ CAB & CB \\ CA^2B & CAB \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 2 & 1 \end{bmatrix}
\end{aligned}
$$

Equations (3.15), (3.16), (3.17) and (3.18) give

$$
\begin{aligned}
H &= \begin{bmatrix} 6 & 2 \\ 2 & 2 \end{bmatrix} \\
f &= \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} -45 + 3x_1(k) + 8x_2(k) \\ -15 + x_1(k) + 3x_2(k) \end{bmatrix} \\
M &= \begin{bmatrix} -I \\ \Phi \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 1 & 0 \\ 2 & 1 \end{bmatrix} \\
\gamma &= \begin{bmatrix} -\Delta U^{min} \\ Y^{max} - Fx(k) \end{bmatrix} = \begin{bmatrix} 5 \\ 5 \\ 15 - x_1(k) - x_2(k) \\ 15 - x_1(k) - 2x_2(k) \\ 15 - x_1(k) - 3x_2(k) \end{bmatrix}
\end{aligned}
$$

The quadratic programming problem (3.13) with constraint (3.14) can be formulated as minimizing

$$\begin{aligned} J &= \frac{1}{2}\Delta U^T H \Delta U + \Delta U^T f \\ M\Delta U &\leq \gamma \end{aligned}$$

Substituting for H, f, M, γ and ΔU gives

$$J = 3\Delta u(k)^2 + [2\Delta u(k+1) + f_1]\Delta u(k) + [\Delta u(k+1) + f_2]\Delta u(k+1) \tag{3.23}$$

where

$$\text{IE1:} \quad \Delta u(k) \geq -5 \tag{3.24}$$

$$\text{IE2:} \quad \Delta u(k+1) \geq -5 \tag{3.25}$$

$$\text{IE3:} \quad 0 \leq 15 - x_1(k) - x_2(k) \tag{3.26}$$

$$\text{IE4:} \quad \Delta u(k) \leq 15 - x_1(k) - 2x_2(k) \tag{3.27}$$

$$\text{IE5:} \quad 2\Delta u(k) + \Delta u(k+1) \leq 15 - x_1(k) - 3x_2(k) \tag{3.28}$$

(b) Find $\Delta u(k)$ for $k = 0$, 1, 2 graphically. □

From (3.23)

$$\Delta u(k) = -\frac{2\Delta u(k+1)+f_1}{6} \pm \frac{\sqrt{[2\Delta u(k+1)+f_1]^2 - 12\{[\Delta u(k+1)+f_2]\Delta u(k+1) - J\}}}{6} \qquad (3.29)$$

$\underline{k=0}$

$$\begin{aligned} x(0) &= \begin{bmatrix} 0 \\ 10 \end{bmatrix} \\ f_1 &= 3x_1(0) + 8x_2(0) - 45 = 3(0) + 8(10) - 45 = 35 \\ f_2 &= x_1(0) + 3x_2(0) - 15 = 0 + 3(10) - 15 = 15 \end{aligned}$$

The constraints (3.24), (3.25), (3.27) $\Delta u(0) \leq -5$ and (3.28) $\Delta u(0) \leq -0.5\Delta u(1) - 7.5$ are superimposed on the $J = -100$ and -102 contours (3.29) in Figure 3.4 which also shows that the minimum $J = -100$ is at $\Delta u(0) = -5$.

$\underline{k=1}$

$$
\begin{aligned}
x(1) &= Ax(0) + B\Delta u(0) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 10 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (-5) = \begin{bmatrix} 10 \\ 5 \end{bmatrix} \\
f_1 &= 3x_1(1) + 8x_2(1) - 45 = 3(10) + 8(5) - 45 = 25 \\
f_2 &= x_1(1) + 3x_2(1) - 15 = 10 + 3(5) - 15 = 10
\end{aligned}
$$

The constraints (3.24), (3.25), (3.27) $\Delta u(1) \leq -5$, (3.27) $\Delta u(1) \leq -0.5\Delta u(2) - 5$ is superimposed on the $J = -50$ and -52 contours (3.29) in Figure 3.5 which also shows that the minimum $J = -50$ is at $\Delta u(1) = -5$. The constraints (3.24) and (3.27) are the tangents to the $J = -50$ contour.

$\underline{k=2}$

$$
\begin{aligned}
x(2) &= Ax(1) + B\Delta u(1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 10 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (-5) = \begin{bmatrix} 15 \\ 0 \end{bmatrix} \\
f_1 &= 3x_1(1) + 8x_2(1) - 45 = 3(15) + 8(0) - 45 = 0 \\
f_2 &= x_1(1) + 3x_2(1) - 15 = 15 + 3(0) - 15 = 0
\end{aligned}
$$

The constraints (3.24), (3.25), (3.27) $\Delta u(2) \leq 0$, (3.27) $\Delta u(2) \leq -0.5\Delta u(3)$ is superimposed on the $J = 0$ and 1 contours (3.29) in Figure 3.6 which also shows that the minimum $J = 0$ is at $\Delta u(2) = 0$.

$\underline{k = 3}$

$$x(3) = Ax(2) + B\Delta u(2) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 15 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (0) = \begin{bmatrix} 15 \\ 0 \end{bmatrix}$$

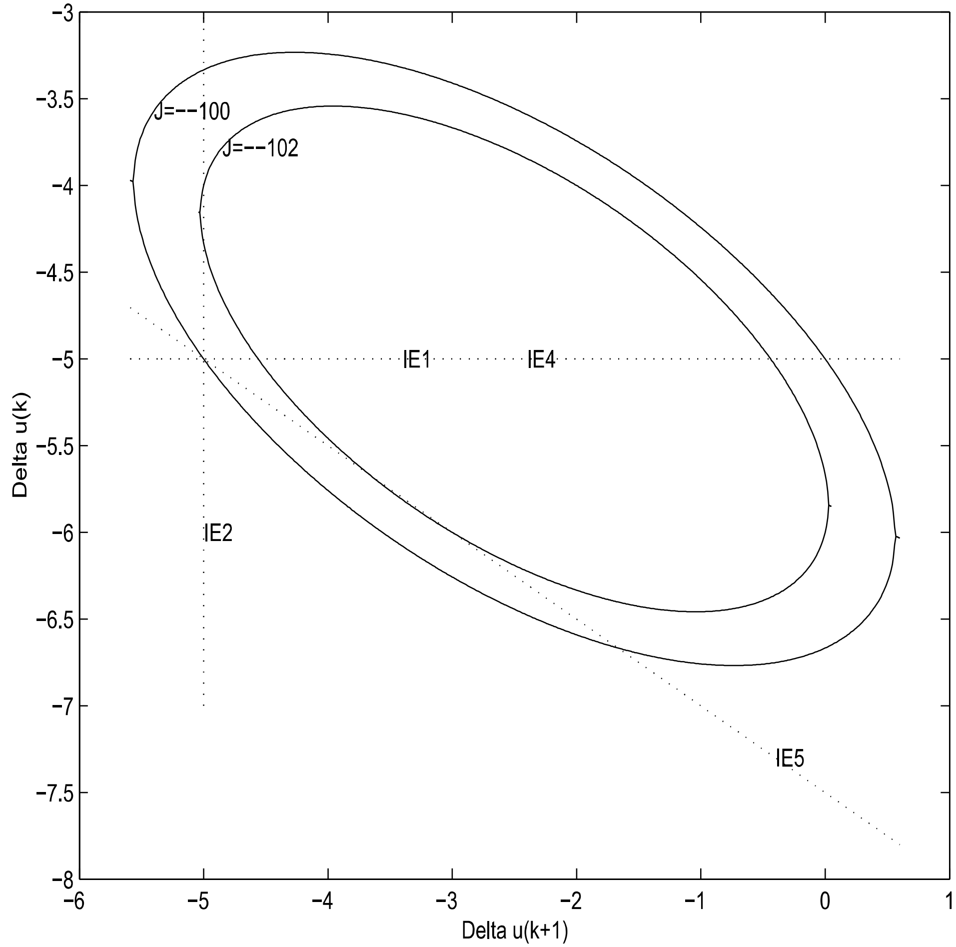
J=--100
J=--102
IE1
IE4
IE2
IE5
Delta u(k)
Delta u(k+1)
-3
-3.5
-4
-4.5
-5
-5.5
-6
-6.5
-7
-7.5
-8
-6
-5
-4
-3
-2
-1
0
1

Figure 3.4

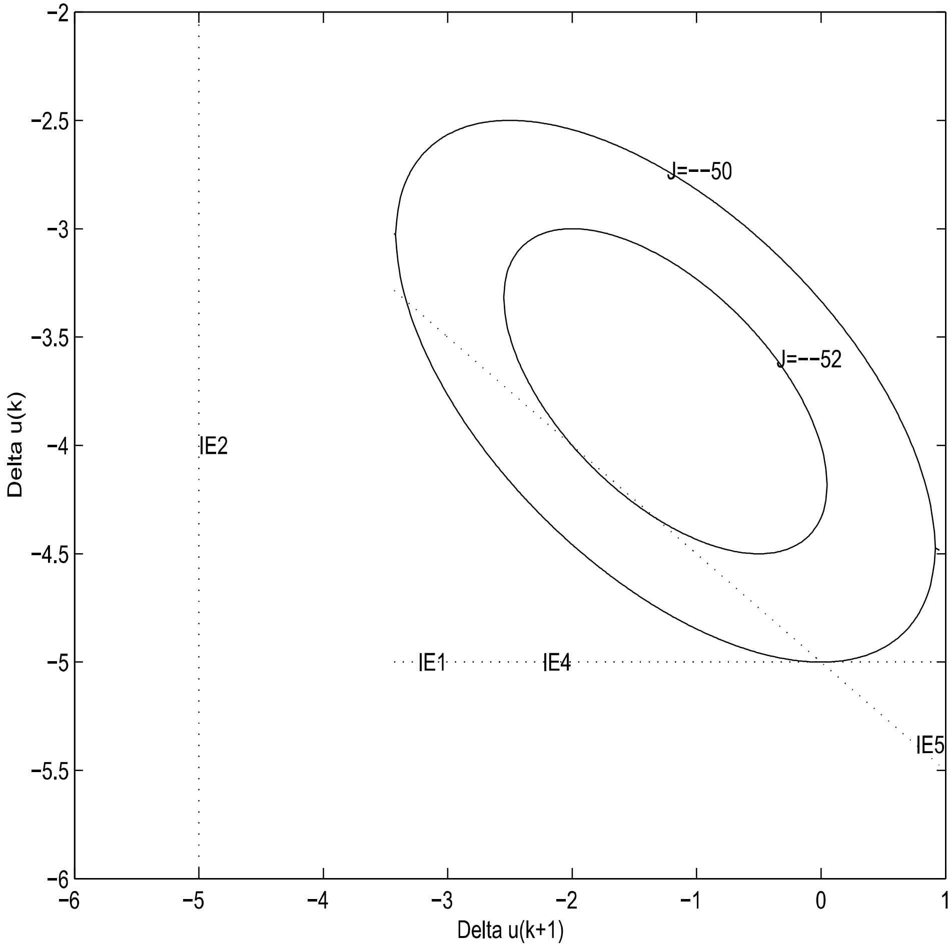
J=--50
J=--52
IE2
IE1
IE4
IE5
Delta u(k)
Delta u(k+1)
-2
-2.5
-3
-3.5
-4
-4.5
-5
-5.5
-6
-5
-4
-3
-2
-1
0
1

Figure 3.5

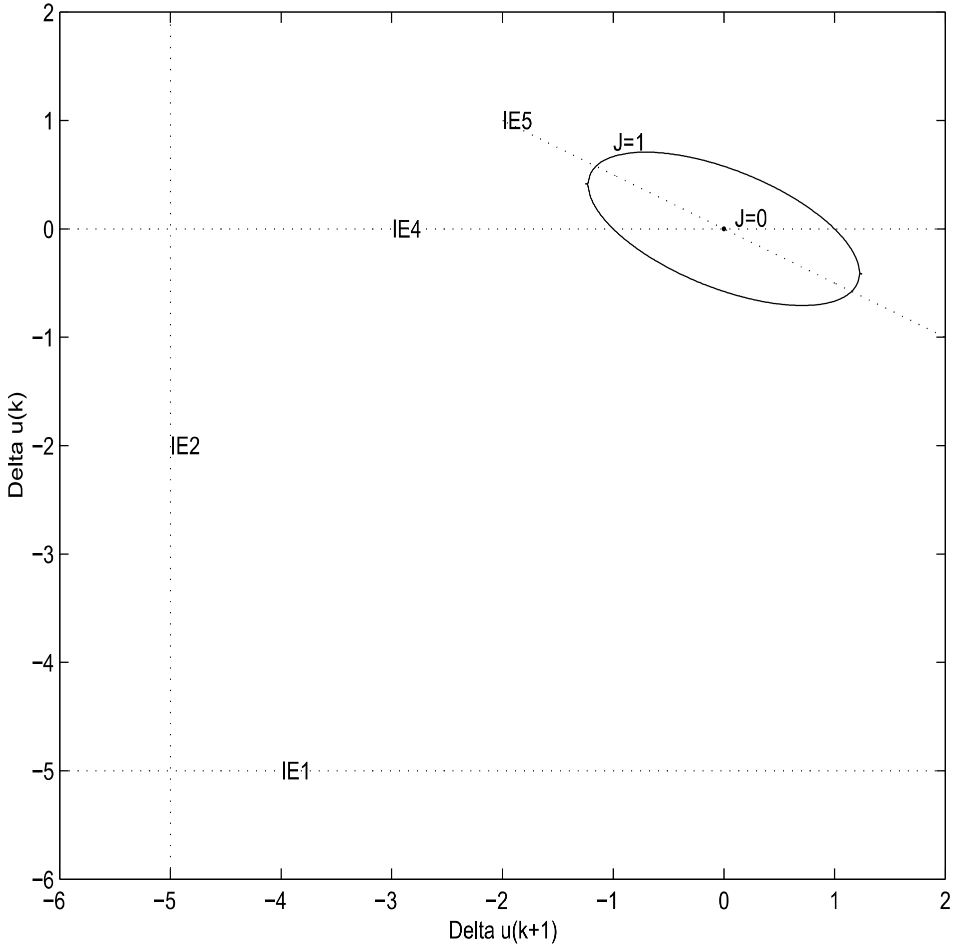

Figure 3.6

Chapter 4

Tutorial

Q.1 Consider the first-order process model

$$x(k+1) = ax(k) + w(k) \qquad (4.1)$$

$$y(k) = x(k) + v(k) \qquad (4.2)$$

which is also the model used by the Kalman filter where $w(k)$ and $v(k)$ are independent Gaussian random variables with variances R_1 and R_2 respectively. The true state and noisy measurement are given by $x(k)$ and $y(k)$ respectively. $|a| \leq 1$. After the Kalman filter gains reached steady-state, find the $\hat{x}(k|k)$ for the following cases (a) $R_1 > 0$, $R_2 = 0$ (b) $R_1 = 0$, $R_2 > 0$ and (c) $a = 1$, $0 \leq \lambda \leq 10$. Plot $K_f(k)$, $K(k)$ and $\frac{P(k|k)}{R_2}$ against λ where $\lambda = \frac{R_1}{R_2}$ $\square$

