

Model Predictive Control

In this chapter we consider *model predictive control (MPC)*, an important advanced control technique for difficult multivariable control problems. The basic MPC concept can be summarized as follows. Suppose that we wish to control a multiple-input, multiple-output process while satisfying inequality constraints on the input and output variables. If a reasonably accurate dynamic model of the process is available, model and current measurements can be used to predict future values of the outputs. Then the appropriate changes in the input variables can be calculated based on both predictions and measurements. In essence, the changes in the individual input variables are coordinated after considering the input-output relationships represented by the process model. In MPC applications, the output variables are also referred to as *controlled variables* or *CVs*, while the input variables are also called *manipulated variables* or *MVs*. Measured disturbance variables are called *DVs* or *feedforward variables*. These terms will be used interchangeably in this chapter.

Model predictive control offers several important advantages: (1) the process model captures the dynamic and static interactions between input, output, and disturbance variables, (2) constraints on inputs and outputs are considered in a systematic manner, (3) the control calculations can be coordinated with the calculation of optimum set points, and (4) accurate model predictions can provide early warnings of potential problems. Clearly, the success of MPC (or any other model-based approach) depends on the accuracy of the process model. Inaccurate predictions can make matters worse, instead of better.

First-generation MPC systems were developed independently in the 1970s by two pioneering industrial research groups. Dynamic Matrix Control (*DMC*), devised by Shell Oil (Cutler and Ramaker, 1980), and a related approach developed by ADERSA

(Richalet et al., 1978) have quite similar capabilities. An adaptive MPC technique, Generalized Predictive Control (GPC), developed by Clarke et al. (1987) has also received considerable attention. Model predictive control has had a major impact on industrial practice. For example, an MPC survey by Qin and Badgwell (2003) reported that there were over 4,500 applications worldwide by the end of 1999, primarily in oil refineries and petrochemical plants. In these industries, MPC has become the method of choice for difficult multivariable control problems that include inequality constraints.

In view of its remarkable success, MPC has been a popular subject for academic and industrial research. Major extensions of the early MPC methodology have been developed, and theoretical analysis has provided insight into the strengths and weaknesses of MPC. Informative reviews of MPC theory and practice are available in books (Camacho and Bordons, 2003; Maciejowski, 2002; Rossiter, 2003; Richalet and O'Donovan, 2009); tutorials (Hokanson and Gerstle, 1992; Rawlings, 2000), and survey papers (Morari and Lee, 1999; Qin and Badgwell, 2003; Canney, 2003; Kano and Ogawa, 2009).

20.1 OVERVIEW OF MODEL PREDICTIVE CONTROL

The overall objectives of an MPC controller have been summarized by Qin and Badgwell (2003):

1. Prevent violations of input and output constraints.
2. Drive some output variables to their optimal set points, while maintaining other outputs within specified ranges (see Section 20.4.2).
3. Prevent excessive movement of the input variables.
4. Control as many process variables as possible when a sensor or actuator is not available.

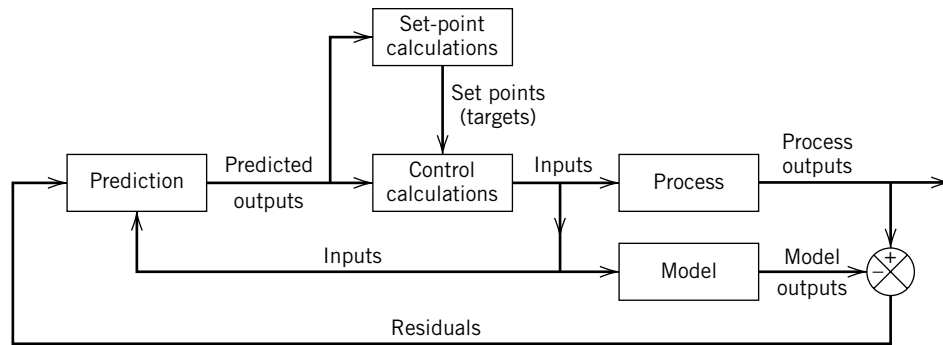


Figure 20.1 Block diagram for model predictive control.

A block diagram of a model predictive control system is shown in Fig. 20.1. A process model is used to predict the current values of the output variables. The residuals, the differences between the actual and predicted outputs, serve as the feedback signal to a *Prediction* block. The predictions are used in two types of MPC calculations that are performed at each sampling instant: set-point calculations and control calculations. Inequality constraints on the input and output variables, such as upper and lower limits, can be included in either type of calculation. Note that the MPC configuration is similar to both the internal model control configuration in Chapter 11 and the Smith predictor configuration of Chapter 15, because the model acts in parallel with the process and the residual serves as a feedback signal. However, the coordination of the control and set-point calculations is a unique feature of MPC. Furthermore, MPC has had a much greater impact on industrial practice than IMC or Smith predictor, because it is more suitable for constrained MIMO control problems.

The set points for the control calculations, also called *targets*, are calculated from an economic optimization

based on a steady-state model of the process, traditionally, a linear model. Typical optimization objectives include maximizing a profit function, minimizing a cost function, or maximizing a production rate. The optimum values of set points change frequently due to varying process conditions, especially changes in the inequality constraints (see Chapter 19). The constraint changes are due to variations in process conditions, equipment, and instrumentation, as well as economic data such as prices and costs. In MPC the set points are typically calculated each time the control calculations are performed, as discussed in Section 20.5.

The MPC calculations are based on current measurements and predictions of the future values of the outputs. The objective of the MPC control calculations is to determine a sequence of *control moves* (that is, manipulated input changes) so that the predicted response moves to the set point in an optimal manner. The actual output y , predicted output \hat{y} , and manipulated input u for SISO control are shown in Fig. 20.2. At the current sampling instant, denoted by k , the MPC strategy calculates a set of M values of the input $\{u(k+i-1)$,

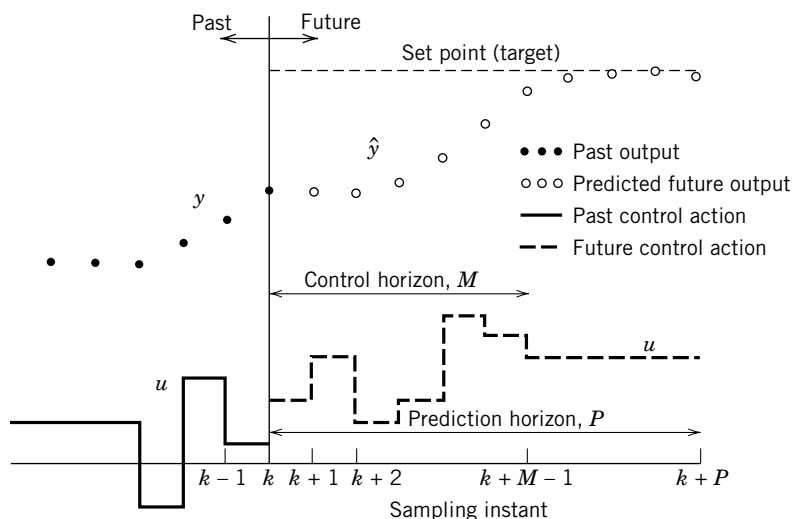


Figure 20.2 Basic concept for model predictive control.

$i = 1, 2, \dots, M$ }. The set consists of the current input $u(k)$ and $M - 1$ future inputs. The input is held constant after the M control moves. The inputs are calculated so that a set of P predicted outputs $\hat{y}(k + i)$, $i = 1, 2, \dots, P$ reaches the set point in an optimal manner. The control calculations are based on optimizing an objective function (cf. Section 20.4). The number of predictions P is referred to as the *prediction horizon* while the number of control moves M is called the *control horizon*.

A distinguishing feature of MPC is its *receding horizon approach*. Although a sequence of M control moves is calculated at each sampling instant, only the first move is actually implemented. Then a new sequence is calculated at the next sampling instant, after new measurements become available; again only the first input move is implemented. This procedure is repeated at each sampling instant. But why is an M -step control strategy calculated if only the first step is implemented? We will answer this question in Section 20.4.

20.2 PREDICTIONS FOR SISO MODELS

The MPC predictions are made using a dynamic model, typically a linear empirical model such as a multivariable version of the step response or difference equation models that were introduced in Chapter 6. Alternatively, transfer function or state-space models (Section 5.5) can be employed. For very nonlinear processes, it can be advantageous to predict future output values using a nonlinear dynamic model. Both physical models and empirical models, such as neural networks (Section 6.3), have been used in nonlinear MPC (Badgwell and Qin, 2001; White, 2008). Step-response models offer the advantage that they can represent stable processes with unusual dynamic behavior that cannot be accurately described by simple transfer function models (cf. Example 6.6). Their main disadvantage is the large number of model parameters. Although step-response models are not suitable for unstable processes, they can be modified to represent integrating processes, as shown in Section 20.2.2.

Next, we demonstrate how step-response models can be used to predict future outputs. Similar predictions can be made using other types of linear models such as transfer function or state-space models.

The step-response model of a stable, single-input, single-output process can be written as

$$y(k + 1) = y_0 + \sum_{i=1}^{N-1} S_i \Delta u(k - i + 1) + S_N u(k - N + 1) \quad (20-1)$$

where $y(k + 1)$ is the output variable at the $(k + 1)$ -sampling instant, and $\Delta u(k - i + 1)$ denotes the change in the manipulated input from one sampling instant to the next, $\Delta u(k - i + 1) = u(k - i + 1) - u(k - i)$. Both y and u are deviation variables. The model parameters are the N step-response coefficients, S_1 to S_N . Typically, N is selected so that $30 \leq N \leq 120$. The initial value, $y(0)$, is denoted by y_0 . For simplicity, we will assume that $y_0 = 0$.

In Section 6.5 we showed that step-response models can be obtained empirically from experimental data. Example 20.1 illustrates that they can also be derived analytically from transfer function models.

EXAMPLE 20.1

Consider a first-order-plus-time-delay model:

$$\frac{Y(s)}{U(s)} = \frac{K e^{-\theta s}}{\tau s + 1} \quad (20-2)$$

- Derive the equivalent step-response model by considering the analytical solution to a unit step change in the input.
- Calculate the step-response coefficients, $\{S_i\}$, for the following parameter values: $K = 5$, $\tau = 15$ min, $\theta = 3$ min, and a sampling period of $\Delta t = 1$ min. Also, calculate and plot the response $y(k)$ for $0 \leq k \leq 80$ after a step change in u from 0 to 3 occurs at $t = 2$ min.