At steady-state, let $K_f(k) = K_f$, $K(k) = K$, $P(k|k) = P_f$, $P(k+1|k) = P$. After substituting from Equation (1.53), Equation (1.57) gives

$$P = a^2 P + R_1 - \frac{a^2 P^2}{P + R_2}$$
$$P^2 + ((1-a^2)R_2 - R_1)P - R_1 R_2 = 0 \tag{4.3}$$

(a) Since $R_2 = 0$, (1.52) and (1.54) give $K_f(k) = 1$ and $\hat{x}(k|k) = y(k)$ respectively. Notice that $\hat{x}(k|k)$ completely follows the measurement $y(k)$. This can be expected, since $R_2 = 0$ gives $v(k) = 0$ and (4.2) simply gives $\hat{x}(k|k) = y(k)$.

(b) Note that since P is the variance, $P \geq 0$. Solving (4.3) with $R_1 = 0$ gives $P = 0$. Equations (1.52) and (1.53) give $K_f = K = 0$. Equation (1.35) gives $\hat{x}(k|k) = a\hat{x}(k-1|k-1)$. Notice that $\hat{x}(k|k)$ completely ignored the measurment $y(k)$. This is expected since $R_1 = 0$ gives $w(k) = 0$ and (4.1) simply gives $\hat{x}(k|k) = a\hat{x}(k-1|k-1)$. Equation (1.56) gives $P_f = 0$.

(c) Equatons (1.52), (1.53) and (1.56) gives

$$K_f = \frac{K}{a} = \frac{P_f}{R_2} = \frac{P}{P + R_2} \tag{4.4}$$

Equation (4.3) gives

$$P = \frac{1}{2}\left(R_1 - (1-a^2)R_2 + \sqrt{((1-a^2)R_2 - R_1)^2 + 4R_1R_2}\right)$$

Substitute into (4.4) gives

$$K_f = \frac{K}{a} = \frac{P_f}{R_2} = \frac{\lambda - (1-a^2) + \sqrt{\lambda^2 - 2\lambda(1-a^2) + (1-a^2)^2 + 4\lambda}}{\lambda - (1-a^2) + \sqrt{\lambda^2 - 2\lambda(1-a^2) + (1-a^2)^2 + 4\lambda} + 2}$$

if we divide the numerator and denominator by R_2.

For $a = 1$, the plot of $K_f = K = \frac{P_f}{R_2}$ against λ is given in Figure 4.1. The figure shows that when
(i) $\lambda \to \infty$ the variance of the estimation error $\mathbb{E}[(x(k) - \hat{x}(k|k))^2] = P_f = R_2$
(ii) $\lambda = 0$, $\mathbb{E}[(x(k) - \hat{x}(k|k))^2] = P_f = 0$

Equation (1.35) gives

$$\begin{aligned} \hat{x}(k|k) &= a\hat{x}(k-1|k-1) + K_f(y(k) - a\hat{x}(k-1|k-1)) \\ &= (1-K_f)a\hat{x}(k-1|k-1) + K_f y(k) \\ &= (1-K_f) \times (\text{answer from process model}) + K_f \times (\text{answer from measurement}) \end{aligned} \tag{4.5}$$

where

$$\text{answer from process model} = a\hat{x}(k-1|k-1) \tag{4.6}$$

$$\text{answer from measurement} = y(k) \tag{4.7}$$

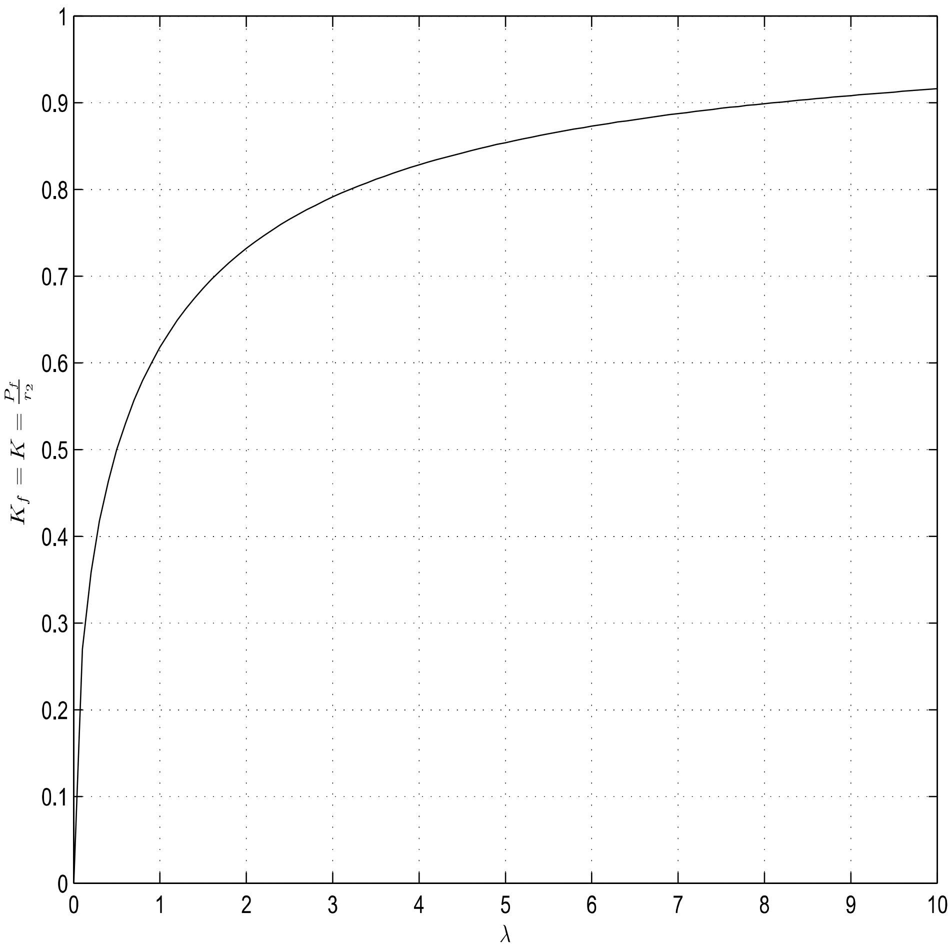

Figure 4.1

Note that the relative contribution of the 2 answers depend on the $\lambda = \frac{R_1}{R_2}$ ratio. If we increase the ratio then $\hat{x}(k|k)$ increasingly follows the 'answer from measurement', $y(k)$, and $\hat{x}(k|k)$ completely follows the 'answer from measurement', $y(k)$, and completely ignores the 'answer from process model', $a\hat{x}(k-1|k-1)$ is reached when $\lambda = \frac{R_1}{R_2} = \infty$ and $K_f = 1$. On the other hand, if we decrease the ratio then $\hat{x}(k|k)$ increasingly ignores the 'answer from measurement', $y(k)$, and $\hat{x}(k|k)$ completely ignore the 'answer from measurement', $y(k)$, and completely follows the 'answer from process model' $a\hat{x}(k-1|k-1)$ is reached when $\lambda = \frac{R_1}{R_2} = 0$ and $K_f = 0$.

We can have an intuitive understanding as follows.

(i) For $w(k) = 0$, the process model of (4.1) and the 'answer from process model' in (4.6) both say that the state is $a \times$ previous state.
(ii) For $v(k) = 0$, the measurement model of Equation (4.2) and the 'answer from measurement' in (4.7) both say that the state is given by the measurement.

So the state is $a \times$ previous state or the measurement ?. The final estimate of the state is a combination of $a \times$ previous state and measurement and how much of which depends on K_f or the ratio of the noise $\lambda = \frac{R_1}{R2}$.

Q.2 The derivation of the Kalman filter assumes that the model used by the Kalman filter matched the model of the true state. In practice, the model assumed by the Kalman filter may be different from the model of the true state.

In an experiment, a target moves at a constant speed of $1ms^{-1}$ i.e. the true state position $x(k)$ is given by

$$x(k+1) = x(k) + 1 \tag{4.8}$$
$$y(k) = x(k) + v(k) \tag{4.9}$$

where $x(0) = 0$. From the sensor data sheet, the measurement noise $v(k)$ is an independent Gaussian random variable with variance $\mathbb{E}v(k)^2 = 1$. In the experiment, the noisy position measurements taken at interval of $1s$ are

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$y(k)$	0	1	2	3	4.3613	5.6733	6.1562	6.0712	8.4228	9.3278	9.7632	10.3418	10.6856

k	13	14	15	16	17	18	19	20
$y(k)$	13.7269	12.0762	15.6357	18.4533	15.7299	17.7600	19.0046	20.5496

A Kalman filter uses the folloiwng constant position model which does not match the true state model to track the target

$$x'(k+1) = x'(k) + w'(k) \tag{4.10}$$
$$y(k) = x'(k) + v'(k) \tag{4.11}$$

where $w'(k)$ and $v'(k)$ are zero-mean independent Gaussian noise with variances $R_1 = \mathbb{E}w'(k)^2 = \sigma_{w'}^2$ and $R_2 = \mathbb{E}v'(k)^2 = \sigma_{v'}^2$ respectively. The initial conditions are $\hat{x}'(0|-1) = 0$ and $P(0|-1) = 1 \times 10^5$.

Find $\hat{x}'(k|k)$ using the Kalman Filter. Find the time average and standard deviation of $x(k) - \hat{x}'(k|k)$ and $x(k) - y(k)$. Let
(a) $R_1 = \sigma_{w'}^2 = 0$, $R_2 = \sigma_{v'}^2 = 1$.
(b) $R_1 = \sigma_{w'}^2 = 10000$, $R_2 = \sigma_{v'}^2 = 1$
(c) $R_1 = \sigma_{w'}^2 = 1$, $R_2 = \sigma_{v'}^2 = 1$,
(d) $R_1 = \sigma_{w'}^2 = 2$, $R_2 = \sigma_{v'}^2 = 1$,
(e) Instead of using the constant position model, repeat part (a) using the following constant velocity

model with sampling period $T = 1s$ for the Kalman filter

$$\begin{bmatrix} x_1'(k+1) \\ x_2'(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1'(k) \\ x_2'(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2}T \\ T \end{bmatrix} w'(k) \tag{4.12}$$

$$y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1'(k) \\ x_2'(k) \end{bmatrix} + v'(k) \tag{4.13}$$

where

$$\begin{aligned} R_1 &= \mathbb{E}\left\{ \begin{bmatrix} \frac{1}{2}T \\ T \end{bmatrix} \begin{bmatrix} \frac{1}{2}T & T \end{bmatrix} w'(k)^2 \right\} = \begin{bmatrix} \frac{1}{4}T^4 & \frac{1}{2}T^3 \\ \frac{1}{2}T^3 & T^2 \end{bmatrix} \sigma_{w'}^2 \\ R_2 &= \mathbb{E}\left\{ v'(k)^2 \right\} = \sigma_{v'}^2 \\ \sigma_{w'}^2 &= 0 \\ \sigma_{v'}^2 &= 1 \\ T &= 1 \end{aligned}$$

Start the simulation with $P(0|-1) = 1 \times 10^5 I_2$ and $x(0|-1) = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$.

(a) The estimate $\hat{x}'(k|k)$ is given by the $\circ$ in Figure 4.2. The detail calculations for $\hat{x}'(0|0)$, $\hat{x}'(1|1)$, and $\hat{x}'(2|2)$ are given in Example 2 in the lecture notes.

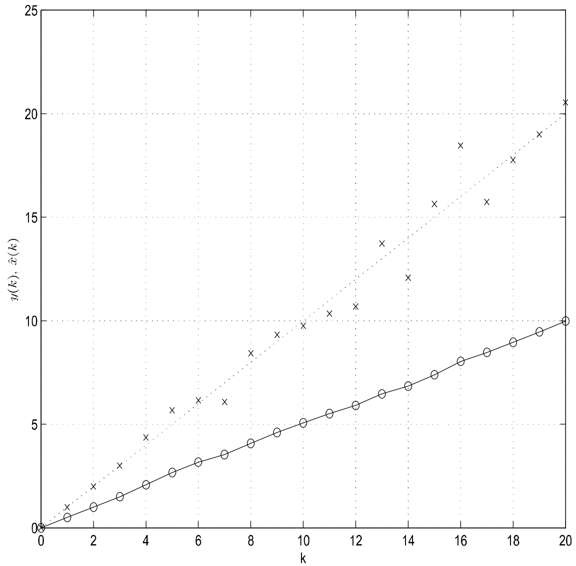

Figure 4.2: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 2

	average	standard deviation
$x(k) - y(k)$	0.0124	0.9109
$x(k) - \hat{x}'(k\|k)$	4.9894	3.1404

Remark: The model of (4.10) and (4.11) with $\sigma_{w'} = 0$ for the Kalman Filter gives at steady-state the estimate $\hat{x}'(k|k) = \hat{x}'(k-1|k-1)$ (see Tutorial Q.1) which is a stationary (not moving) target and clearly not correct. Notice that results in Row 2 of Table 2 are very far away from Row 1. The detail results are given in Table 3.

Table 3

k	0	1	2	3	4	5	6	7	8	9
$x(k)$	0	1	2	3	4	5	6	7	8	9
$y(k)$	0	1.0000	2.0000	3.0000	4.3613	5.6733	6.1562	6.0712	8.4228	9.3278
$K_f(k)$	1.0000	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429	0.1250	0.1111	0.1000
$K(k)$	1.0000	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429	0.1250	0.1111	0.1000
$\hat{x}'(k\|k)$	0	0.5000	1.0000	1.5000	2.0723	2.6724	3.1701	3.5327	4.0761	4.6013
$\hat{x}'(k+1\|k)$	0	0.5000	1.0000	1.5000	2.0723	2.6724	3.1701	3.5327	4.0761	4.6013
$P(k)$	1.0000	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429	0.1250	0.1111	0.1000
$P(k+1\|k)$	1.0000	0.5000	0.3333	0.2500	0.2000	0.1667	0.1429	0.1250	0.1111	0.1000

k	10	11	12	13	14	15	16	17	18	19	20
$x(k)$	10	11	12	13	14	15	16	17	18	19	20
$y(k)$	9.7632	10.3418	10.6856	13.7269	12.0762	15.6357	18.4533	15.7299	17.7600	19.0046	20.5496
$K_f(k)$	0.0909	0.0833	0.0769	0.0714	0.0667	0.0625	0.0588	0.0556	0.0526	0.0500	0.0476
$K(k)$	0.0909	0.0833	0.0769	0.0714	0.0667	0.0625	0.0588	0.0556	0.0526	0.0500	0.0476
$\hat{x}'(k\|k)$	5.0705	5.5098	5.9079	6.4664	6.8404	7.3901	8.0409	8.4681	8.9571	9.4595	9.9876
$\hat{x}'(k+1\|k)$	5.0705	5.5098	5.9079	6.4664	6.8404	7.3901	8.0409	8.4681	8.9571	9.4595	9.9876
$P(k)$	0.0909	0.0833	0.0769	0.0714	0.0667	0.0625	0.0588	0.0556	0.0526	0.0500	0.0476
$P(k+1\|k)$	0.0909	0.0833	0.0769	0.0714	0.0667	0.0625	0.0588	0.0556	0.0526	0.0500	0.0476

(b) The estimate $\hat{x}'(k|k)$ is given by the $\circ$ in Figure 4.3.

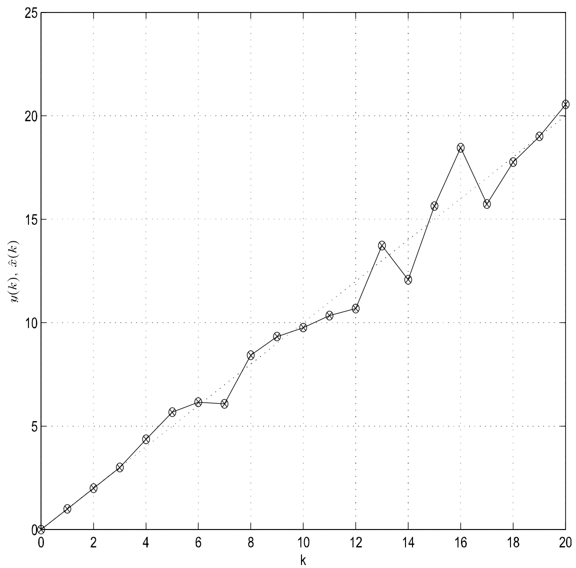

Figure 4.3: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 4

	average	standard deviation
$x(k) - y(k)$	0.0124	0.9109
$x(k) - \hat{x}'(k\|k)$	0.0124	0.9109

Remark: There is absolutely no filtering of the measurement $y(k)$ (see Tutorial Q.1). Notice that Row 2 of Table 4 is the same as Row 1. The detail results are given in Table 5.

Table 5

k	0	1	2	3	4	5	6	7	8	9
$x(k)$	0	1	2	3	4	5	6	7	8	9
$y(k)$	0	1.0000	2.0000	3.0000	4.3613	5.6733	6.1562	6.0712	8.4228	9.3278
$K_f(k)$	1	1	1	1	1	1	1	1	1	1
$K(k)$	1	1	1	1	1	1	1	1	1	1
$\hat{x}'(k\|k)$	0	1	2	3	4.3612	5.6732	6.1562	6.0712	8.4226	9.3277
$\hat{x}'(k+1\|k)$	0	1	2	3	4.3612	5.6732	6.1562	6.0712	8.4226	9.3277
$P(k)$	1	1	1	1	1	1	1	1	1	1
$P(k+1\|k)$	1	1	1	1	1	1	1	1	1	1

k	10	11	12	13	14	15	16	17	18	19	20
$x(k)$	10	11	12	13	14	15	16	17	18	19	20
$y(k)$	9.7632	10.3418	10.6856	13.7269	12.0762	15.6357	18.4533	15.7299	17.7600	19.0046	20.5496
$K_f(k)$	1	1	1	1	1	1	1	1	1	1	1
$K(k)$	1	1	1	1	1	1	1	1	1	1	1
$\hat{x}'(k\|k)$	9.7632	10.3417	10.6856	13.7266	12.0764	15.6353	18.4530	15.7302	17.7598	19.0045	20.5494
$\hat{x}'(k+1\|k)$	9.7632	10.3417	10.6856	13.7266	12.0764	15.6353	18.4530	15.7302	17.7598	19.0045	20.5494
$P(k)$	1	1	1	1	1	1	1	1	1	1	1
$P(k+1\|k)$	1	1	1	1	1	1	1	1	1	1	1

(c) The estimate $\hat{x}(k|k)$ is given by the $\circ$ in Figure 4.4.

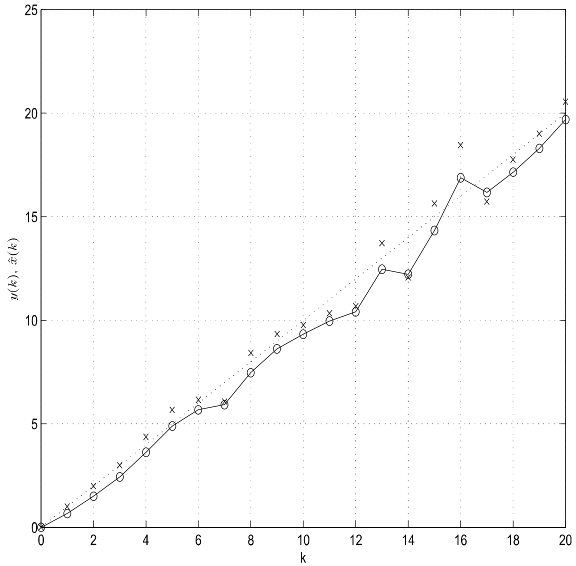

Figure 4.4: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 6

	average	standard deviation
$x(k) - y(k)$	0.0124	0.9109
$x(k) - \hat{x}'(k\|k)$	0.5873	0.5509

Remark: Compare Row 1 and 2 of Table 6 shows that we improve the standard deviation at the expense of the mean. Whether this is acceptable depends on application. The detail results are given in Table 7.

Table 7

k	0	1	2	3	4	5	6	7	8	9
$x(k)$	0	1	2	3	4	5	6	7	8	9
$y(k)$	0	1.0000	2.0000	3.0000	4.3613	5.6733	6.1562	6.0712	8.4228	9.3278
$K_f(k)$	1.0000	0.6667	0.6250	0.6190	0.6182	0.6181	0.6180	0.6180	0.6180	0.6180
$K(k)$	1.0000	0.6667	0.6250	0.6190	0.6182	0.6181	0.6180	0.6180	0.6180	0.6180
$\hat{x}'(k\|k)$	0	0.6667	1.5000	2.4286	3.6233	4.8903	5.6727	5.9190	7.4664	8.6168
$\hat{x}'(k+1\|k)$	0	0.6667	1.5000	2.4286	3.6233	4.8903	5.6727	5.9190	7.4664	8.6168
$P(k\|k)$	1.0000	0.6667	0.6250	0.6190	0.6182	0.6181	0.6180	0.6180	0.6180	0.6180
$P(k+1\|k)$	2.0000	1.6667	1.6250	1.6190	1.6182	1.6181	1.6180	1.6180	1.6180	1.6180

k	10	11	12	13	14	15	16	17	18	19	20
$x(k)$	10	11	12	13	14	15	16	17	18	19	20
$y(k)$	9.7632	10.3418	10.6856	13.7269	12.0762	15.6357	18.4533	15.7299	17.7600	19.0046	20.5496
$K_f(k)$	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180
$K(k)$	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180
$\hat{x}'(k\|k)$	9.3253	9.9535	10.4060	12.4584	12.2222	14.3319	16.8790	16.1688	17.1522	18.2971	19.6892
$\hat{x}'(k+1\|k)$	9.3253	9.9535	10.4060	12.4584	12.2222	14.3319	16.8790	16.1688	17.1522	18.2971	19.6892
$P(k\|k)$	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180	0.6180
$P(k+1\|k)$	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180	1.6180

(d) The estimate $\hat{x}(k|k)$ is given by the $\circ$ in Figure 4.5.

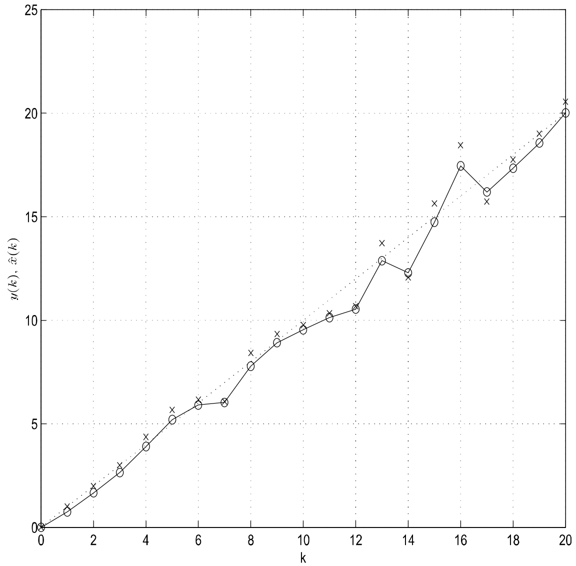

Figure 4.5: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 8

	average	standard deviation
$x(k) - y(k)$	0.0124	0.9109
$x(k) - \hat{x}'(k\|k)$	0.3600	0.6398

Remark: Compare Row 1 and 2 of Table 8 shows that we also improve the standard deviation at the expense of the mean. Whether this is acceptable depends on application and R_1 can be used as a tuning parameter. The detail results are given in Table 9.

Table 9

k	0	1	2	3	4	5	6	7	8	9
$x(k)$	0	1	2	3	4	5	6	7	8	9
$y(k)$	0	1.0000	2.0000	3.0000	4.3613	5.6733	6.1562	6.0712	8.4228	9.3278
$K_f(k)$	1.0000	0.7500	0.7333	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321
$K(k)$	1.0000	0.7500	0.7333	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321
$\hat{x}'(k\|k)$	0	0.7500	1.6667	2.6429	3.9009	5.1984	5.8996	6.0252	7.7804	8.9132
$\hat{x}'(k+1\|k)$	0	0.7500	1.6667	2.6429	3.9009	5.1984	5.8996	6.0252	7.7804	8.9132

k	10	11	12	13	14	15	16	17	18	19	20
$x(k)$	10	11	12	13	14	15	16	17	18	19	20
$y(k)$	9.7632	10.3418	10.6856	13.7269	12.0762	15.6357	18.4533	15.7299	17.7600	19.0046	20.5496
$K_f(k)$	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321
$K(k)$	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321	0.7321
$\hat{x}'(k\|k)$	9.5354	10.1257	10.5356	12.8718	12.2894	14.7391	17.4581	16.1930	17.3401	18.5586	20.0161
$\hat{x}'(k+1\|k)$	9.5354	10.1257	10.5356	12.8718	12.2894	14.7391	17.4581	16.1930	17.3401	18.5586	20.0161

(e) The estimate $\hat{x}(k|k)$ is given by the $\circ$ in Figure 4.6.