SOLUTION

- The step response for a first-order model without a time delay ($\theta = 0$) was derived in Chapter 4

$$y(t) = KM(1 - e^{-t/\tau}) \quad (4-18)$$

where M is the magnitude of the step change. The corresponding response for the model with a time delay is

$$y(t) = \begin{cases} 0 & \text{for } t \leq \theta \\ KM(1 - e^{-(t-\theta)/\tau}) & \text{for } t > \theta \end{cases} \quad (20-3)$$

The sampling instants are denoted by $t = i\Delta t$ where Δt is the sampling period and $i = 1, 2, \dots$. Substituting $t = i\Delta t$ into (20-3) gives the response for $0 \leq i \leq 80$:

$$y(i\Delta t) = \begin{cases} 0 & \text{for } i\Delta t \leq \theta \\ KM(1 - e^{-(i\Delta t - \theta)/\tau}) & \text{for } i\Delta t > \theta \end{cases} \quad (20-4)$$

where $i = 1, 2, \dots, 80$

The number of step-response coefficients, N in (20-1), is specified to be $N = 80$ so that $N\Delta t$ is slightly larger than the process settling time of approximately $5\tau + \theta$. As indicated in Section 6.5, the i th step-response coefficient is the value of the unit step response at the i th sampling instant. Thus, the step-response coefficients can be determined from (20-4) after setting $M = 1$:

$$S_i = \begin{cases} 0 & \text{for } i\Delta t \leq \theta \\ K(1 - e^{-(i\Delta t - \theta)/\tau}) & \text{for } i\Delta t > \theta \end{cases} \quad (20-5)$$

where $i = 1, 2, \dots, 80$

- (b) Substituting numerical values into (20-5) gives the step-response coefficients in Table 20.1. The step response $y(k)$ in Fig. 20.3 can be calculated either from (20-1) and (20-5), or from (20-4). For $\theta = 2$ min, $S_1 = S_2 = 0$, and the step response is zero until $t > 2$ min. The new

Table 20.1 Step-Response Coefficients for Example 20.1

Sampling Instant	S_i	Sampling Instant	S_i	Sampling Instant	S_i
1	0	28	4.06	55	4.84
2	0	29	4.12	56	4.85
3	0	30	4.17	57	4.86
4	0.32	31	4.23	58	4.87
5	0.62	32	4.28	59	4.88
6	0.91	33	4.32	60	4.89
7	1.17	34	4.37	61	4.90
8	1.42	35	4.41	62	4.90
9	1.65	36	4.45	63	4.91
10	1.86	37	4.48	64	4.91
11	2.07	38	4.52	65	4.92
12	2.26	39	4.55	66	4.93
13	2.43	40	4.58	67	4.93
14	2.60	41	4.60	68	4.93
15	2.75	42	4.63	69	4.94
16	2.90	43	4.65	70	4.94
17	3.03	44	4.68	71	4.95
18	3.16	45	4.70	72	4.95
19	3.28	46	4.72	73	4.95
20	3.39	47	4.73	74	4.96
21	3.49	48	4.75	75	4.96
22	3.59	49	4.77	76	4.96
23	3.68	50	4.78	77	4.96
24	3.77	51	4.80	78	4.97
25	3.85	52	4.81	79	4.97
26	3.92	53	4.82	80	4.97
27	3.99	54	4.83		

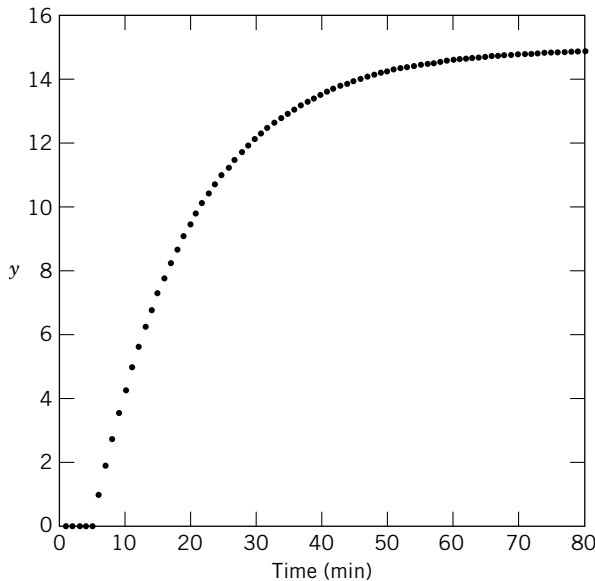


Figure 20.3 Step response for Example 20.1.

steady-state value is $y = 15$ because the steady-state gain in (20-2) is $K = 5$ and the magnitude of the step change is $M = 3$. Because $\Delta t = 1$ min and the step change occurs at $t = 3$ min, $\Delta u(k) = M = 3$ for $k = 3$ and $\Delta u(k) = 0$ for all other values of k . Recall that $\Delta u(k)$ is defined as $\Delta u(k) \triangleq u(k) - u(k-1)$.

For this example, the response $y(k)$ could be calculated analytically from (20-4) because a transfer function model was assumed. However, in many MPC applications, the transfer function model is not known, and thus the response must be calculated from the step-response model in (20-1).

Model predictive control is based on predictions of future outputs over a prediction horizon, P . We now consider the calculation of these predictions. Let k denote the current sampling instant and $\hat{y}(k+1)$ denote the prediction of $y(k+1)$ that is made at time k . If $y_0 = 0$, this one-step-ahead prediction can be obtained from Eq. (20-1) by replacing $y(k+1)$ with $\hat{y}(k+1)$:

$$\hat{y}(k+1) = \sum_{i=1}^{N-1} S_i \Delta u(k-i+1) + S_N u(k-N+1) \quad (20-6)$$

Equation 20-6 can be expanded as

$$\hat{y}(k+1) = \underbrace{S_1 \Delta u(k)}_{\text{Effect of current control action}} + \underbrace{\sum_{i=2}^{N-1} S_i \Delta u(k-i+1)}_{\text{Effect of past control actions}} + S_N u(k-N+1) \quad (20-7)$$

The first term on the right-hand side indicates the effect of the current manipulated input $u(k)$ because $\Delta u(k) = u(k) - u(k-1)$. The second term represents the effects of past inputs, $\{u(i), i < k\}$. An analogous expression for a two-step-ahead prediction can be derived in a similar manner. Substitute $k = k' + 1$ into Eq. 20-6:

$$\hat{y}(k'+2) = \sum_{i=1}^{N-1} S_i \Delta u(k'-i+2) + S_N u(k'-N+2) \quad (20-8)$$

Because Eq. 20-8 is valid for all positive values of k' , without loss of generality, we can replace k' with k and then expand the right-hand side to identify the contributions relative to the current sampling instant, k :

$$\begin{aligned} \hat{y}(k+2) &= \underbrace{S_1 \Delta u(k+1)}_{\text{Effect of future control action}} + \underbrace{S_2 \Delta u(k)}_{\text{Effect of current control action}} \\ &+ \underbrace{\sum_{i=3}^{N-1} S_i \Delta u(k-i+2)}_{\text{Effect of past control actions}} + S_N u(k-N+2) \quad (20-9) \end{aligned}$$

An analogous derivation provides an expression for a *j*-step-ahead prediction where *j* is an arbitrary positive integer:

$$\hat{y}(k + j) = \underbrace{\sum_{i=1}^j S_i \Delta u(k + j - i)}_{\text{Effect of current and future control actions}} + \underbrace{\sum_{i=j+1}^{N-1} S_i \Delta u(k + j - i) + S_N u(k + j - N)}_{\text{Effect of past control actions}} \quad (20-10)$$

The second and third terms on the right-hand side of Eq. 20-10 represent the predicted response when there are no current or future control actions, that is, the predicted response when $u(k + i) = u(k - 1)$ for $i \geq 0$, or equivalently, $\Delta u(k + i) = 0$ for $i \geq 0$. Because this term accounts for past control actions, it is referred to as the *predicted unforced response* and is denoted by the symbol, $\hat{y}^o(k + j)$. Thus, we define $\hat{y}^o(k + j)$ as

$$\hat{y}^o(k + j) \triangleq \sum_{i=j+1}^{N-1} S_i \Delta u(k + j - i) + S_N u(k + j - N) \quad (20-11)$$

and write Eq. 20-10 as

$$\hat{y}(k + j) = \sum_{i=1}^j S_i \Delta u(k + j - i) + \hat{y}^o(k + j) \quad (20-12)$$

Examples 20.2 and 20.3 demonstrate that Eq. 20-12 can be used to derive a simple predictive control law based on a *single* prediction.

EXAMPLE 20.2

Derive a predictive control law that is based on the following concept. A single control move, $\Delta u(k)$, is calculated so that the *J*-step-ahead prediction is equal to the set point, that is, $\hat{y}(k + J) = y_{sp}$ where integer *J* is a tuning parameter. This sampling instant, $k + J$, is referred to as a *coincidence point*. Assume that *u* is held constant after the single control move, so that $\Delta u(k + i) = 0$ for $i > 0$.

SOLUTION

In the proposed predictive control strategy, only a single prediction for *J* steps ahead is considered. Thus, we let $j = J$ in Eq. 20-12. Similarly, because we are only interested in calculating the current control move, $\Delta u(k)$, the future control moves

in Eq. 20-12 are set equal to zero: $\Delta u(k + J - i) = 0$ for $i = 1, 2, \dots, J - 1$. Thus, (20-12) reduces to

$$\hat{y}(k + J) = S_J \Delta u(k) + \hat{y}^o(k + J) \quad (20-13)$$

Setting $\hat{y}(k + J) = y_{sp}$ and rearranging gives the desired predictive controller:

$$\Delta u(k) = \frac{y_{sp} - \hat{y}^o(k + J)}{S_J} \quad (20-14)$$

The predicted unforced response $\hat{y}^o(k + J)$ can be calculated from Eq. 20-11 with $j = J$.

The control law in (20-14) is based on a single prediction that is made for *J* steps in the future. Note that the control law can be interpreted as the inverse of the predictive model in (20-13).

EXAMPLE 20.3

Apply the predictive control law of Example 20.2 to a fifth-order process:

$$\frac{Y(s)}{U(s)} = \frac{1}{(5s + 1)^5} \quad (20-15)$$

Evaluate the effect of tuning parameter *J* on the set-point responses for values of $J = 3, 4, 6,$ and 8 and $\Delta t = 5$ min.