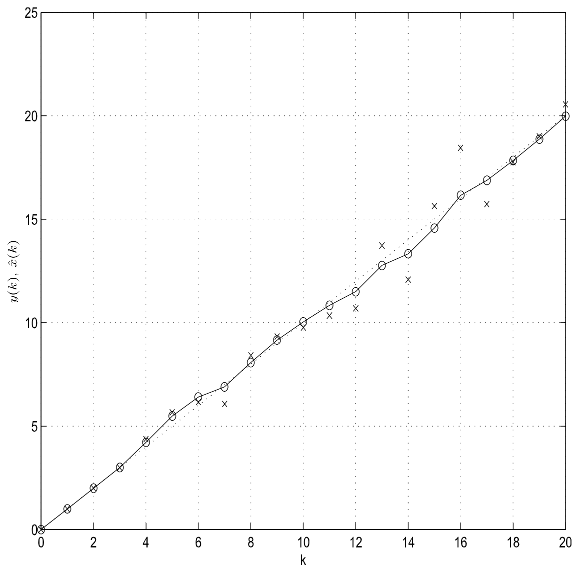

Figure 4.6: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 10

	average	standard deviation
$x(k) - y(k)$	0.0124	0.9109
$x(k) - \hat{x}'(k\|k)$	0.0491	0.2746

Remark: In Row 2 of Table 10, the mean is essentially 0 but the standard deviation improved to 0.2746. Instead of (4.8) and (4.9), the true state position and measurement can also be modeled in

(4.12) and (4.13) with $\sigma^2_{w'} = 0$, $\sigma^2_{v'} = 1$ and initial condition $x'(k) = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$ giving $x'_2(k) = 1$ and

$$\begin{aligned} x'_1(k+1) &= x'_1(k) + 1 \\ y(k) &= x'_1(k) + v(k) \end{aligned}$$

Hence the model used by the Kalman filter matched the process model. The detail results for $k = 0 \ldots 5$ are given Table 11.

Table 11

k	0		1		2		3		4		5	
$y(k)$	0		1		2		3		4.3613		5.6733	
$K_f(k)$	1		1		0.8333		0.7		0.6		0.5238	
	0		1		0.5		0.3		0.2		0.1429	
$K(k)$	1		2		1.3333		1		0.8		0.6667	
	0		1		0.5		0.3		0.2		0.1429	
$\hat{x}'(k\|k)$	0		1		2		3		4.2168		5.4903	
	0		1		1		1		1.0723		1.1272	
$\hat{x}'(k+1\|k)$	0		2		3		4		5.289		6.6175	
	0		1		1		1		1.0723		1.1272	
$P(k\|k)$	1	0	1	1	0.8333	0.5	0.7	0.3	0.6	0.2	0.5238	0.1429
	0	100000	1	2	0.5	0.5	0.3	0.2	0.2	0.1	0.1429	0.0571
$P(k+1\|k)$	100000	100000	5	3	2.3333	1	1.5	0.5	1.1	0.3	0.8667	0.2
	100000	100000	3	2	1	0.5	0.5	0.2	0.3	0.1	0.2	0.0571

Q.3 Repeat Q.2 (a) and (e) for a target that is stationary at 20 i.e. the true state position $x(k)$ is given by

$$x(k+1) = x(k) \tag{4.14}$$

$$y(k) = x(k) + v(k) \tag{4.15}$$

where x(0)=20. From the sensor data sheet, the measurement noise $v(k)$ is an independent Gaussian random variable with variance $\mathbb{E}v(k)^2 = 1$. The initial conditions are $\hat{x}(0|-1) = 0$ and $P(0|-1) = 1 \times 10^5 I$. In the experiment, the noisy position measurements taken at interval of $1s$ are

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$y(k)$	20.7304	17.8389	20.8586	19.8416	20.1098	21.0132	20.5073	21.1431	19.1466	18.8669	20.3414	20.0498	19.0486

k	13	14	15	16	17	18	19	20
$y(k)$	19.9327	18.3149	20.6768	20.2561	20.6644	21.1749	19.3384	19.5482

Repeat Q2(a): The working is similar to Q2(a) and only the results are given here.

Table 12

k	$y(k)$	$K_f(k)$	$K(k)$	$\hat{x}(k\|k)$	$\hat{x}(k+1\|k)$	$P(k\|k)$	$P(k+1\|k)$
0	20.7304	1	1	20.7302	20.7302	1	1
1	17.8389	$\frac{1}{2}$	$\frac{1}{2}$	19.2846	19.2846	$\frac{1}{2}$	$\frac{1}{2}$
2	20.8586	$\frac{1}{3}$	$\frac{1}{3}$	19.8092	19.8092	$\frac{1}{3}$	$\frac{1}{3}$
3	19.8416	$\frac{1}{4}$	$\frac{1}{4}$	19.8173	19.8173	$\frac{1}{4}$	$\frac{1}{4}$
4	20.1098	$\frac{1}{5}$	$\frac{1}{5}$	19.8758	19.8758	$\frac{1}{5}$	$\frac{1}{5}$
5	21.0132	$\frac{1}{6}$	$\frac{1}{6}$	20.0654	20.0654	$\frac{1}{6}$	$\frac{1}{6}$

The estimate $\hat{x}'(k|k)$ is given by the $\circ$ in Figure 4.7.

Table 13

	average	standard deviation
$x(k) - y(k)$	0.0284	0.9333
$x(k) - \hat{x}'(k\|k)$	0.0131	0.2527

Remark: The constant position model of (4.10) and (4.11) matched the true position and measurement (4.14) and (4.15) and we expect to obtain the results predicted by the Kalman filter theory i.e. $E[x(k) - x'(k|k)] = 0$ and $E[(x(k) - \hat{x}'(k|k))^2] = P(k|k)$.

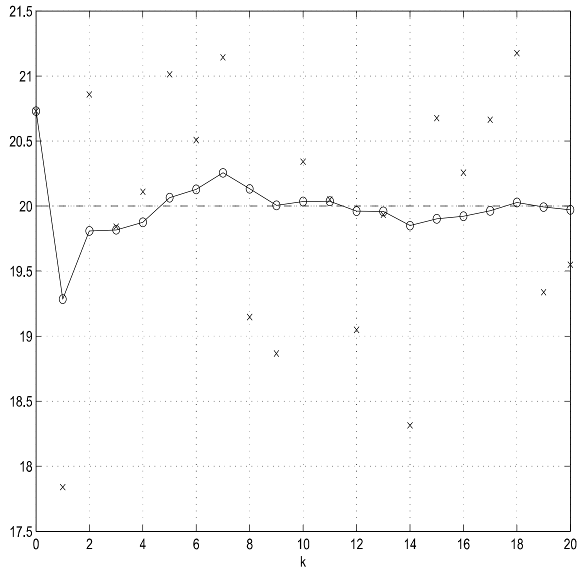

Figure 4.7: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Repeat Q2(e): The estimate $\hat{x}'(k|k)$ is given by the $\circ$ in Figure 4.8.

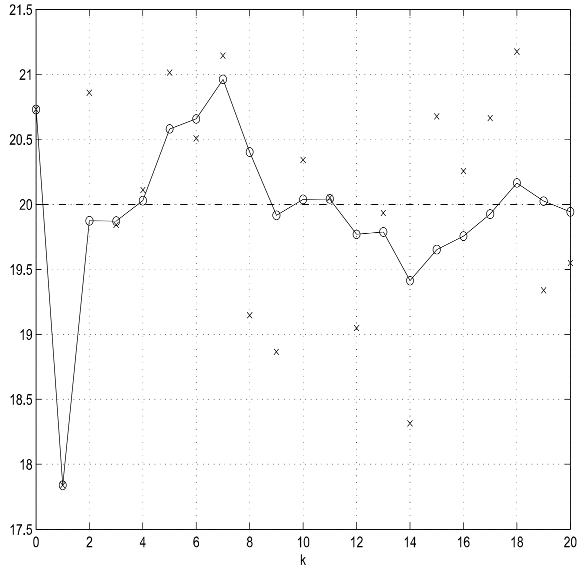

Figure 4.8: The noisy measurement $y(k)$, estimate $\hat{x}'(k|k)$ and true state $x(k)$ are given by the $\times$, $\circ$ and dotted-line respectively.

Table 14

	average	standard deviation
$x(k) - y(k)$	0.0284	0.9333
$x(k) - \hat{x}'(k\|k)$	0.0300	0.6223

Remark: The true state position and measurement (4.14) and (4.15) can also be modeled in (4.12) and (4.13) with $\sigma^2_{w'} = 0$, $\sigma^2_{v'} = 1$, $T = 1$ and initial condition $x'(0) = \begin{bmatrix} 20 & 0 \end{bmatrix}^T$ giving

$$\begin{aligned} x'_1(k+1) &= x'_1(k) \\ y(k) &= x'_1(k) + v(k) \end{aligned}$$

Hence the model used by the Kalman filter matched the process model and we expect to obtain the results predicted by the Kalman filter theory i.e. $E[x(k) - x'(k|k)] = 0$ and $E[(x(k) - \hat{x}'(k|k))^2]$ $= P(k|k)$. Note that the values of $K_f(k)$, $K(k)$, $P(k|k)$ and $P(k+1|k)$ are already given in Table 11.

Q.4 Consider the discrete-time plant

$$G(z) = \frac{Y(z)}{U(z)} = \frac{h_1 z^{-1} + h_2 z^{-2}}{1 + f_1 z^{-1} + f_2 z^{-2}} z^{-3}$$

with sampling interval of 1 second.

(a) Formulate a state-space model with the input and output signals as the states. The advantage of this formulation is we can assume that all the states are measurable because they are the input and output signals and there is no need for an observer or state estimator to estimate the states. □

(b) Give the augmented state-space formulation. □

(c) Let the MPC gains be given by

$$\begin{aligned} K_{mpc} &= \begin{bmatrix} k_1^y & k_2^y & k_1^u & k_2^u & k_3^u & k_4^u & k_0^y \end{bmatrix} \\ K_r &= k_r \end{aligned}$$

Give the MPC in transfer function form. □

(d) Find the closed-loop transfer function. □

(e) Draw the closed-loop block diagram. □

(f) Given $h_1 = 1$, $h_2 = -0.1$, $f_1 = -1.6$, $f_2 = 0.68$, $N_c = 1$, $N_p = 5$, $r_w = 2$, find K_r and K_{mpc}. □

(g) Given

$$
\begin{aligned}
r(k) &= \begin{cases} 0 & \text{for } k < 0 \\ 1 & \text{for } k \geq 0 \end{cases} \\
u(k) &= 0 \quad \text{for} \quad k < 0 \\
y(k) &= 0 \quad \text{for} \quad k \leq 0
\end{aligned}
$$

find $y(k)$ for $k = 0,\ 1,\ \ldots,\ 9.$ $\square$

(a) The discrete-time plant can be written as

$$\begin{aligned} y(k) &= -f_1 y(k-1) - f_2 y(k-2) + h_1 u(k-4) + h_2 u(k-5) & (4.16) \\ y(k+1) &= -f_1 y(k) - f_2 y(k-1) + h_1 u(k-3) + h_2 u(k-4) & (4.17) \end{aligned}$$

(4.17) − (4.16) gives

$$\begin{aligned} y(k+1) - y(k) = \Delta y(k+1) &= -f_1(y(k) - y(k-1)) - f_2(y(k-1) - y(k-2)) \\ &\quad + h_1(u(k-3) - u(k-4)) + h_2(u(k-4) - u(k-5)) \\ &= -f_1 \Delta y(k) - f_2 \Delta y(k-1) + h_1 \Delta u(k-3) + h_2 \Delta u(k-4) \end{aligned}$$

Let the state be given by

$$x_p(k) = \begin{bmatrix} y(k) & y(k-1) & u(k-1) & u(k-2) & u(k-3) & u(k-4) \end{bmatrix}^T$$

and the state-space model

$$
\overbrace{\begin{bmatrix} y(k+1) \\ y(k) \\ u(k) \\ u(k-1) \\ u(k-2) \\ u(k-3) \end{bmatrix}}^{x_p(k+1)}
=
\overbrace{\begin{bmatrix} -f_1 & -f_2 & 0 & 0 & h_1 & h_2 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}}^{A_p}
\overbrace{\begin{bmatrix} y(k) \\ y(k-1) \\ u(k-1) \\ u(k-2) \\ u(k-3) \\ u(k-4) \end{bmatrix}}^{x_p(k)}
+
\overbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{B_p} u(k)
$$

$$
y(k) = \overbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}^{C_p}
\overbrace{\begin{bmatrix} y(k) \\ y(k-1) \\ u(k-1) \\ u(k-2) \\ u(k-3) \\ u(k-4) \end{bmatrix}}^{x_p(k)}
$$