SOLUTION

The *y* and *u* responses for a unit set-point change at $t = 0$ are shown in Figs. 20.4 and 20.5, respectively. As *J* increases,

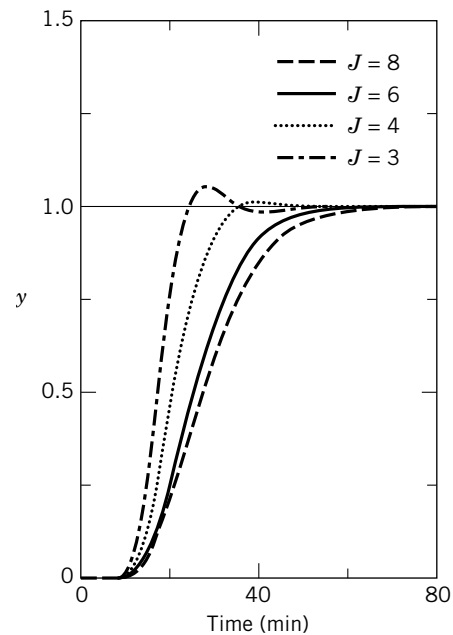


Figure 20.4 Set-point responses for Example 20.3 and different values of *J*.

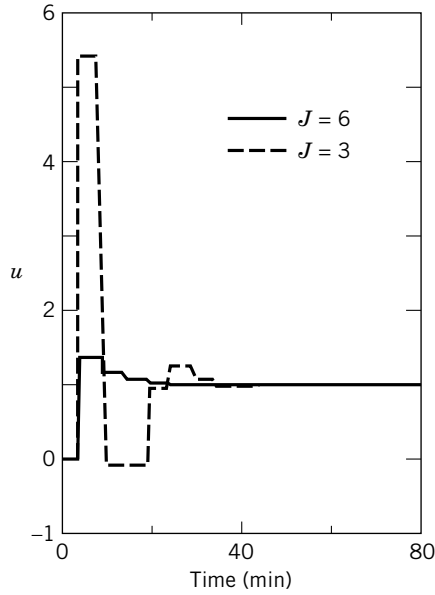


Figure 20.5 Input responses for Fig. 20.4.

the y responses become more sluggish while the u responses become smoother. These trends occur because larger values of J allow the predictive controller more time before the J -step ahead prediction $\hat{y}(k + J)$ must equal the set point. Consequently, less strenuous control action is required. The J th step-response coefficient S_J increases monotonically as J increases. Consequently, the input moves calculated from (20-14) tend to become smaller as S_J increases. (The u responses for $J = 4$ and 8 are omitted from Fig. 20.5.)

The previous two examples have considered a simple predictive controller based on single prediction made J steps ahead. Now, we consider the more typical situation in which the MPC calculations are based on multiple predictions rather than on a single prediction. The notation is greatly simplified if vector-matrix notation is employed. Consequently, we define a vector of predicted responses for the next P sample instants as

$$\hat{\mathbf{Y}}(k + 1) \triangleq \text{col}[\hat{y}(k + 1), \hat{y}(k + 2), \dots, \hat{y}(k + P)] \quad (20-16)$$

where col denotes a column vector. Similarly, a vector of predicted unforced responses from Eq. 20-11 is defined as

$$\hat{\mathbf{Y}}^o(k + 1) \triangleq \text{col}[\hat{y}^o(k + 1), \hat{y}^o(k + 2), \dots, \hat{y}^o(k + P)] \quad (20-17)$$

Define $\Delta \mathbf{U}(k)$ to be a vector of control actions for the next M sampling instants:

$$\Delta \mathbf{U}(k) \triangleq \text{col}[\Delta u(k), \Delta u(k + 1), \dots, \Delta u(k + M - 1)] \quad (20-18)$$

The control horizon M and prediction horizon P are key design parameters, as discussed in Section 20.6. In general, $M \leq P$ and $P \leq N + M$.

The MPC control calculations are based on calculating $\Delta \mathbf{U}(k)$ so that the predicted outputs move optimally to the new set points. For the control calculations, the model predictions in Eq. 20-12 are conveniently written in vector-matrix notation as

$$\hat{\mathbf{Y}}(k + 1) = S \Delta \mathbf{U}(k) + \hat{\mathbf{Y}}^o(k + 1) \quad (20-19)$$

where S is the $P \times M$ dynamic matrix:

$$S \triangleq \begin{bmatrix} S_1 & 0 & \cdots & 0 \\ S_2 & S_1 & 0 & \vdots \\ \vdots & \vdots & \ddots & 0 \\ S_M & S_{M-1} & \cdots & S_1 \\ S_{M+1} & S_M & \cdots & S_2 \\ \vdots & \vdots & \ddots & \vdots \\ S_P & S_{P-1} & \cdots & S_{P-M+1} \end{bmatrix} \quad (20-20)$$

Equations 20-19 and 20-20 can be derived from (20-12) and (20-16) to (20-18).

20.2.1 Output Feedback and Bias Correction

The predictions in Eqs. 20-12 and 20-19 do not make use of the latest measurement $y(k)$. Consequently, the cumulative effects of model inaccuracy and unmeasured disturbances can lead to inaccurate predictions. However, prediction accuracy can be improved by utilizing the latest measurement in the predictions. This general strategy is referred to as *output feedback* (Qin and Badgwell, 2003). A typical approach is to add a *bias correction*, $b(k + j)$, to the prediction. The *corrected prediction*, $\tilde{y}(k + j)$, is defined as

$$\tilde{y}(k + j) \triangleq \hat{y}(k + j) + b(k + j) \quad (20-21)$$

We will refer to $\hat{y}(k + j)$ as the *uncorrected prediction*. In practice, the bias correction is often specified to be the difference between the latest measurement $y(k)$ and the corresponding predicted value, $\hat{y}(k)$:

$$b(k + j) = y(k) - \hat{y}(k) \quad (20-22)$$

The difference, $y(k) - \tilde{y}(k)$, is also referred to as a *residual* or an *estimated disturbance*. The block diagram for MPC in Fig. 20.1 includes the bias correction.

In (20-22) $\hat{y}(k)$ is a one-step ahead prediction made at the previous sampling instant, $k - 1$. Using Eq. 20-22 is equivalent to assuming that a process disturbance is added to the output and is constant for $j = 1, 2, \dots, P$. Furthermore, the assumed value of the additive disturbance is the residual, $y(k) - \hat{y}(k)$.

Substituting Eq. 20-22 into 20-21 gives

$$\tilde{y}(k + j) \triangleq \hat{y}(k + j) + [y(k) - \hat{y}(k)] \quad (20-23)$$

In a similar fashion, adding the bias correction to the right side of Eq. 20-19 provides a vector of corrected predictions,

$$\tilde{\mathbf{Y}}(k + 1) = \mathbf{S}\Delta\mathbf{U}(k) + \hat{\mathbf{Y}}^o(k + 1) + [y(k) - \hat{y}(k)]\mathbf{1} \quad (20-24)$$

where $\mathbf{1}$ is a P -dimensional column vector with each element having a value of one. Thus the same correction is made for all P predictions. Vector $\tilde{\mathbf{Y}}(k + 1)$ is defined as

$$\tilde{\mathbf{Y}}(k + 1) \triangleq \text{col}[\tilde{y}(k + 1), \tilde{y}(k + 2), \dots, \tilde{y}(k + P)] \quad (20-25)$$

Incorporating output feedback as a bias correction has been widely applied, but it can result in excessively sluggish responses for certain classes of disturbances. Consequently, other types of output feedback and disturbance estimation methods have been proposed (Maciejowski, 2002; Qin and Badgwell, 2003).

EXAMPLE 20.4

The benefits of using corrected predictions will be illustrated by a simple example, the first-order-plus-time-delay model:

$$\frac{Y(s)}{U(s)} = \frac{5e^{-2s}}{15s + 1} \quad (20-26)$$

Assume that the disturbance transfer function is identical to the process transfer function, $G_d(s) = G_p(s)$. A unit change in u occurs at time $t = 2$ min, and a step disturbance, $d = 0.15$, occurs at $t = 8$ min. The sampling period is $\Delta t = 1$ min.

- (a) Compare the process response $y(k)$ with the predictions that were made 15 steps earlier based on a step-response model with $N = 80$. Consider both the corrected prediction $\tilde{y}(k)$ and the uncorrected prediction $\hat{y}(k)$ over a time period, $0 \leq k \leq 90$.
- (b) Repeat (a) for the situation where the step-response coefficients are calculated using an incorrect model:

$$\frac{Y(s)}{U(s)} = \frac{4e^{-2s}}{20s + 1} \quad (20-27)$$

SOLUTION

The output response to the step changes in u and d can be derived from (20-26) using the analytical techniques developed in Appendix A and Chapter 3. Because $\theta = 2$ min and the step in u begins at $t = 2$ min, $y(t)$ first starts to respond at $t = 5$ min. The disturbance at $t = 8$ min begins to affect y at $t = 11$ min. Thus, the response can be written as

$$\begin{aligned} y(t) &= 0 && \text{for } t \leq 4 \text{ min} \\ y(t) &= 5(1)(1 - e^{-(t-4)/15}) && \text{for } 4 < t \leq 10 \text{ min} \\ y(t) &= 5(1)(1 - e^{-(t-4)/15}) && \text{for } t > 10 \text{ min} \\ &\quad + 5(0.15)(1 - e^{-(t-10)/15}) \end{aligned} \quad (20-28)$$

The 15-step-ahead prediction, $\hat{y}(k + 15)$, can be obtained using Eq. 20-12 with $j = 15$ and $N = 80$. The corrected prediction, $\tilde{y}(k + 15)$, can be calculated from Eqs. 20-21 and 20-22 with $j = 15$. But in order to compare actual and predicted responses, it is more convenient to write these equations in an equivalent form:

$$\tilde{y}(k) \triangleq \hat{y}(k) + b(k) \quad (20-29)$$

$$b(k) = y(k - 15) - \hat{y}(k - 15) \quad (20-30)$$

- (a) The actual and predicted responses are compared in Fig. 20.6. For convenience, the plots are shown as lines rather than as discrete points. After the step change in u at $t = 2$ min (or equivalently, at $k = 4$), the 15-step-ahead predictions are identical to the actual response until the step disturbance begins to affect $y(k)$ starting at $k = 11$. For $k > 10$, $\hat{y}(k) < y(k)$ because $\hat{y}(k)$ does not include the effect of the unknown disturbance. Note that $\tilde{y}(k) = \hat{y}(k)$ for $10 < k < 25$ because $b(k) = 0$ during this period. For $k > 25$, $b(k) \neq 0$ and $\tilde{y}(k)$ converges to $y(k)$. Thus, the corrected prediction $\tilde{y}(k)$ is more accurate than the uncorrected prediction, $\hat{y}(k)$.
- (b) Figure 20.7 compares the actual and predicted responses for the case of the plant-model mismatch in

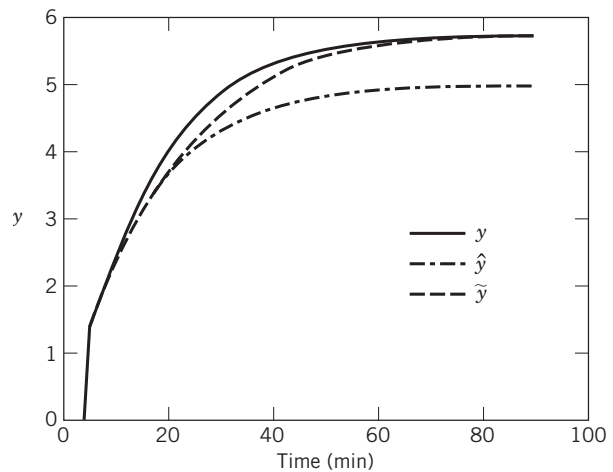


Figure 20.6 Comparison of actual (y), predicted (\hat{y}), and corrected (\tilde{y}) responses when the model is perfect.