(b) Equations (2.5) and (2.6) give

$$\overbrace{\begin{bmatrix} \Delta y(k+1) \\ \Delta y(k) \\ \Delta u(k) \\ \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \\ y(k+1) \end{bmatrix}}^{x(k+1)} = \overbrace{\begin{bmatrix} -f_1 & -f_2 & 0 & 0 & h_1 & h_2 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -f_1 & -f_2 & 0 & 0 & h_1 & h_2 & 1 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} \Delta y(k) \\ \Delta y(k-1) \\ \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \\ \Delta u(k-4) \\ y(k) \end{bmatrix}}^{x(k)} + \overbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{B} u(k)$$

$$y(k) = \overbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}^{C} \overbrace{\begin{bmatrix} \Delta y(k) \\ \Delta y(k-1) \\ \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \\ \Delta u(k-4) \\ y(k) \end{bmatrix}}^{x(k)}$$

(c) From (2.29)

$$\Delta u(k) = k_r r(k) - \left[\begin{array}{ccccccc} k_1^y & k_2^y & k_1^u & k_2^u & k_3^u & k_4^u & k_0^y \end{array}\right] \left[\begin{array}{c} \Delta y(k) \\ \Delta y(k-1) \\ \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \\ \Delta u(k-4) \\ y(k) \end{array}\right]$$

Take z-transform

$$(1-z^{-1})U(z) = k_r R(z) - \left[\begin{array}{ccccccc} k_1^y & k_2^y & k_1^u & k_2^u & k_3^u & k_4^u & k_0^y \end{array}\right] \left[\begin{array}{c} (1-z^{-1}Y(z) \\ z^{-1}(1-z^{-1})Y(z) \\ z^{-1}(1-z^{-1})U(z) \\ z^{-2}(1-z^{-1})U(z) \\ z^{-3}(1-z^{-1})U(z) \\ z^{-4}(1-z^{-1})U(z) \\ Y(z) \end{array}\right]$$

$$
\begin{aligned}
(1-z^{-1})U(z) &= k_r R(z) - (k_1^y + k_2^y z^{-1})(1-z^{-1})Y(z) \\
&\quad -(k_1^u z^{-1} + k_2^u z^{-2} + k_3^u z^{-3} + k_4^u z^{-4})(1-z^{-1})U(z) - k_0^y Y(z) \\
U(z) &= \frac{k_r R(z) - k_0^y Y(z)}{Q(z)L(z)} - \frac{P(z)Y(z)}{L(z)}
\end{aligned}
$$

where

$$
\begin{aligned}
L(z) &= 1 + k_1^u z^{-1} + k_2^u z^{-2} + k_3^u z^{-3} + k_4^u z^{-4} \\
P(z) &= k_1^y + k_2^y z^{-1} \\
Q(z) &= 1 - z^{-1}
\end{aligned}
$$

(d) Let

$$
G(z) = \frac{N(z)}{D(z)}
$$

where

$$
\begin{aligned}
N(z) &= h_1 z^{-4} + h_2 z^{-5} \\
D(z) &= 1 + f_1 z^{-1} + f_2 z^{-2}
\end{aligned}
$$

Then

$$
\begin{aligned}
Y(z) &= G(z)U(z) = \frac{N(z)}{D(z)}U(z) \\
&= \frac{N(z)}{D(z)}\left(\frac{k_r R(z) - k_0^y Y(z)}{Q(z)L(z)} - \frac{P(z)Y(z)}{L(z)}\right) \\
\frac{Y(z)}{R(z)} &= \frac{k_r N(z)}{Q(z)D(z)L(z) + Q(z)N(z)P(z) + k_0^y N(z)}
\end{aligned}
\tag{4.18}
$$

Divide numerator and denominator by $Q(z)D(z)L(z)$

$$
\frac{Y(z)}{R(z)} = \frac{\frac{k_r}{Q(z)L(z)}G(z)}{1 + G(z)\left(\frac{P(z)}{L(z)} + \frac{k_0^y}{Q(z)L(z)}\right)}
\tag{4.19}
$$

(e) See Figure 4.9.

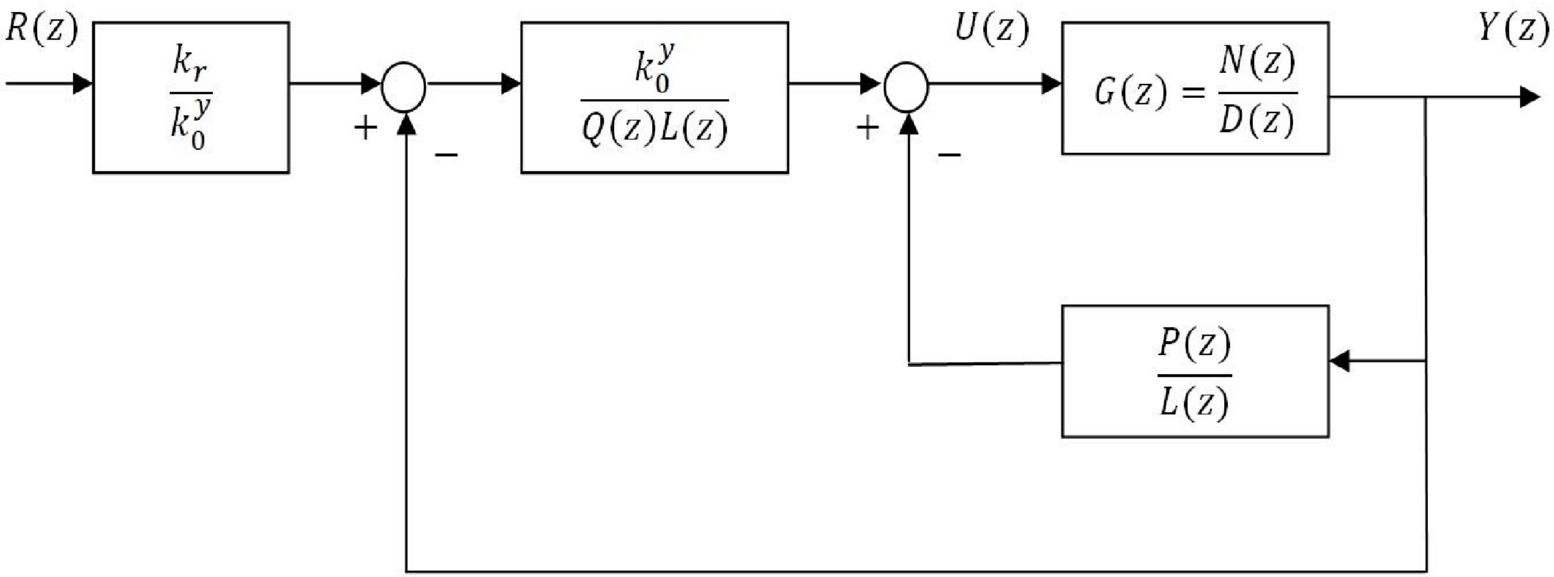

Figure 4.9

(f) From (2.18), (2.21), (2.27), (2.17), (2.30) and (2.31)

$$\Phi = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 2.5 \end{bmatrix}; \qquad \bar{R} = 2; \qquad \bar{R}_s = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 1.6000 & -0.6800 & 0 & 0 & 1.0000 & -0.1000 & 1.0000 \\ 3.4800 & -1.7680 & 0 & 1.0000 & 2.5000 & -0.2600 & 1.0000 \\ 5.4000 & -3.0464 & 1.0000 & 2.5000 & 4.2200 & -0.4480 & 1.0000 \\ 7.1936 & -4.3520 & 2.5000 & 4.2200 & 5.9520 & -0.6400 & 1.0000 \\ 8.7578 & -5.5716 & 4.2200 & 5.9520 & 7.5536 & -0.8194 & 1.0000 \end{bmatrix}$$

$$K_r = 0.3784$$

$$K_{mpc} = \begin{bmatrix} 3.1446 & -1.9763 & 1.4108 & 2.0649 & 2.6850 & -0.2906 & 0.3784 \end{bmatrix}$$

(g) Equation (4.18) gives

$$\frac{Y(z)}{R(z)} = \frac{0.3784z^{-4} - 0.0378z^{-5}}{1 - 1.1892z^{-1} + 0.6768z^{-2} - 0.147z^{-3}}$$

$$y(k) = 1.1892y(k-1) - 0.6768y(k-2) + 0.147y(k-3) + 0.3784r(k-4) - 0.0378r(k-5)$$

The step response is given in the table below and Figure 4.10.

k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$y(k)$	0	0	0	0	0.3784	0.7905	1.0245	1.0796	1.0472	1.0059	0.9868	0.9872	0.9946	1.0003	1.0021

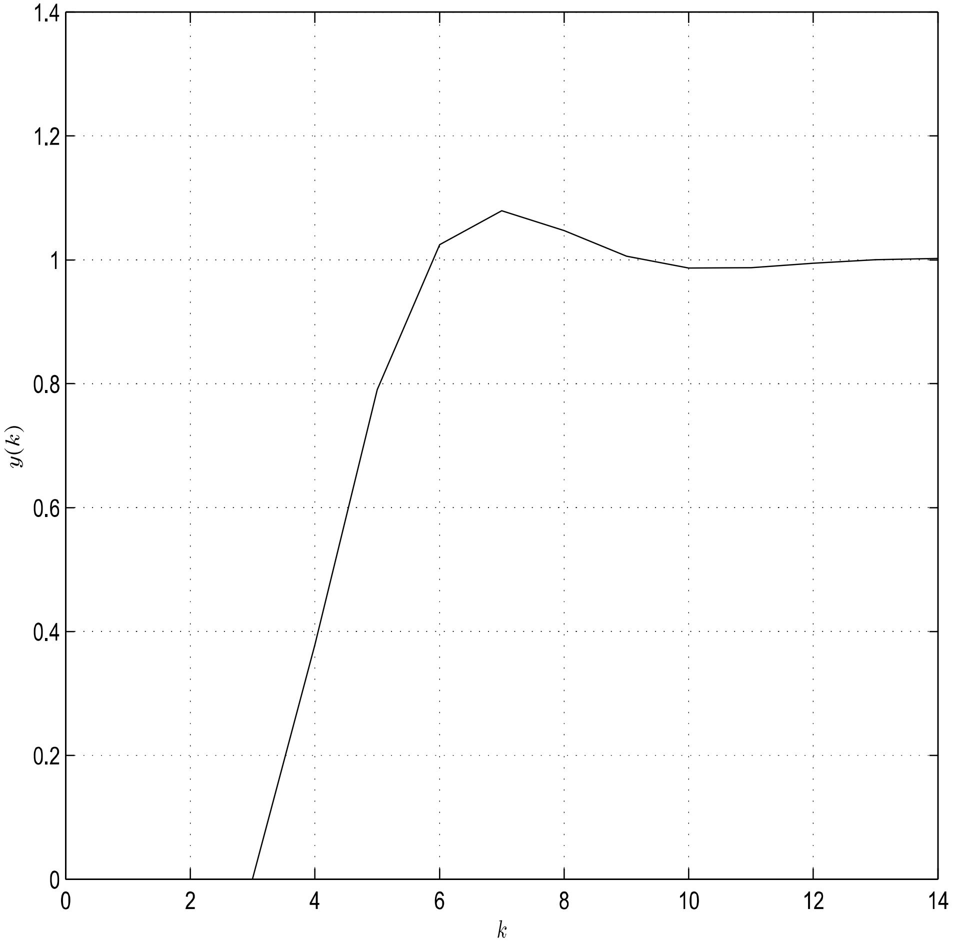

Figure 4.10

Q.5 Assume that the state $x(k)$ in Example 7 is not measurable. The Kalman filter (predicted estimate) can is used as an estimate of the state $\hat{x}(k)$. For the sake of computing the observer gain K_{ob}, pretend that the model is corrupted by noise as follows.

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

where

$$w(k) = \begin{bmatrix} w_1(k) \\ 0 \end{bmatrix}$$

The covariance matrix

$$\begin{aligned} R_1 &= \mathbb{E}w(k)w(k)^T = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \\ R_2 &= \mathbb{E}v(k)^2 = 0.1 \end{aligned}$$

(a) Verify that the steady-state Kalman filter covariance matrix and gain converged to

$$
\begin{aligned}
P(k|k-1) = P_{ob} &= \begin{bmatrix} 1.7229 & 0.7834 \\ 0.7834 & 0.9344 \end{bmatrix} \\
K(k) = K_{ob} &= \begin{bmatrix} 0.6059 \\ 1.5093 \end{bmatrix}
\end{aligned}
$$

□

(b) Initialize $\hat{x}(0) = \begin{bmatrix} -0.1 & -0.1 \end{bmatrix}^T \neq x(0) = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$. Use the estimate $\hat{x}(k)$ to find $\Delta u(k)$ and hence $y(k)$ for $k = 0, 1, 2$. □

(c) Use the closed-loop equation to find $y(k)$, $k = 0, 1, 2, 3$ when the state $x(k)$ in the MPC is replaced by $\hat{x}(k)$ from the steady-state Kalman filter (predicted estimate). Given the initial state $x(1) = \begin{bmatrix} 0 & 0 \end{bmatrix}$ and estimate $\hat{x}(1) = \begin{bmatrix} -0.1 & -0.1 \end{bmatrix}$. □

(d) Give the closed-loop poles from the controller and Kalman filter. □

(e) Use the closed-loop poles found in part (d) to find K_{ob}. □

(a) Substituting P_{ob} and K_{ob} into the RHS of (1.53) and (1.57) give the same P_{ob} and K_{ob} on the LHS. Hence $P(k|k-1)$ and $K(k)$ converged to P_{ob} and K_{ob}.