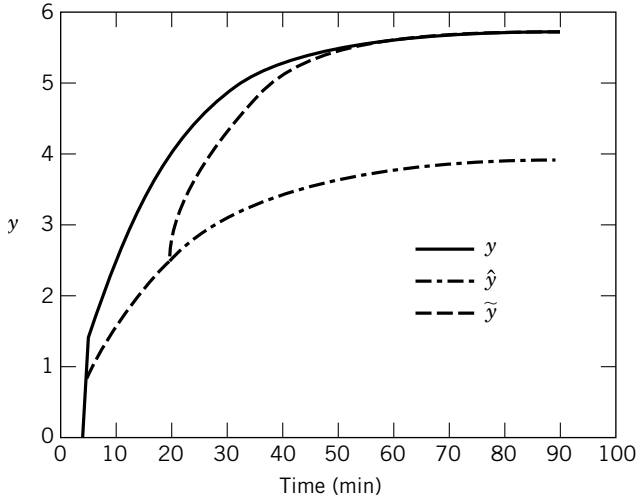


Figure 20.7 Comparison of actual and predicted responses for plant-model mismatch.

Eqs. 20-26 and 20-27. The responses in Fig. 20.7 are similar to those in Fig. 20.6, but there are a few significant differences. Both of the predicted responses in Fig. 20.7 differ from the actual response for $t > 4$, as a result of the model inaccuracy. Figure 20.7 demonstrates that the corrected predictions are much more accurate than the uncorrected predictions even when a significant plant-model mismatch occurs. This improvement occurs because new information is used as soon as it becomes available.

20.2.2 Extensions of the Basic MPC Model Formulation

We will now consider several extensions of the basic MPC problem formulation that are important for practical applications.

Integrating Processes

The standard step-response model in Eq. 20-6 is not appropriate for an integrating process because its step response is unbounded. However, because the output rate of change, $\Delta y(k) = y(k+1) - y(k)$, is bounded, a simple modification eliminates this problem. Replacing $\hat{y}(k+1)$ in Eq. 20-6 by $\Delta \hat{y}(k+1) = \hat{y}(k+1) - \hat{y}(k)$ provides an appropriate step-response model for integrating processes (Hokanson and Gerstle, 1992):

$$\Delta \hat{y}(k+1) = \sum_{i=1}^{N-1} S_i \Delta u(k-i+1) + S_N u(k-N+1) \quad (20-31)$$

or, equivalently,

$$\hat{y}(k+1) = \hat{y}(k) + \sum_{i=1}^{N-1} S_i \Delta u(k-i+1) + S_N u(k-N+1) \quad (20-32)$$

Although the bias correction approach of Eq. 20-22 is not valid for integrating processes, several modifications are available (Qin and Badgwell, 2003).

Known Disturbances

If a disturbance variable is known or can be measured, it can be included in the step-response model. Let d denote a measured disturbance and $\{S_i^d\}$ its step-response coefficients. Then the standard step-response model in Eq. 20-6 can be modified by adding a disturbance term,

$$\hat{y}(k+1) = \sum_{i=1}^{N-1} S_i \Delta u(k-i+1) + S_N u(k-N+1) + \sum_{i=1}^{N_d-1} S_i^d \Delta d(k-i+1) + S_{N_d}^d d(k-N_d+1) \quad (20-33)$$

where N_d is the number of step-response coefficients for the disturbance variable (in general, $N_d \neq N$). This same type of modification can be made to other step-response models such as Eq. 20-19 or 20-24. However, predictions made more than one step ahead require an assumption about future disturbances. If no other information is available, the usual assumption is that the future disturbances will be equal to the current disturbance: $d(k+j) = d(k)$ for $j = 1, 2, \dots, P$. However, if a disturbance model is available, the prediction accuracy can improve.

20.3 PREDICTIONS FOR MIMO MODELS

The previous analysis for SISO systems can be generalized to MIMO systems by using the Principle of Superposition. For simplicity, we first consider a process control problem with two outputs, y_1 and y_2 , and two inputs, u_1 and u_2 . The predictive model consists of two equations and four individual step-response models, one for each input-output pair:

$$\hat{y}_1(k+1) = \sum_{i=1}^{N-1} S_{11,i} \Delta u_1(k-i+1) + S_{11,N} u_1(k-N+1) + \sum_{i=1}^{N-1} S_{12,i} \Delta u_2(k-i+1) + S_{12,N} u_2(k-N+1) \quad (20-34)$$

$$\hat{y}_2(k+1) = \sum_{i=1}^{N-1} S_{21,i} \Delta u_1(k-i+1) + S_{21,N} u_1(k-N+1) + \sum_{i=1}^{N-1} S_{22,i} \Delta u_2(k-i+1) + S_{22,N} u_2(k-N+1) \quad (20-35)$$

where $S_{12,i}$ denotes the i th step-response coefficient for the model that relates y_1 and u_2 . The other step-response coefficients are defined in an analogous manner. This MIMO model is a straightforward generalization of the SISO model in Eq. 20-6. In general, a different model horizon can be specified for each input-output pair. For example, the upper limits for the summations in Eq. 20-35 can be specified as N_{21} and N_{22} , if y_2 has very different settling times for changes in u_1 and u_2 .

Next, the analysis is generalized to MIMO models with arbitrary numbers of inputs and outputs. Suppose that there are r inputs and m outputs. In a typical MPC application, $r < 20$ and $m < 40$, but applications with much larger numbers of inputs and outputs have also been reported (Qin and Badgwell, 2003; Canney, 2003). It is useful to display the individual step-response models graphically as shown in Fig. 20.8 (Hokanson and Gerstle, 1992), where the output variables (or CVs) are arranged as the columns and the

inputs and disturbances (the MVs and DVs) are arranged as the rows.

It is convenient to express MIMO step-response models in vector-matrix notation. Let the output vector be $\mathbf{y} = [y_1, y_2, \dots, y_m]^T$ and the input vector be $\mathbf{u} = [u_1, u_2, \dots, u_r]^T$ where superscript T denotes the transpose of a vector of matrix. In analogy with the derivation of Eq. 20-24 for SISO systems, the MIMO model for the corrected predictions can be expressed in dynamic matrix form:

$$\tilde{\mathbf{Y}}(\mathbf{k} + \mathbf{1}) = \mathbf{S}\Delta\mathbf{U}(\mathbf{k}) + \hat{\mathbf{Y}}^o(\mathbf{k} + \mathbf{1}) + [\mathbf{y}(\mathbf{k}) - \hat{\mathbf{y}}(\mathbf{k})] \quad (20-36)$$

where $\tilde{\mathbf{Y}}(\mathbf{k} + \mathbf{1})$ is the mP -dimensional vector of corrected predictions over the prediction horizon P ,

$$\tilde{\mathbf{Y}}(\mathbf{k} + \mathbf{1}) \triangleq \text{col} [\tilde{\mathbf{y}}(\mathbf{k} + \mathbf{1}), \tilde{\mathbf{y}}(\mathbf{k} + \mathbf{2}), \dots, \tilde{\mathbf{y}}(\mathbf{k} + \mathbf{P})] \quad (20-37)$$

$\hat{\mathbf{Y}}^o(\mathbf{k} + \mathbf{1})$ is the mP -dimensional vector of predicted unforced responses,

$$\hat{\mathbf{Y}}^o(\mathbf{k} + \mathbf{1}) \triangleq \text{col} [\hat{\mathbf{y}}^o(\mathbf{k} + \mathbf{1}), \hat{\mathbf{y}}^o(\mathbf{k} + \mathbf{2}), \dots, \hat{\mathbf{y}}^o(\mathbf{k} + \mathbf{P})] \quad (20-38)$$

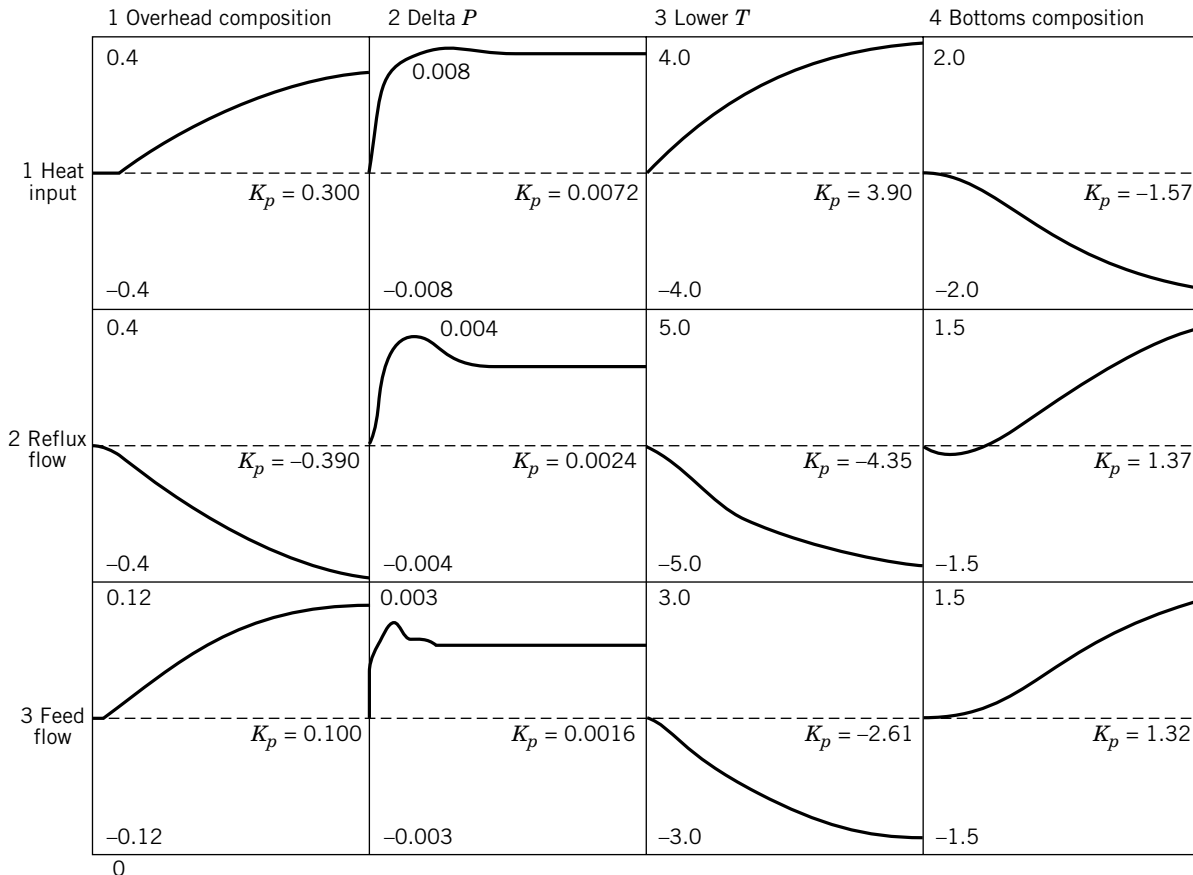


Figure 20.8 Individual step-response models for a distillation column with three inputs and four outputs. Each model represents the step response for 120 minutes (Hokanson and Gerstle, 1992).

and $\Delta \mathbf{U}(k)$ is the rM -dimensional vector of the next M control moves,

$$\Delta \mathbf{U}(k) \triangleq \text{col} [\Delta \mathbf{u}(k), \Delta \mathbf{u}(k+1), \dots, \Delta \mathbf{u}(k+M-1)] \quad (20-39)$$

The $mP \times m$ matrix (in Eq. 20-36) is defined as

$$\mathbf{S} \triangleq \underbrace{[\mathbf{I}_m \ \mathbf{I}_m \ \cdots \ \mathbf{I}_m]}_{P \text{ times}}^T \quad (20-40)$$

where \mathbf{I}_m denotes the $m \times m$ identity matrix.

The dynamic matrix \mathbf{S} is defined as

$$\mathbf{S} \triangleq \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{S}_2 & \mathbf{S}_1 & \mathbf{0} & \vdots \\ \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{S}_M & \mathbf{S}_{M-1} & \cdots & \mathbf{S}_1 \\ \mathbf{S}_{M+1} & \mathbf{S}_M & \cdots & \mathbf{S}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_P & \mathbf{S}_{P-1} & \cdots & \mathbf{S}_{P-M+1} \end{bmatrix} \quad (20-41)$$

where \mathbf{S}_i is the $m \times r$ matrix of step-response coefficients for the i th time step.

$$\mathbf{S}_i \triangleq \begin{bmatrix} S_{11,i} & S_{12,i} & \cdots & S_{1r,i} \\ S_{21,i} & \cdots & \cdots & S_{2r,i} \\ \vdots & \vdots & \vdots & \vdots \\ S_{m1,i} & \cdots & \cdots & S_{mr,i} \end{bmatrix} \quad (20-42)$$

Note that the dynamic matrix in Eq. 20-41 for MIMO systems has the same structure as the one for SISO systems in Eq. 20-20.

The dimensions of the vectors and matrices in Eq. 20-36 are as follows. Both $\tilde{\mathbf{Y}}(k+1)$ and $\tilde{\mathbf{Y}}^o(k+1)$ are mP -dimensional vectors where m is the number of outputs and P is the prediction horizon. Also, $\Delta \mathbf{U}(k)$ is an rM -dimensional vector where r is the number of manipulated inputs and M is the control horizon. Consequently, the dimensions of step-response matrix \mathbf{S} are $mP \times rM$. The MIMO model in (20-36) through (20-42) is the MIMO generalization of the SISO model in (20-24). It is also possible to write MIMO models in an alternative form, a generalization of Eqs. 20-34 and 20-35. An advantage of this alternative formulation is that the new dynamic matrix is partitioned into the individual SISO models, a convenient form for real-time predictions.

For stable models, the predicted unforced response, $\hat{\mathbf{Y}}^o(k+1)$ in Eq. 20-38, can be calculated from a recursive relation (Lundström et al., 1995) that is in the form of a discrete-time version of a state-space model:

$$\hat{\mathbf{Y}}^o(k+1) = \mathbf{M} \hat{\mathbf{Y}}^o(k) + \mathbf{S}^* \Delta \mathbf{u}(k) \quad (20-43)$$

where:

$$\hat{\mathbf{Y}}^o(k) = \text{col} [\hat{\mathbf{y}}^o(k), \hat{\mathbf{y}}^o(k+1), \dots, \hat{\mathbf{y}}^o(k+P-1)] \quad (20-44)$$

$$\mathbf{M} \triangleq \begin{bmatrix} \mathbf{0} & \mathbf{I}_m & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_m & \ddots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_m \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_m \end{bmatrix} \quad (20-45)$$

$$\mathbf{S}^* \triangleq \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_{P-1} \\ \mathbf{S}_P \end{bmatrix} \quad (20-46)$$

where \mathbf{M} is an $mP \times mP$ matrix and \mathbf{S}^* is an $mP \times r$ matrix. The MIMO models in Eqs. 20-36 through 20-46 can be extended to include measured disturbances and integrating variables, in analogy to the SISO case in the previous section.

Most of the current MPC research is based on state-space models, because they provide an important theoretical advantage, namely, a unified framework for both linear and nonlinear control problems. State-space models are also more convenient for theoretical analysis and facilitate a wider range of output feedback strategies (Rawlings, 2000; Maciejowski, 2002; Qin and Badgwell, 2003).

20.4 MODEL PREDICTIVE CONTROL CALCULATIONS

The flowchart in Fig. 20.9 provides an overview of the MPC calculations. The seven steps are shown in the order they are performed at each control execution time. For simplicity, we assume that the control execution times coincide with the measurement sampling instants.

In MPC applications, the calculated MV moves are usually implemented as set points for regulatory control loops at the Distributed Control System (DCS) level, such as flow control loops. If a DCS control loop has been disabled or placed in manual, the MV is no longer available for control. In this situation, the control degrees of freedom are reduced by one. Even though an MV is unavailable for control, it can serve as a disturbance variable if it is measured.

In Step 1 of the MPC calculations, new process data are acquired via the regulatory control system (DCS) that is interfaced to the process. Then new output predictions are calculated in Step 2 using the process model and the new data (see Eqs. 20-21 and 20-22, for example).

Before each control execution, it is necessary to determine which outputs (CVs), inputs (MVs), and disturbance variables (DVs) are currently available for the MPC calculations. This Step 3 activity is referred to

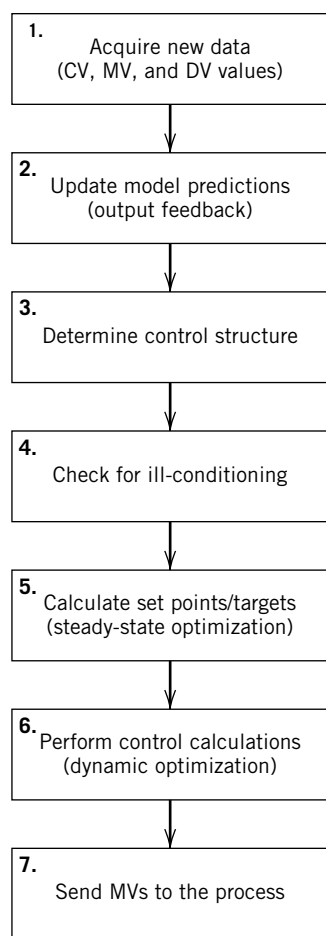


Figure 20.9 Flow chart for MPC calculations (modified from Qin and Badgwell (2003)).

as determining the current control structure. The variables available for the control calculations can change from one control execution time to the next, for a variety of reasons. For example, a sensor may not be available due to routine maintenance or recalibration. Output variables are often classified as being either *critical* or *noncritical*. If the sensor for a critical output is not available, the MPC calculations can be stopped immediately or after a specified number of control execution steps. For a noncritical output, missing measurements could be replaced by model predictions or the output could be removed from the control structure (Qin and Badgwell, 2003).

If the control structure changes from one control execution time to another, the subsequent control calculations can become *ill-conditioned*. It is important to identify and correct these situations before executing the MPC calculations in Steps 5 and 6. Ill-conditioning occurs when the available MVs have very similar effects on two or more outputs. For example, consider a high-purity distillation column where the product compositions are controlled by manipulating the reflux flow rate and the reboiler heat duty. Ill-conditioning occurs

because each input has approximately the same effect on both outputs, but in different directions. As a result, the process gain matrix is nearly singular, and large input movements are required to control these outputs independently. Consequently, it is important to check for ill-conditioning (Step 4) by calculating the *condition number* of the process gain matrix for the current control structure (see Chapter 16). If ill-conditioning is detected, effective strategies are available for its removal (Maciejowski, 2002; Qin and Badgwell, 2003).

In MPC applications, the major benefits result from determining the optimal operating conditions (set-point calculations) and from moving the process to these set points in an optimal manner based on the control calculations. Both types of calculations optimize a specified objective function while satisfying inequality constraints, such as upper and lower limits on the inputs or outputs. Set-point calculations are the subject of Section 20.5, while control calculations are considered in the next section.

The final step, Step 7 of Fig. 20.9, is to implement the calculated control actions, usually as set points to regulatory PID control loops at the DCS level.

20.4.1 Unconstrained MPC

This section considers the control calculations of Step 6 for the special case in which inequality constraints are not included in the problem formulation. In Section 20.4.2, the analysis is extended to the more typical situation where there are inequality constraints on \mathbf{u} , $\Delta\mathbf{u}$, and \mathbf{y} .

As noted earlier, the MPC control calculations are based on both current measurements and model predictions. The control objective is to calculate a set of control moves (MV changes) that make the corrected predictions as close to a reference trajectory as possible. Thus, an optimization approach is employed. For unconstrained linear control problems, an analytical expression for the MPC control law is available.

Reference Trajectory

In MPC applications, a reference trajectory can be used to make a gradual transition to the desired set point. The *reference trajectory* \mathbf{y}_r can be specified in several different ways (Maciejowski, 2002; Qin and Badgwell, 2003; Rossiter, 2003). We briefly introduce several typical approaches.