(b) Replace $x(k)$ by $\hat{x}(k)$ in the MPC formula (2.29) to give

$$\Delta u(k) = K_r r(k) - K_{mpc}\hat{x}(k) \tag{4.20}$$

The observer (2.36) is given as

$$\hat{x}(k+1) = A\hat{x}(k) + B\Delta u(k) + K_{ob}(y(k) - C\hat{x}(k)) \tag{4.21}$$

Equation (4.20) gives $\Delta u(k)$, (2.5) $x(k)$, (2.6) $y(k)$, (4.21) $\hat{x}(k+1)$.

$$\underline{\text{k} = 0}$$

$$\begin{aligned}
y(0) &= Cx(0) \\
&= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} = 0 \\
\Delta u(0) &= K_r r(0) - K_{mpc}\hat{x}(0) \\
&= (4.9819)(1) - \begin{bmatrix} 5.9364 & 4.9819 \end{bmatrix} \begin{bmatrix} -0.1 \\ -0.1 \end{bmatrix} = 6.0737
\end{aligned}$$

$$\underline{\text{k} = 1}$$

$$\begin{aligned}
x(1) &= Ax(0) + B\Delta u(0) \\
&= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} 6.0737 = \begin{bmatrix} 0.6074 \\ 0.6074 \end{bmatrix} \\
y(1) &= Cx(1) \\
&= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0.6074 \\ 0.6074 \end{bmatrix} = 0.6074
\end{aligned}$$

$$
\begin{aligned}
\hat{x}(1) &= A\hat{x}(0) + B\Delta u(0) + K_{ob}(y(0) - C\hat{x}(0)) \\
&= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \begin{bmatrix} -0.1 \\ -0.1 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} 6.0737 + \begin{bmatrix} 0.6059 \\ 1.5093 \end{bmatrix} \left(0 - \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} -0.1 \\ -0.1 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0.5880 \\ 0.5783 \end{bmatrix} \\
\Delta u(1) &= K_r r(1) - K_{mpc}\hat{x}(1) \\
&= (4.9819)(1) - \begin{bmatrix} 5.9364 & 4.9819 \end{bmatrix} \begin{bmatrix} 0.5880 \\ 0.5783 \end{bmatrix} = -1.3895
\end{aligned}
$$

<u>k = 2</u>

$$
\begin{aligned}
x(2) &= Ax(1) + B\Delta u(1) \\
&= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \begin{bmatrix} 0.6074 \\ 0.6074 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} (-1.3895) = \begin{bmatrix} 0.3469 \\ 0.9543 \end{bmatrix} \\
y(2) &= Cx(2) \\
y(2) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0.3469 \\ 0.9543 \end{bmatrix} = 0.9543
\end{aligned}
$$

$$\begin{aligned}
\hat{x}(2) &= A\hat{x}(1) + B\Delta u(1) + K_{ob}(y(1) - C\hat{x}(1)) \\
&= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \begin{bmatrix} 0.5880 \\ 0.5783 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} (-1.3895) + \begin{bmatrix} 0.6059 \\ 1.5093 \end{bmatrix} \left(0.6074 - \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0.5880 \\ 0.5783 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0.3490 \\ 0.9536 \end{bmatrix} \\
\Delta u(2) &= K_r r(2) - K_{mpc}\hat{x}(2) \\
&= (4.9819)(1) - \begin{bmatrix} 5.9364 & 4.9819 \end{bmatrix} \begin{bmatrix} 0.3490 \\ 0.9536 \end{bmatrix} = -1.8409
\end{aligned}$$

<u>k = 3</u>

$$\begin{aligned}
x(3) &= Ax(2) + B\Delta u(2) \\
&= \begin{bmatrix} 0.8 & 0 \\ 0.8 & 1 \end{bmatrix} \begin{bmatrix} 0.3469 \\ 0.9543 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} (-1.8409) = \begin{bmatrix} 0.0935 \\ 1.0478 \end{bmatrix} \\
y(3) &= Cx(3) \\
&= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0.0935 \\ 1.0478 \end{bmatrix} = 1.0478
\end{aligned}$$

(c) Consider (2.42)

$$x_f(k+1) = A_f x_f(k) + B_f r(k) \tag{4.22}$$
$$y_f(k) = C_f x_f(k) \tag{4.23}$$

where

$$x_f(k) = \begin{bmatrix} \tilde{x}(k) \\ x(k) \end{bmatrix} = \begin{bmatrix} x_1(k) - \hat{x}_1(k) \\ x_2(k) - \hat{x}_2(k) \\ x_1(k) \\ x_2(k) \end{bmatrix} \qquad x_f(0) = \begin{bmatrix} 0 - (-0.1) = 0.1 \\ 0 - (-0.1) = 0.1 \\ 0 \\ 0 \end{bmatrix}$$

$$A_f = \begin{bmatrix} A - K_{ob}C & o_{n\times n} \\ BK_{mpc} & A - BK_{mpc} \end{bmatrix} = \begin{bmatrix} 0.8000 & -0.6059 & 0 & 0 \\ 0.8000 & -0.5093 & 0 & 0 \\ 0.5936 & 0.4982 & 0.2064 & -0.4982 \\ 0.5936 & 0.4982 & 0.2064 & 0.5018 \end{bmatrix}$$

$$B_f = \begin{bmatrix} o_{n\times m} \\ BK_r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.4982 \\ 0.4982 \end{bmatrix}$$

$$C_f = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$$

$$r(k) = 1$$

Iterating (4.22) and (4.23) gives $y(0) = 0$, $y(1) = 0.6074$, $y(2) = 0.9543$, $y(3) = 1.0478$ which is equal to the results in Part (b).

(d) The closed-loop poles from observer and controller are given by (2.44) and (2.45). Solving gives $z = 0.1454 \pm j0.2371$ and $z = 0.3541 \pm j0.2846$ respectively.

(e) Equation (2.44) gives

$$\begin{aligned}
\det\left\{\begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} - \left(\begin{bmatrix} 0.8 & 1 \\ 0.8 & 1 \end{bmatrix} - \begin{bmatrix} K_{ob1} \\ K_{ob2} \end{bmatrix}\begin{bmatrix} 0 & 1 \end{bmatrix}\right)\right\} &= 0 \\
\det\begin{bmatrix} z - 0.8 & K_{ob1} \\ -0.8 & z + K_{ob2} - 1 \end{bmatrix} &= 0 \\
z^2 + z(K_{ob2} - 1.8) + 0.8(K_{ob1} - K_{ob2} + 1) &= 0 \qquad (4.24)
\end{aligned}$$

The closed-loop poles found in Part (d) are $z = 0.1454 \pm j0.2371$. The characteristic equation is

$$\begin{aligned}
(z - 0.1454 + j0.2371)(z - 0.1454 - j0.2371) &= 0 \\
z^2 - 0.2908z + 0.0774 &= 0 \qquad (4.25)
\end{aligned}$$

Compare (4.24) with (4.25) gives

$$\begin{aligned}
K_{ob2} - 1.8 &= -0.2908 \\
0.8(K_{ob1} - K_{ob2} + 1) &= 0.0774
\end{aligned}$$

and solving gives $K_{ob} = \begin{bmatrix} 0.6059 & 1.5093 \end{bmatrix}^T$, the K_{ob} given in Part (a). In other words, K_{ob} need not be found from the Kalman filter. It can be determined by specifying the desired closed-loop poles.

Q.6

Consider the same plant and MPC in Example 8. In this question we make the following distinction: $\Delta u(k)$ and $u(k)$ are outputs computed by the MPC while $u_p(k)$ is the input to the plant. Constraint on u_p depends on the plant, constraint on $u(k)$, MPC calculations. If the 2 constraints are equal then $u(k) = u_p(k)$, otherwise they may not be equal (see Figure 4.11).

(a) Let the constraint on $u_p(k)$ be given by $u_p(k) \leq U^{max}$ where $U^{max} = 0.2$ while there is no constraint on $u(k)$. Find $u(k)$, $u_p(k)$ and $y(k)$ for $k = 0, 1, \ldots, 9$. □

(b) Formulate a quadratic programming problem with the constraint $u(k) \leq U^{max}$ where $U^{max} = 0.2$. □

(c) Solve the quadratic programming problem in Part (b). □

(d) If $r_w = 40$ so that $u(k) \leq U^{max}$, find $u(k)$, $u_p(k)$ and $y(k)$ for k =0, 1, $\ldots$9. □

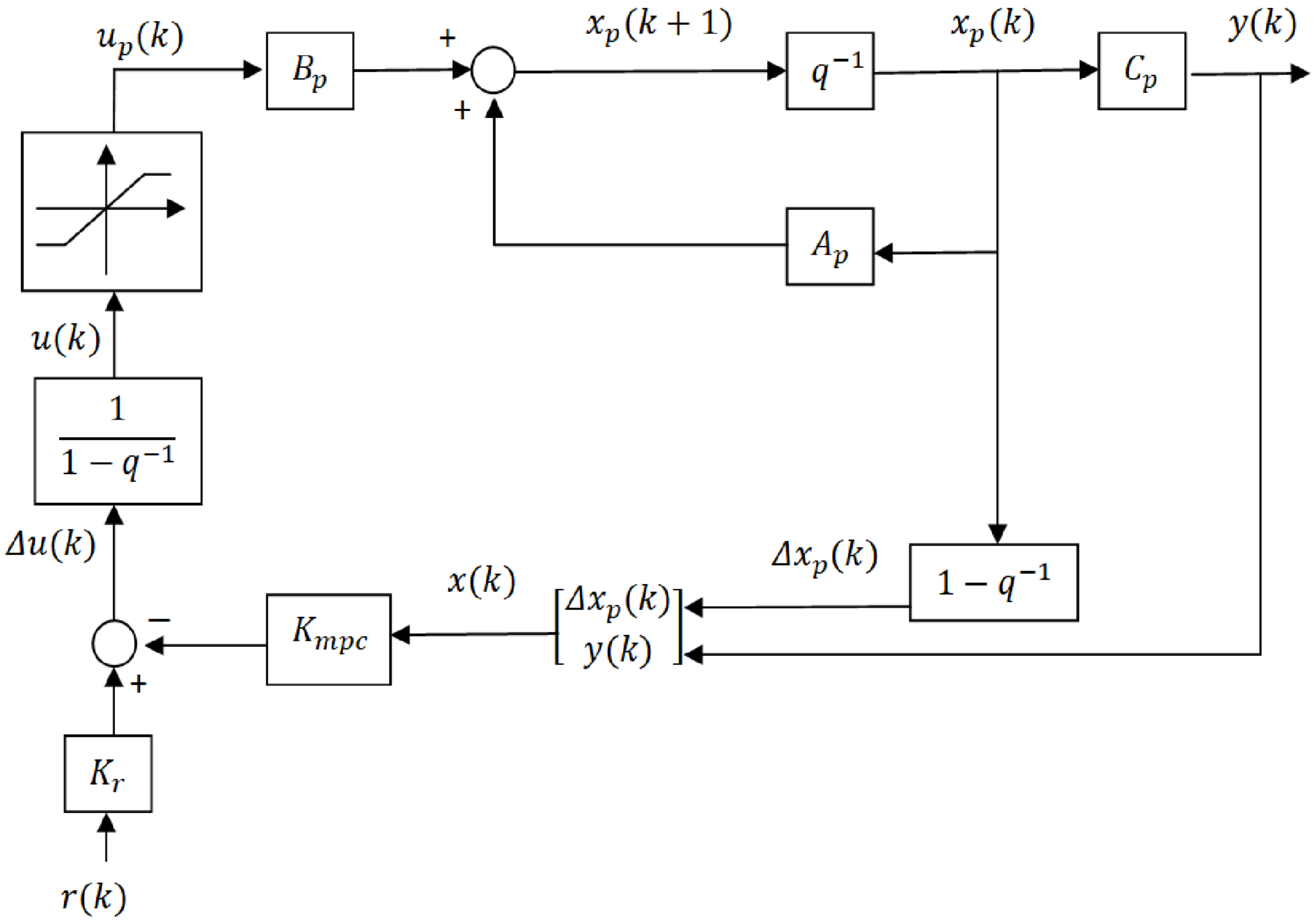

Figure 4.11: Input Saturation

(a) Since there is constraint on $u_p(k)$ but not on $u(k)$, $u_p(k)$ need not be equal to $u(k)$. Since the plant and MPC are the same as Example 8, R_s, F, Φ and $\bar{R}$ are given in Example 8.

Equations (2.30) and (2.31) give

$$\begin{aligned} K_r &= 0.3 \\ K_{mpc} &= \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \end{aligned}$$

Equation (2.29) gives

$$\Delta u(k) = 0.3r(k) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} x(k)$$

<u>$k = 0$</u>

$$\begin{aligned} x_2(0) &= y(0) = x_p(0) = 0 \\ x_1(0) &= x_p(0) - x_p(-1) = 0 - 0 = 0 \\ \Delta u(0) &= K_r r(0) - K_{mpc} x(0) = (0.3)(1) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix}^T = 0.3 \\ u(0) &= u(-1) + \Delta u(0) = 0 + 0.3 = 0.3 \end{aligned}$$

Since $u(0) \geq U^{max}$, $u_p(0) = U^{max} = 0.2$.