Let the reference trajectory over the prediction horizon P be denoted as

$$\mathbf{Y}_r(k+1) \triangleq \text{col}[\mathbf{y}_r(k+1), \mathbf{y}_r(k+2), \dots, \mathbf{y}_r(k+P)] \quad (20-47)$$

where \mathbf{Y}_r is an mP -dimensional vector. A reasonable approach is to specify the reference trajectory to be the *filtered set point*,

$$y_{i,r}(k+j) = (\alpha_i)^j y_{i,r}(k) + [1 - (\alpha_i)^j] y_{i,sp}(k) \quad (20-48)$$

for $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, P$

where $y_{i,r}$ is the i th element of \mathbf{y}_r , y_{sp} denotes the set point, and α_i is a filter constant, $0 < \alpha_i < 1$. For $j = 1$, Eq. 20-48 reduces to the set-point filtering expression for PID controllers that was considered in Chapter 11. It is also equivalent to the exponential filter introduced in Chapter 17. Note that $\mathbf{y}_r = \mathbf{y}_{sp}$ for the limiting case of $\alpha_i = 0$. An alternative approach is to specify the reference trajectory for the i th output as an exponential trajectory from the current measurement $y_i(k)$ to the set point, $y_{i,sp}(k)$:

$$y_{i,r}(k+j) = (\alpha_i)^j y_i(k) + [1 - (\alpha_i)^j] y_{i,sp}(k) \quad (20-49)$$

for $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, P$

In some commercial MPC products, the desired reference trajectory for each output is specified indirectly by a *performance ratio* for the output. The performance ratio is defined to be the ratio of the desired closed-loop settling time to the open-loop settling time. Thus, small values of the performance ratios correspond to small values of α_i in (20-48) or (20-49).

Model Predictive Control Law

The control calculations are based on minimizing the predicted deviations from the reference trajectory. Let k denote the current sampling instant. The *predicted error vector*, $\hat{\mathbf{E}}(k+1)$, is defined as

$$\hat{\mathbf{E}}(k+1) \triangleq \mathbf{Y}_r(k+1) - \tilde{\mathbf{Y}}(k+1) \quad (20-50)$$

where $\tilde{\mathbf{Y}}(k+1)$ was defined in (20-37). Similarly, $\hat{\mathbf{E}}^o(k+1)$ denotes the *predicted unforced error vector*,

$$\hat{\mathbf{E}}^o(k+1) \triangleq \mathbf{Y}_r(k+1) - \tilde{\mathbf{Y}}^o(k+1) \quad (20-51)$$

where the *corrected prediction for the unforced case*, $\tilde{\mathbf{Y}}^o(k+1)$, is defined as

$$\tilde{\mathbf{Y}}^o(k+1) \triangleq \hat{\mathbf{Y}}^o(k+1) + \mathbf{I}[\mathbf{y}(k) - \hat{\mathbf{y}}(k)] \quad (20-52)$$

Thus, $\hat{\mathbf{E}}^o(k+1)$ represents the predicted deviations from the reference trajectory when no further control action is taken, that is, the predicted deviations when $\Delta \mathbf{u}(k+j) = \mathbf{0}$ for $j = 0, 1, \dots, M-1$. Note that $\hat{\mathbf{E}}(k+1)$ and $\hat{\mathbf{E}}^o(k+1)$ are mP -dimensional vectors.

The general objective of the MPC control calculations is to determine $\Delta \mathbf{U}(k)$, the control moves for the next M time intervals,

$$\Delta \mathbf{U}(k) = \text{col} [\Delta \mathbf{u}(k), \Delta \mathbf{u}(k+1), \dots, \Delta \mathbf{u}(k+M-1)] \quad (20-53)$$

The rM -dimensional vector $\Delta \mathbf{U}(k)$ is calculated so that an objective function (also called a *performance index*) is minimized. Typically, either a linear or a quadratic objective function is employed. For unconstrained MPC,

the objective function is based on minimizing some (or all) of three types of deviations or errors (Qin and Badgwell, 2003):

1. The predicted errors over the predicted horizon, $\tilde{\mathbf{E}}(k+1)$
2. The next M control moves, $\Delta \mathbf{U}(k)$
3. The deviations of $\mathbf{u}(k+i)$ from its desired steady-state value \mathbf{u}_{sp} over the control horizon

For MPC based on linear process models, both linear and quadratic objective functions can be used (Maciejowski, 2002; Qin and Badgwell, 2003). To demonstrate the MPC control calculations, consider a quadratic objective function J based on the first two types of deviations:

$$\min_{\Delta \mathbf{U}(k)} \mathbf{J} = \hat{\mathbf{E}}(k+1)^T \mathbf{Q} \hat{\mathbf{E}}(k+1) + \Delta \mathbf{U}(k)^T \mathbf{R} \Delta \mathbf{U}(k) \quad (20-54)$$

where \mathbf{Q} is a positive-definite weighting matrix and \mathbf{R} is a positive semi-definite matrix. Both are usually diagonal matrices with positive diagonal elements. The weighting matrices are used to weight the most important elements of $\hat{\mathbf{E}}(k+1)$ or $\Delta \mathbf{U}(k)$, as described in Section 20.6. If diagonal weighting matrices are specified, these elements are weighted individually.

The MPC control law that minimizes the objective function in Eq. (20-54) can be calculated analytically.

$$\Delta \mathbf{U}(k) = (\mathbf{S}^T \mathbf{Q} \mathbf{S} + \mathbf{R})^{-1} \mathbf{S}^T \mathbf{Q} \hat{\mathbf{E}}^o(k+1) \quad (20-55)$$

This control law can be written in a more compact form,

$$\Delta \mathbf{U}(k) = \mathbf{K}_c \hat{\mathbf{E}}^o(k+1) \quad (20-56)$$

where the controller gain matrix \mathbf{K}_c is defined to be

$$\mathbf{K}_c \triangleq (\mathbf{S}^T \mathbf{Q} \mathbf{S} + \mathbf{R})^{-1} \mathbf{S}^T \mathbf{Q} \quad (20-57)$$

Note that \mathbf{K}_c is an $rM \times mP$ matrix that can be evaluated off-line rather than on-line provided that the dynamic matrix \mathbf{S} and weighting matrices, \mathbf{Q} and \mathbf{R} , are constant.

The MPC control law in Eq. 20-56 can be interpreted as a multivariable, proportional control law based on the predicted error rather than the conventional control error (set point–measurement). The control law utilizes the latest measurement $\mathbf{y}(k)$ because it appears in the expressions for the corrected prediction $\tilde{\mathbf{y}}(k)$, and thus also in the predicted unforced error, $\tilde{\mathbf{E}}^o(k+1)$. Furthermore, the MPC control law in Eq. 20-56 implicitly contains integral control action because \mathbf{u} tends to change until the unforced error $\hat{\mathbf{E}}^o$ becomes zero. Thus, offset is eliminated for set-point changes or sustained disturbances.

Although the MPC control law calculates a set of M input moves, $\Delta \mathbf{U}(k)$, only the first control move,

$\Delta u(k)$, is actually implemented. Then at the next sampling instant, new data are acquired and a new set of control moves is calculated. Once again, only the first control move is implemented. These activities are repeated at each sampling instant, and the strategy is referred to as a *receding horizon approach*. The first control move, $\Delta u(k)$, can be calculated from Eqs. 20-53 and 20-56,

$$\Delta u(k) = \mathbf{K}_{c1} \hat{E}^o(k + 1) \quad (20-58)$$

where matrix \mathbf{K}_{c1} is defined to be the first r rows of \mathbf{K}_c . Thus, \mathbf{K}_{c1} has dimensions of $r \times mP$.

It may seem strange to calculate an M -step control policy and then only implement the first move. The important advantage of this receding horizon approach is that new information in the form of the most recent measurement $y(k)$ is utilized immediately instead of being ignored for the next M sampling instants. Otherwise, the multistep predictions and control moves would be based on old information and thus be adversely affected by unmeasured disturbances, as demonstrated in Example 20.4.

The calculation of \mathbf{K}_c requires the inversion of an $rM \times rM$ matrix where r is the number of input variables and M is the control horizon. For large problems with many inputs, the required computational effort can be reduced by using *input blocking* (Maciejowski, 2002; Qin and Badgwell, 2003). In this approach, the inputs are not changed at every sampling instant. Instead, $\Delta u = \mathbf{0}$ for “blocks” of sampling instants. Input blocking is illustrated in Fig. 20.10 where a single input is changed at each sampling instant for the first four sampling instants (k through $k + 3$). Starting at $k + 4$, u is blocked so that it changes every three sampling instants until the steady-state value is reached at $k + 13$. The design parameters are the block length and the time at which blocking begins.

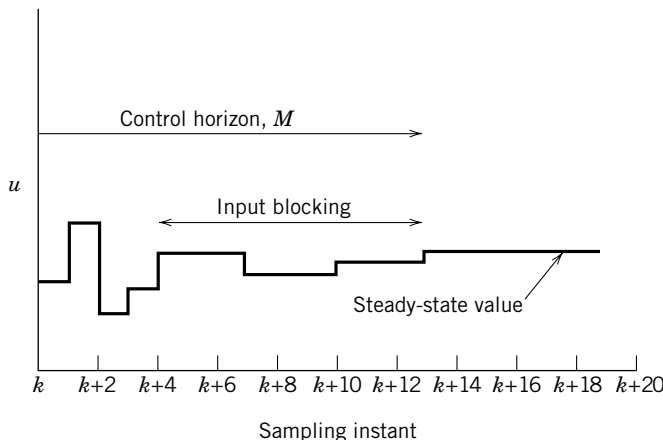


Figure 20.10 Input blocking.

20.4.2 MPC with Inequality Constraints

Inequality constraints on input and output variables are important characteristics for MPC applications. In fact, inequality constraints were a primary motivation for the early development of MPC. Input constraints occur as a result of physical limitations on plant equipment such as pumps, control valves, and heat exchangers. For example, a manipulated flow rate might have a lower limit of zero and an upper limit determined by the pump, control valve, and piping characteristics. The dynamics associated with large control valves impose rate-of-change limits on manipulated flow rates.

Constraints on output variables are a key component of the plant operating strategy. For example, a common distillation column control objective is to maximize the production rate while satisfying constraints on product quality and avoiding undesirable operating regimes such as flooding or weeping. Additional examples of inequality constraints were given in Chapter 19. The set of inequality constraints for u and y define an operating window for the process, as shown in Fig. 19.6.

Inequality constraints can be included in the control calculations in many different ways (Maciejowski, 2002; Qin and Badgwell, 2003). It is convenient to make a distinction between *hard constraints* and *soft constraints*. As the name implies, a hard constraint cannot be violated at any time. By contrast, a soft constraint can be violated, but the amount of violation is penalized by a modification of the objective function, as described below. This approach allows small constraint violations to be tolerated for short periods of time.

For MPC the inequality constraints for u and Δu are typically hard constraints specified as upper and lower limits:

$$u^-(k) \leq u(k + j) \leq u^+(k) \quad j = 0, 1, \dots, M - 1 \quad (20-59)$$

$$\Delta u^-(k) \leq \Delta u(k + j) \leq \Delta u^+(k) \quad j = 0, 1, \dots, M - 1 \quad (20-60)$$

The analogous hard constraints for the predicted outputs are:

$$y^-(k + j) \leq \tilde{y}(k + j) \leq y^+(k + j) \quad j = 1, 2, \dots, P \quad (20-61)$$

Unfortunately, hard output constraints can result in infeasible solutions for the optimization problem, especially for large disturbances. Consequently, output constraints are usually expressed as soft constraints involving *slack variables* s_j (Qin and Badgwell, 2003):

$$y^-(k + j) - s_j \leq \tilde{y}(k + j) \leq y^+(k + j) + s_j \quad j = 1, 2, \dots, P \quad (20-62)$$

The numerical values of the slack variables can be determined during constrained optimization if the performance

index in Eq. 20-54 is modified by adding a penalty term for the slack variables. Thus, an mP -dimensional vector of slack variables is defined as $\bar{\mathbf{S}} \triangleq \text{col} [s_1, s_2, \dots, s_P]$. The modified performance index is

$$\min_{\Delta \mathbf{U}(k)} \mathbf{J} = \hat{\mathbf{E}}(k+1)^T \mathbf{Q} \hat{\mathbf{E}}(k+1) + \Delta \mathbf{U}(k)^T \mathbf{R} \Delta \mathbf{U}(k) + \bar{\mathbf{S}}^T \mathbf{T} \bar{\mathbf{S}} \quad (20-63)$$

where \mathbf{T} is an $mP \times mP$ weighting matrix for the slack variables. Note that inequality constraints in (20-61) and (20-62) are imposed on the corrected prediction $\tilde{\mathbf{y}}$, rather than the actual output \mathbf{y} , because future values of \mathbf{y} are not available. Consequently, \mathbf{y} may violate a constraint even though $\tilde{\mathbf{y}}$ does not, as a result of modeling errors. Slack variables can also be used to weight positive and negative errors, differently.

Range Control

An unusual feature of MPC applications is that many output variables do not have set points. For these outputs, the control objective is to keep them between upper and lower limits, an approach called *range control* (or *zone control*). The limits can vary with time, as shown in Eq. 20-61. The advantage of range control is that it creates additional degrees of freedom for the control calculations. Furthermore, many output variables such as the liquid level in a surge tank do not have to be regulated at a set point. Consequently, in many MPC applications, range control is the rule rather than the exception. Set points are only specified for output variables that must be kept close to a specified value (for example, pH or a quality variable). Note that control to a set point can be considered to be a special case of range control that occurs when the upper and lower limits in (20-61) are equal.

The constraint limits in Eqs. 20-59 to 20-62 can vary with time as a result of changes in process equipment or instrumentation. However, it can also be beneficial to allow the limits to change in a specified manner over the control or prediction horizons. For example, in the *limit funnel* technique, the output limits in (20-61) or (20-62) gradually become closer together over the prediction horizon (Maciejowski, 2002; Qin and Badgwell, 2003).

The introduction of inequality constraints results in a constrained optimization problem that can be solved numerically using linear or quadratic programming techniques (Edgar et al., 2001). As an example, consider the addition of inequality constraints to the MPC design problem in the previous section. Suppose that it is desired to calculate the M -step control policy $\Delta \mathbf{U}(k)$ that minimizes the quadratic objective function J in Eq. 20-54, while satisfying the constraints in Eqs. 20-59, 20-60, and 20-61. The output predictions are made using the step-response model in Eq. 20-36. This MPC

design problem can be solved numerically using the quadratic programming technique in Chapter 19.

20.5 SET-POINT CALCULATIONS

As indicated in Section 20.1 and Fig. 20.9, the MPC calculations at each control execution time are typically performed in two steps. First, the optimum set points (or *targets*) for the control calculations are determined. Then, a set of M control moves are generated by the control calculations, and the first move is implemented. In practical applications, significant economic benefits result from both types of calculations, but the steady-state optimization is usually more important. In this section, the set-point calculations are described in more detail.

The MPC set points are calculated so that they maximize or minimize an economic objective function. The calculations are usually based on linear steady-state models and a simple objective function, typically a linear or quadratic function of the MVs and CVs. The linear model can be a linearized version of a complex nonlinear model or the steady-state version of the dynamic model that is used in the control calculations. Linear inequality constraints for the MVs and CVs are also included in the steady-state optimization. The set-point calculations are repeated at each sampling instant because the active constraints can change frequently due to disturbances, instrumentation, equipment availability, or varying process conditions.

Because the set-point calculations are repeated as often as every minute, the steady-state optimization problem must be solved quickly and reliably. If the optimization problem is based on a linear process model, linear inequality constraints, and either a linear or a quadratic cost function, the linear and quadratic programming techniques discussed in Chapter 19 can be employed.

20.5.1 Formulation of the Set-Point Optimization Problem

Next, we provide an overview of the set-point calculation problem. More detailed descriptions are available elsewhere (Sorensen and Cutler, 1998; Kassman et al., 2000; Rawlings, 2000; Maciejowski, 2002).

Consider an MIMO process with r MVs and m CVs. Denote the current values of \mathbf{u} and \mathbf{y} as $\mathbf{u}(k)$ and $\mathbf{y}(k)$. The objective is to calculate the optimum set point \mathbf{y}_{sp} for the next control calculation (at $k+1$) and also to determine the corresponding steady-state value of \mathbf{u} , \mathbf{u}_{sp} . This value is used as the set point for \mathbf{u} for the next control calculation.

A general, linear steady-state process model can be written as

$$\Delta \mathbf{y} = \mathbf{K} \Delta \mathbf{u} \quad (20-64)$$

where \mathbf{K} is the steady-state gain matrix and $\Delta \mathbf{y}$ and $\Delta \mathbf{u}$ denote steady-state changes in \mathbf{y} and \mathbf{u} . It is convenient to define $\Delta \mathbf{y}$ and $\Delta \mathbf{u}$ as

$$\Delta \mathbf{y} \triangleq \mathbf{y}_{sp} - \mathbf{y}_{OL}(k) \quad (20-65)$$

$$\Delta \mathbf{u} \triangleq \mathbf{u}_{sp} - \mathbf{u}(k) \quad (20-66)$$

In Eq. 20-65 $\mathbf{y}_{OL}(k)$ represents the steady-state value of \mathbf{y} that would result if \mathbf{u} were held constant at its current value, $\mathbf{u}(k)$, until steady state was achieved. In general, $\mathbf{y}_{OL}(k) \neq \mathbf{y}(k)$ except for the ideal situation where the process is at steady state at time k . In order to incorporate output feedback, the steady-state model in Eq. 20-64 is modified as

$$\Delta \mathbf{y} = \mathbf{K} \Delta \mathbf{u} + [\mathbf{y}(k) - \hat{\mathbf{y}}(k)] \quad (20-67)$$

A representative formulation for the set-point optimization is to determine the optimum values, \mathbf{u}_{sp} and \mathbf{y}_{sp} , that minimize a quadratic objective function,

$$\min_{\mathbf{u}_{sp}, \mathbf{y}_{sp}} J_s = \mathbf{c}^T \mathbf{u}_{sp} + \mathbf{d}^T \mathbf{y}_{sp} + \mathbf{e}_y^T \mathbf{Q}_{sp} \mathbf{e}_y + \mathbf{e}_u^T \mathbf{R}_{sp} \mathbf{e}_u + \mathbf{S}^T \mathbf{T}_{sp} \mathbf{S} \quad (20-68)$$

subject to satisfying Eq. 20-64 and inequality constraints on the MVs and CVs:

$$\mathbf{u}^- \leq \mathbf{u}_{sp} \leq \mathbf{u}^+ \quad (20-69)$$

$$\Delta \mathbf{u}^- \leq \Delta \mathbf{u}_{sp} \leq \Delta \mathbf{u}^+ \quad (20-70)$$

$$\mathbf{y}^- - \mathbf{s} \leq \mathbf{y}_{sp} \leq \mathbf{y}^+ + \mathbf{s} \quad (20-71)$$

where

$$\mathbf{e}_y \triangleq \mathbf{y}_{sp} - \mathbf{y}_{ref} \quad (20-72)$$

$$\mathbf{e}_u \triangleq \mathbf{u}_{sp} - \mathbf{u}_{ref} \quad (20-73)$$

The \mathbf{s} vector in (20-71) denotes the slack elements. In (20-72) and (20-73), \mathbf{y}_{ref} and \mathbf{u}_{ref} are the desired steady-state values of \mathbf{y} and \mathbf{u} that are often determined by a higher-level optimization (for example, Level 4 in Fig. 19.1). The weighting factors in (20-68), \mathbf{c} , \mathbf{d} , \mathbf{Q}_{sp} , \mathbf{R}_{sp} , and \mathbf{T}_{sp} , are selected based on economic considerations. Although the weighting factors are constants in Eq. 20-68, in MPC applications they can vary with time to accommodate process changes or changes in economic conditions such as product prices or raw material costs. Similarly, it can be advantageous to allow the limits in Eqs. 20-69 to 20-71 (\mathbf{u}^- , \mathbf{u}^+ , etc.) to vary from one execution time to the next, as discussed in Section 20.4. Fortunately, new values of weighting factors and constraint limits are easily accommodated, because the optimum set points are recalculated at each execution time.

It is important to make a distinction between \mathbf{y}_{ref} and \mathbf{u}_{ref} , and \mathbf{y}_{sp} and \mathbf{u}_{sp} . Both pairs represent desired values of \mathbf{y} and \mathbf{u} , but they have different origins and are used in different ways. Reference values, \mathbf{y}_{ref} and \mathbf{u}_{ref} , are often

determined infrequently by a higher-level optimization. They are used as the desired values for the steady-state optimization of Step 5 of Fig. 20.9. By contrast, \mathbf{y}_{sp} and \mathbf{u}_{sp} are calculated at each MPC control execution time and serve as set points for the control calculations of Step 6.