$\underline{k=1}$

$$\begin{aligned} x_2(1) &= x_p(1) = y(1) = A_p x_p(0) + B_p u_p(0) = (1)(0) + (1)(0.2) = 0.2 \\ x_1(1) &= x_p(1) - x_p(0) = 0.2 - 0 = 0.2 \\ \Delta u(1) &= K_r r(1) - K_{mpc} x(1) = (0.3)(1) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \begin{bmatrix} 0.2 & 0.2 \end{bmatrix}^T = 0.14 \\ u(1) &= u(0) + \Delta u(1) = 0.3 + 0.14 = 0.44 \end{aligned}$$

Since $u(1) \geq U^{max}$, $u_p(1) = U^{max} = 0.2$.

$\underline{k=2}$

$$\begin{aligned} x_2(2) &= x_p(2) = y(2) = A_p x_p(1) + B_p u_p(1) = (1)(0.2) + (1)(0.2) = 0.4 \\ x_1(2) &= x_p(2) - x_p(1) = 0.4 - 0.2 = 0.2 \\ \Delta u(2) &= K_r r(2) - K_{mpc} x(2) = (0.3)(1) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \begin{bmatrix} 0.2 & 0.4 \end{bmatrix}^T = 0.08 \\ u(2) &= u(1) + \Delta u(2) = 0.44 + 0.08 = 0.52 \end{aligned}$$

Since $u(2) \geq U^{max}$, $u_p(2) = U^{max} = 0.2$.

The rest of the results are given in Table 15 and Figure 4.12.

Table 15

k	$x_p(k) = y(k) = x_2(k)$	$x_1(k)$	$\Delta u(k)$	$u(k)$	$u_p(k)$
0	0	0	0.3	0.3	0.2
1	0.2	0.2	0.14	0.44	0.2
2	0.4	0.2	0.08	0.52	0.2
3	0.6	0.2	0.02	0.54	0.2
4	0.8	0.2	−0.04	0.5	0.2
5	1	0.2	−0.1	0.4	0.2
6	1.2	0.2	−0.16	0.24	0.2
7	1.4	0.2	−0.22	0.02	0.02
8	1.42	0.02	−0.136	−0.116	−0.116
9	1.304	−0.116	−0.0332	−0.1492	−0.1492

(b) Equations (3.15), (3.16), (3.17) and (3.18) give

$$\begin{aligned} H &= 10 \\ f &= \begin{bmatrix} 5x_1(k) + 3x_2(k) - 3 \end{bmatrix} \\ M &= 1 \\ \gamma &= U^{max} - u(k-1) \end{aligned}$$

The quadratic programming problem (3.13) with constraint (3.14) can be formulated as minimizing

$$\begin{aligned} J &= \frac{1}{2}\Delta U^T H \Delta U + \Delta U^T f \\ M\Delta U &\leq \gamma \end{aligned}$$

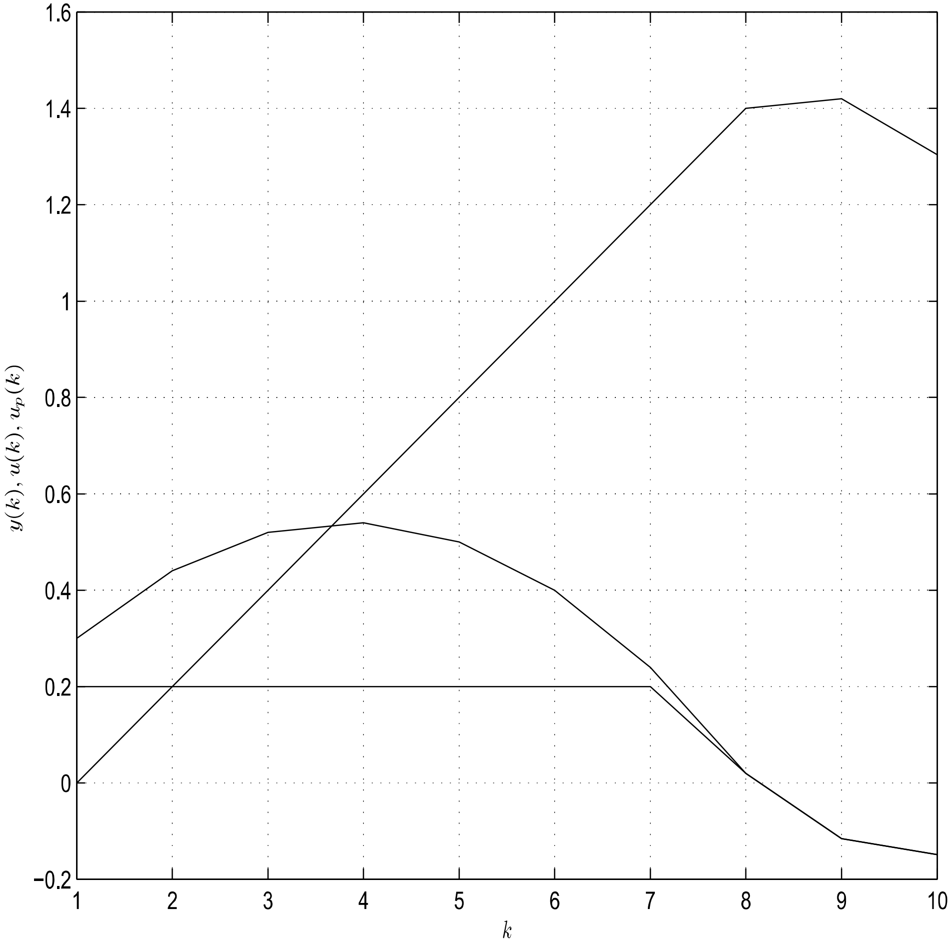

Figure 4.12

Substituting for H, f, M, γ and ΔU gives

$$J = 5\Delta u(k)^2 + (5x_1(k) + 3x_2(k) - 3)\Delta u(k)$$

with constraints

$$\Delta u(k) \ \leq \ U^{max} - u(k-1) \tag{4.26}$$

(c) Since the constraint for $u_p(k) = u(k) = U^{max}$, $u_p(k) = u(k)$.

<u>$k = 0$</u>

$$\begin{aligned} x_2(0) &= y(0) = x_p(0) = 0 \\ x_1(0) &= x_p(0) - x_p(-1) = 0 - 0 = 0 \\ \Delta u(0) &= K_r r(0) - K_{mpc} x(0) = (0.3)(1) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix}^T = 0.3 \end{aligned}$$

Constraint (4.26) gives $\Delta u(0) \leq U^{max} - u(-1) = 0.2 - 0 = 0.2$. The minimum J is obtained from the minimum of $\Delta u(0) = 0.3$, 0.2. Hence $\Delta u(0) = 0.2$, $u(0) = \Delta u(0) + u(-1) = 0.2 + 0 = 0.2$.

Table 16

k	$x_1(k)$ (3.19)	$x_2(k) = y(k) = x_p(k)$ (3.19)	Constraint (4.26)	(3.22)	$\Delta u(k)$	$u(k) = u_p(k)$
0	0	0	≤ 0.2	0.3	0.2	0.2
1	0.2	0.2	≤ 0	0.14	0	0.2
2	0.2	0.4	≤ 0	0.08	0	0.2
3	0.2	0.6	≤ 0	0.02	0	0.2
4	0.2	0.8	≤ 0	-0.04	-0.04	0.16
5	0.16	0.96	≤ 0.04	-0.068	-0.068	0.092
6	0.092	1.052	≤ 0.108	-0.0616	-0.0616	0.0304
7	0.0304	1.0824	≤ 0.1696	-0.0399	-0.0399	-0.0095
8	-0.0095	1.0729	≤ 0.2095	-0.0171	-0.0171	-0.0266
9	-0.026	1.0463	≤ 0.2266	-0.0006	-0.0006	-0.0272

$\underline{k = 1}$

$$
\begin{aligned}
x(1) &= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} (0.2) = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix} \\
\Delta u(1) &= K_r r(1) - K_{mpc} x(1) = (0.3)(1) - \begin{bmatrix} 0.5 & 0.3 \end{bmatrix} \begin{bmatrix} 0.2 & 0.2 \end{bmatrix}^T = 0.14
\end{aligned}
$$

Constraint (4.26) gives $\Delta u(1) \leq U^{max} - u(0) = 0.2 - 0.2 = 0$. The minimum J is obtained from the minimum of $\Delta u(1) = 0.14, 0$. Hence $\Delta u(1) = 0$, $u(1) = \Delta u(1) + u(0) = 0 + 0.2 = 0.2$. The rest of the results are given in the Table 16 and Figure 4.13.

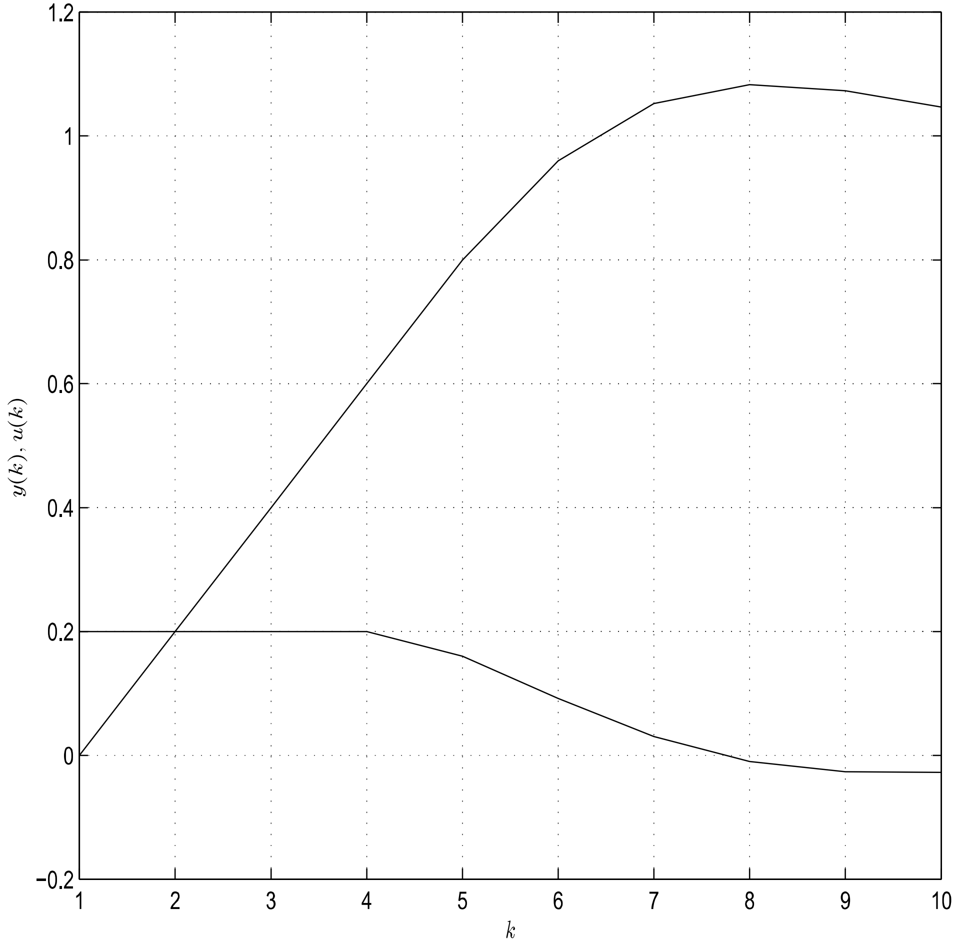

Figure 4.13

(d) Equations (2.30) and (2.31) give

$$\begin{aligned} K_r &= 0.0667 \\ K_{mpc} &= \begin{bmatrix} 0.1111 & 0.0667 \end{bmatrix} \end{aligned}$$

Equation (2.29) gives

$$\Delta u(k) = 0.0667 r(k) - \begin{bmatrix} 0.1111 & 0.0667 \end{bmatrix} x(k)$$

<u>$k = 0$</u>

$$\begin{aligned} \Delta u(0) &= K_r r(0) - K_{mpc} x(0) = (0.0667)(1) - \begin{bmatrix} 0.1111 & 0.0667 \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix}^T = 0.0667 \\ u(0) &= u(-1) + \Delta u(0) = 0 + 0.0667 = 0.0667 \end{aligned}$$

<u>$k = 1$</u>

$$\begin{aligned} x(1) &= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} (0.0667) = \begin{bmatrix} 0.0667 \\ 0.0667 \end{bmatrix} \\ \Delta u(1) &= K_r r(1) - K_{mpc} x(1) = (0.0667)(1) - \begin{bmatrix} 0.1111 & 0.0667 \end{bmatrix} \begin{bmatrix} 0.0667 & 0.0667 \end{bmatrix}^T = 0.0548 \end{aligned}$$

<u>$k = 2$</u>

$$
\begin{aligned}
x(2) &= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} (0.0548) = \begin{bmatrix} 0.1215 \\ 0.1881 \end{bmatrix} \\
\Delta u(1) &= K_r r(1) - K_{mpc} x(1) = (0.0667)(1) - \begin{bmatrix} 0.1111 & 0.0667 \end{bmatrix} \begin{bmatrix} 0.1215 & 0.1881 \end{bmatrix}^T = 0.0406
\end{aligned}
$$

The rest of the results are given in Table 17 and Figure 4.14.

Table 17

k	$x_p(k) = y(k) = x_2(k)$	$x_1(k)$	$\Delta u(k)$	$u(k) = u_p(k)$
0	0	0	0.0667	0.0667
1	0.0667	0.0667	0.0548	0.1215
2	0.1881	0.1215	0.0406	0.1621
3	0.3503	0.1621	0.0253	0.1874
4	0.5377	0.1874	0.0100	0.1974
5	0.7351	0.1974	−0.0043	0.1931
6	0.9282	0.1931	−0.0167	0.1765
7	1.1047	0.1765	−0.0266	0.1499
8	1.2546	0.1499	−0.0336	0.1163
9	1.3708	0.1163	−0.0376	0.0786

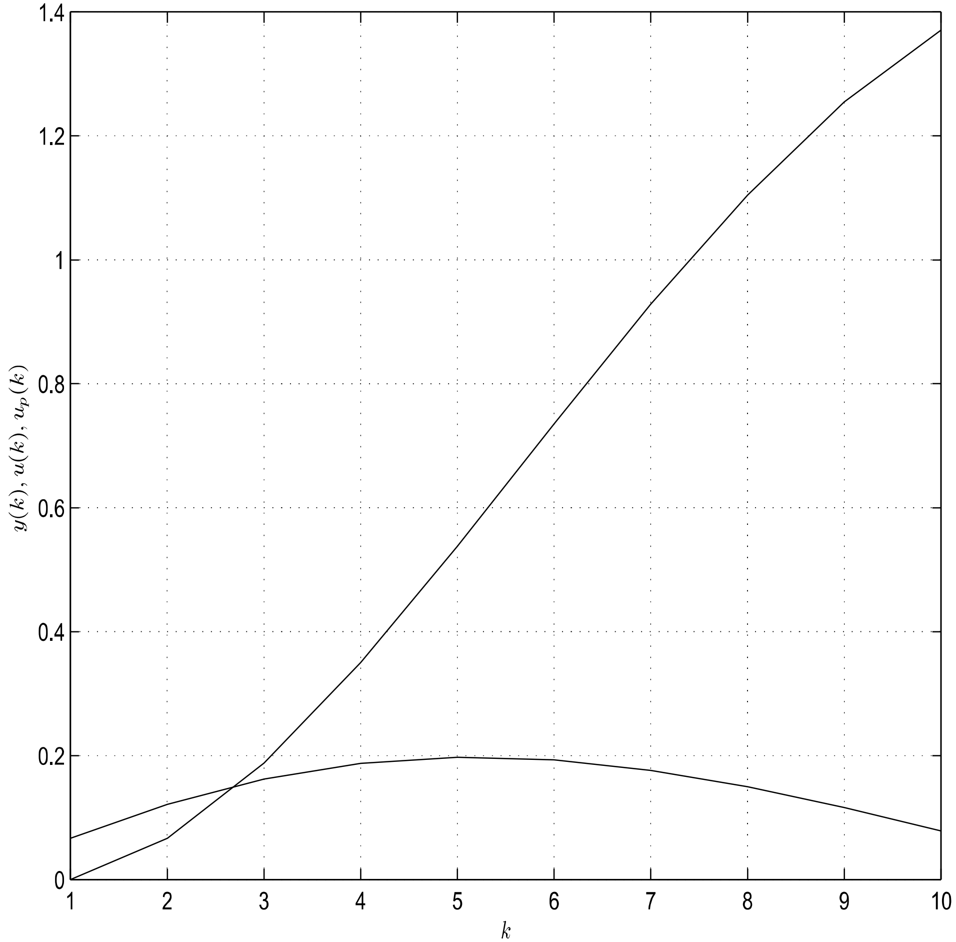

Figure 4.14

Note that $y(k)$ converges at $y(6) = 1.052$ in Table 16 unlike Tables 15 and 17. In Table 16 the MPC optimization took the constraints into consideration unlike Tables 15 and 17.

Chapter 5

Project

Part 1: The true state position $x(k)$ of a target is given by the following equations

$$x(k+1) = x(k) + w(k) \tag{5.1}$$

$$y(k) = x(k) + v(k) \tag{5.2}$$

where $w(k)$ and $v(k)$ are zero-mean independent Gaussian random variables with standard deviations $\sigma_w = 0.1$ and $\sigma_v = 1$ respectively. The initial condition is given by $x(0) = 5$. Design a Kalman filter using the model of (5.1) and (5.2). Simulate for $k = 0, 1, \ldots, N$ where $N = 10{,}000$. Plot the following variables against k.

Graph 1: $x(k)$ (solid-line), $\hat{x}(k|k)$ (dotted-line)

Graph 2: $P(k|k)$
Graph 3: $K_f(k)$

Calculate the following
Bias: $\frac{1}{N+1}\sum_{k=0}^{N}(x(k)-\hat{x}(k|k))$
Variance: $\frac{1}{N+1}\sum_{k=0}^{N}(x(k)-\hat{x}(k|k))^2$.

<u>Part 2</u>: The true state position $x_1(k)$ and velocity $x_2(k)$ of a moving target are given by the following equations

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix} w(k) \tag{5.3}$$

$$y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + v(k) \tag{5.4}$$

where $w(k)$ and $v(k)$ are zero-mean independent Gaussian random variables with standard deviations $\sigma_w = 0.1$ and $\sigma_v = 1$ respectively. The sampling period is $T = 1$ and the initial conditions are $x_1(0) = 0$ and $x_2(0) = 30$. Design a Kalman filter using the model of (5.3) and (5.4). Simulate for $k = 0, 1, \ldots, N$ where $N = 10{,}000$. Plot the following variables against k.
Graph 4: $x_1(k)$ (solid-line), $\hat{x}_1(k|k)$ (dotted-line)
Graph 5: $x_2(k)$ (solid-line), $\hat{x}_2(k|k)$ (dotted-line)

Graph 6: $P_{11}(k|k)$
Graph 7: $P_{22}(k|k)$
Graph 8: $K_{f1}(k)$
Graph 9: $K_{f2}(k)$

Calculate the biases
$\frac{1}{N+1}\sum_{k=0}^{N}\left(x_1(k)-\hat{x}_1(k|k)\right)$
$\frac{1}{N+1}\sum_{k=0}^{N}\left(x_2(k)-\hat{x}_2(k|k)\right)$
the variances
$\frac{1}{N+1}\sum_{k=0}^{N}\left(x_1(k)-\hat{x}_1(k|k)\right)^2$
$\frac{1}{N+1}\sum_{k=0}^{N}\left(x_2(k)-\hat{x}_2(k|k)\right)^2$

Part 3: Indoor positioning systems have proven to be useful in applications such as indoor navigation, equipment tracking and inventory management. For instance, indoor tracking systems have been used in hospitals to keep track of expensive equipment and elderly patients. Receivers at known positions act as reference nodes. The target broadcast an unique identifier at a regular interval and the received signal strength at each reciver is trasformed into distance from the target. The distances are then used as weights and the target position is the centroid. However, factors like signal scattering or blockage due to obstacles, multipath interference caused by walls and furniture, and even human movements can significantly affect the received signal strength. Therefore the position obtained is usually highly corrupted by noise and Kalman filter is necessary to reduce the effect of noise.

The true state positions on the yz-plane $\left(10\cos\frac{1}{12}k\pi,\quad 10\sin\frac{1}{12}k\pi\right)$, $k = 0, 1, \ldots, N$ where $N = 24$, of a target moving in a circle is shown in Figure 5.1 as $\circ$ while the noisy $y(k)$ and $z(k)$ measurements are shown as $\bullet$.

The $y(k)$ and $z(k)$ measurements are given as

k	0	1	2	3	4	5	6	7
$y(k)$	7.1165	9.6022	8.9144	9.2717	6.3400	4.0484	0.3411	−0.6784
$z(k)$	0.000	3.1398	6.3739	9.5877	10.1450	10.1919	9.0683	10.2254

k	8	9	10	11	12	13	14	15
$y(k)$	−5.7726	−5.4925	−9.4523	−9.7232	−9.5054	−9.7908	−7.7300	−5.9779
$z(k)$	7.5799	7.7231	5.4721	3.3990	0.9172	−1.3551	−5.2708	−9.7011

k	16	17	18	19	20	21	22	23	24
$y(k)$	−4.5535	−1.5042	−0.7044	3.2406	8.3029	6.1925	9.1178	9.0904	9.0662
$z(k)$	−9.4256	−9.3053	−9.3815	−9.8822	−8.1876	−8.7501	−4.5653	−1.9179	−1.0000

Since it is a 2 dimensional problem, design a system of Kalman filters. The inputs to the system are the y and z measurements and the outputs, $\hat{x}_y$ and $\hat{x}_z$ estimates. The variances of the y and z measurement noises are 1. Give the models you used for the Kalman filters. Copy Figure 5.1 into your report and superimpose on it the estimates $(\hat{x}_y(k|k),\ \hat{x}_z(k|k))$ using $\times$. Join the $\times$ using a solid-line.

Give $K_f(k)$, $K(k)$, $P(k|k)$ and $P(k+1|k)$ for the Kalman filter.

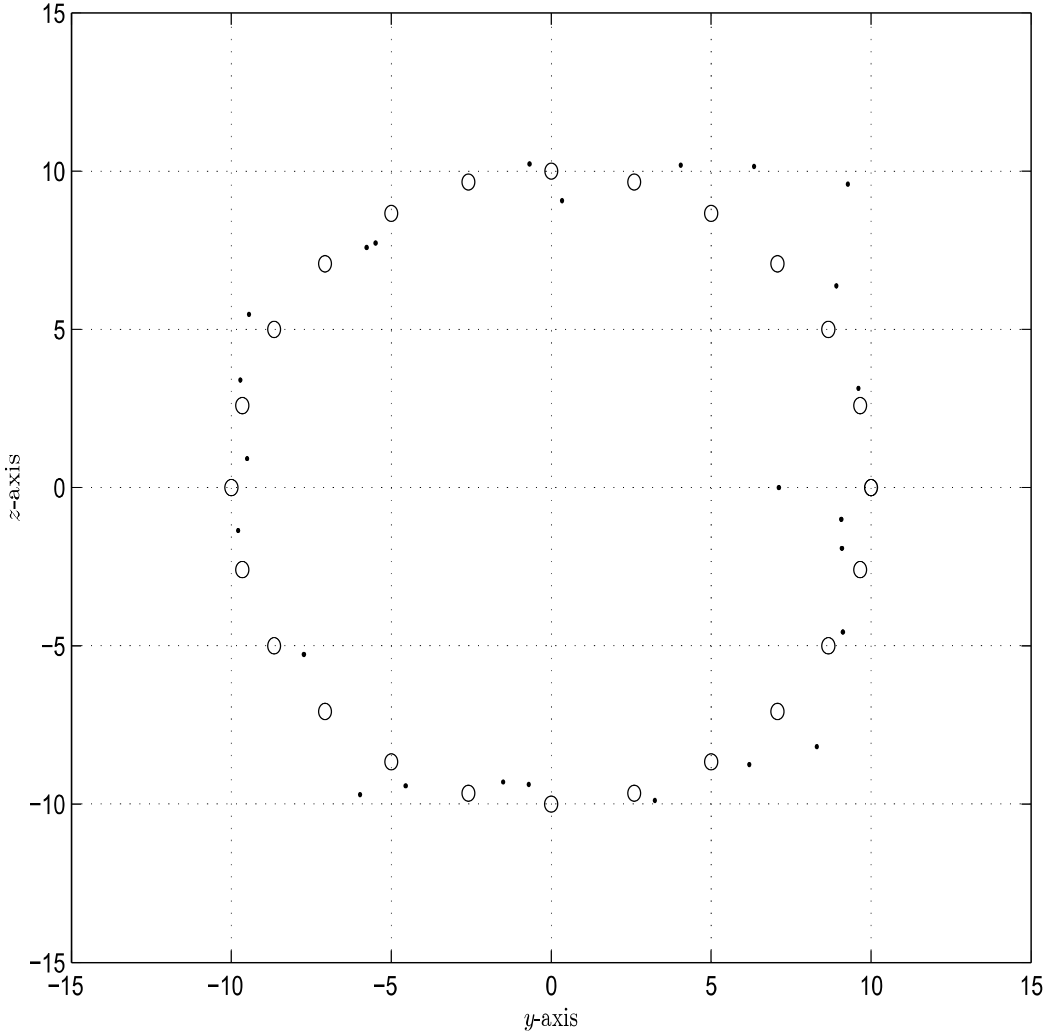

Figure 5.1: Target moving in a circle of radius 10 and angular velocity of $\frac{1}{12}\pi$ rad/s.
true state: ∘; measurement: •

Calculate the biases
$\frac{1}{N+1}\sum_{k=0}^{N}\left\{10\cos\frac{2\pi k}{N} - \hat{x}_y(k|k)\right\}$,
$\frac{1}{N+1}\sum_{k=0}^{N}\left\{10\sin\frac{2\pi k}{N} - \hat{x}_z(k|k)\right\}$
and variances
$\frac{1}{N+1}\sum_{k=0}^{N}\left\{10\cos\frac{2\pi k}{N} - \hat{x}_y(k|k)\right\}^2$,
$\frac{1}{N+1}\sum_{k=0}^{N}\left\{10\sin\frac{2\pi k}{N} - \hat{x}_z(k|k)\right\}^2$
where $\hat{x}_y(k|k)$ and $\hat{x}_z(k|k)$ are the Kalman filter estimates of the y- and z- axis true state positions $10\cos\frac{2\pi k}{N}$ and $10\sin\frac{2\pi k}{N}$ respectively.

For the 3 parts where possible give comments and provide equations and analytical interpretations of your results.