We have emphasized that the goal of this steady-state optimization is to determine \mathbf{y}_{sp} and \mathbf{u}_{sp} , the set points for the control calculations in Step 6 of Fig. 20.9. But why not use \mathbf{y}_{ref} and \mathbf{u}_{ref} for this purpose? The reason is that \mathbf{y}_{ref} and \mathbf{u}_{ref} are ideal values that may not be attainable for the current plant conditions and constraints, which could have changed since \mathbf{y}_{ref} and \mathbf{u}_{ref} were calculated. Thus, steady-state optimization (Step 5) is necessary to calculate \mathbf{y}_{sp} and \mathbf{u}_{sp} , target values that more accurately reflect current conditions. In Eq. 20-68, \mathbf{y}_{sp} and \mathbf{u}_{sp} are shown as the independent values for the optimization. However, \mathbf{y}_{sp} can be eliminated by substituting the steady-state model, $\mathbf{y}_{sp} = \mathbf{K} \mathbf{u}_{sp}$.

Next, we demonstrate that the objective function J_s is quite flexible, by showing how it is defined for three different types of applications.

Application 1: Maximize operating profit.

In Chapter 19, real-time optimization was considered problems where the operating profit was expressed in terms of product values and feedstock and utility costs. If the product, feedstock, and utility flow rates are manipulated or disturbance variables in the MPC control structure, they can be included in objective function J_s . In order to maximize the operating profit (OP), the objective function is specified to be $J_s = -OP$, because minimizing J_s is equivalent to maximizing $-J_s$. The weighting matrices for two quadratic terms, \mathbf{Q}_{sp} and \mathbf{R}_{sp} , are set equal to zero.

Application 2: Minimize deviations from the reference values.

Suppose that the objective of the steady-state optimization is to calculate \mathbf{y}_{sp} and \mathbf{u}_{sp} so that they are as close as possible to the reference values, \mathbf{y}_{ref} and \mathbf{u}_{ref} . This goal can be achieved by setting $\mathbf{c} = \mathbf{0}$ and $\mathbf{d} = \mathbf{0}$ in (20-68). Weighting matrices \mathbf{Q}_{sp} , \mathbf{R}_{sp} , and \mathbf{T}_{sp} should be chosen according to the relative importance of the MVs, CVs, and constraint violations.

Application 3: Maximize the production rate.

Suppose that the chief control objective is to maximize a production rate while satisfying inequality constraints on the inputs and the outputs. Assume that the production rate can be adjusted via a flow control loop whose set point is denoted as u_{1sp} in the MPC control structure. Thus, the optimization objective is to maximize u_{1sp} , or equivalently, to minimize $-u_{1sp}$. Consequently, the performance index in (20-68) becomes $J_s = -u_{1sp}$. This expression can be derived by setting all of the weighting factors equal to zero except for c_1 , the first element of \mathbf{c} . It is chosen to be $c_1 = -1$.

The set-point optimization problem can be summarized as follows. At each sampling instant, the optimum values of \mathbf{u} and \mathbf{y} for the next sampling instant (\mathbf{u}_{sp} and \mathbf{y}_{sp}) are calculated by minimizing the cost function in Eq. 20-68, subject to satisfying the model equation 20-64 and the constraints in Eqs. 20-69 to 20-71. This optimization problem can be solved efficiently using the standard LP or QP techniques of Chapter 19.

Infeasible calculations can occur if the calculations of Steps 5 and 6 are based on constrained optimization, because feasible solutions do not always exist (Edgar et al., 2001). Infeasible problems can result when the control degrees of freedom are reduced (e.g., control valve maintenance), large disturbances occur, or the inequality constraints are inappropriate for current conditions. For example, the allowable operating window in Fig. 19.6 could disappear for inappropriate choices of the y_1 and y_2 limits. Other modifications can be made to ensure that the optimization problem always has a feasible solution (Kassmann et al., 2000).

In view of the dramatic decreases in the ratio of computer cost to performance in recent years, it can be argued that physically based, nonlinear process models should be used in the set-point calculations, instead of approximate linear models. However, linear models are still widely used in MPC applications for three reasons: First, linear models are reasonably accurate for small changes in \mathbf{u} and \mathbf{d} and can easily be updated based on current data or a physically based model. Second, some model inaccuracy can be tolerated, because the calculations are repeated on a frequent basis and they include output feedback from the measurements. Third, the computational effort required for constrained, nonlinear optimization is still relatively large, but is decreasing.

20.6 SELECTION OF DESIGN AND TUNING PARAMETERS

A number of design parameters must be specified in order to design an MPC system. In this section, we consider key design issues and recommended values for the parameters. Several design parameters can also be used to tune the MPC controller. The effects of the MPC design parameters will be illustrated in two examples.

Sampling period Δt and model horizon N . The sampling period Δt and model horizon N (in Eq. 20-6) should be chosen so that $N\Delta t = t_s$ where t_s is the settling time for the open-loop response. This choice ensures that the model reflects the full effect of a change in an input variable over the time required to reach steady state. Typically, $30 \leq N \leq 120$. If the output variables respond on different time scales, a different value of N can be used for each output, as noted earlier. Also, different model horizons can be used for the MVs and DVs, as illustrated in Eq. 20-33.

Control M and prediction P horizons. As control horizon M increases, the MPC controller tends to become more aggressive and the required computational effort increases. However, the computational effort can be reduced by input blocking, as shown in Fig. 20.10. Some typical rules of thumb are $5 \leq M \leq 20$ and $N/3 < M < N/2$. A different value of M can be specified for each input.

The prediction horizon P is often selected to be $P = N + M$ so that the full effect of the last MV move is taken into account. Decreasing the value of P tends to make the controller more aggressive. A different value of P can be selected for each output if their settling times are quite different. An infinite prediction horizon can also be used and has significant theoretical advantages (Maciejowski, 2002; Rawlings, 2000).

Weighting Matrices, Q and R

The output weighting matrix Q in Eq. 20-54 allows the output variables to be weighted according to their relative importance. Thus, an $mP \times mP$ diagonal Q matrix allows the output variables to be weighted individually, with the most important variables having the largest weights. For example, if a reactor temperature is considered more important than a liquid level, the temperature will be assigned a larger weighting factor. The inverse of a diagonal weighting factor is sometimes referred to as an *equal concern factor* (Qin and Badgwell, 2003).

It can be advantageous to adjust the output weighting over the prediction horizon. For example, consider an SISO model with a time delay θ . Suppose that an input change Δu occurs at $k = 0$. Then $y(k) = 0$ until $k\Delta t > \theta$ due to the time delay. Consequently, it would be reasonable to set the corresponding elements of the Q matrix equal to zero, or, equivalently, to make the corresponding predictions zero. These approaches tend to make the control calculations better conditioned (see Section 20.4).

As a second example, the elements of Q that correspond to predicted errors early in the prediction horizon (for example, at time $k + 1$) can be weighted more heavily than the predicted errors at the end of the horizon, $k + P$, or vice versa. The use of coincidence points is a special case of this strategy. Here, the corrected errors only have nonzero weights for a subset of the P sampling instants called coincidence points. The corrected errors at other times are given zero weighting. In Example 20.2 a simple predictive control strategy was derived based on a single coincidence point.

A time-varying Q matrix can also be used to implement soft constraints by real-time adjustment of Q . For example, if an output variable approaches an upper or lower limit, the corresponding elements of Q would be temporarily increased.

In a similar fashion, \mathbf{R} in Eq. 20-54 allows input MVs to be weighted according to their relative importance. This $rM \times rM$ matrix is referred to as the *input weighting matrix* or the *move suppression matrix*. It is usually chosen to be a diagonal matrix with the diagonal elements r_{ii} , referred to as *move suppression factors*. They provide convenient tuning parameters, because increasing the value of r_{ii} tends to make the MPC controller more conservative by reducing the magnitudes of the MV moves.

If a reference trajectory is employed, move suppression is not required, and thus \mathbf{R} can be set equal to zero.

Reference Trajectory α_i

In MPC applications, the desired future output behavior can be specified in several different ways: as a set point, high and low limits, a reference trajectory, or a funnel (Qin and Badgwell, 2003). Both the reference trajectory and the funnel approaches have a tuning factor that can be used to adjust the desired speed of response for each output. Consider Eq. 20-48 or 20-49, for example. As α_i increases from zero to one, the desired reference trajectory becomes slower. Alternatively, the performance ratio concept can be used to specify the reference trajectories. As mentioned earlier, the performance ratio is defined to be the ratio of the desired closed-loop settling time to the open-loop settling time.

The influence of MPC design parameters is illustrated by a simple example.

EXAMPLE 20.5

A process has the transfer function,

$$\frac{Y(s)}{U(s)} = \frac{e^{-s}}{(10s + 1)(5s + 1)}$$

- (a) Use Eq. 20-57 to calculate the controller gain matrix, \mathbf{K}_c , for $\mathbf{Q} = \mathbf{I}$, $\mathbf{R} = \mathbf{0}$ two cases:

- (i) $P = 3, M = 1$
- (ii) $P = 4, M = 2$

Assume that $N = 70$, $\Delta t = 1$, and that u is unconstrained for each case.

- (b) Compare the set-point responses of two MPC controllers and a digital PID controller with $\Delta t = 0.5$ and ITAE set-point tuning (Chapter 11): $K_c = 2.27$, $\tau_I = 16.6$, and $\tau_D = 1.49$. Compare both y and u responses.
- (c) Repeat (b) for a unit step disturbance and a PID controller with ITAE disturbance tuning: $K_c = 3.52$, $\tau_I = 6.98$, and $\tau_D = 1.73$.

SOLUTION

- (a) The step-response coefficients are obtained by evaluating the step response at the sampling instants, $t = i\Delta t = i$ (because $\Delta t = 1$):

$$S_1 = 0$$

$$S_i = 1 - 2e^{-0.1(i-1)} + e^{-0.2(i-1)} \text{ for } i = 2, 3, \dots, 70$$

The controller matrix \mathbf{K}_c for each case is shown in Table 20.2. Note that the dimensions of \mathbf{K} are different for the two cases, because \mathbf{K}_c has dimensions of $rM \times mP$, as noted earlier. For this SISO example, $r = m = 1$, and the values of M and P differ for the two cases.

Table 20.2 Feedback Matrices \mathbf{K}_c for Example 20.5

For $P = 3$ and $M = 1$:	$K_c = [0 \quad 7.79 \quad 28.3]$
For $P = 4$ and $M = 2$:	$K_c = \begin{bmatrix} 0 & 33.1 & 48.8 & -13.4 \\ 0 & -71.4 & -97.4 & 57.3 \end{bmatrix}$

- (b) The unit step response can be derived analytically using Laplace transforms:

$$y(t) = 0 \text{ for } t \leq 1$$

$$y(t) = 1 - 2e^{-0.1(t-1)} + e^{-0.2(t-1)} \text{ for } t > 1$$

Figure 20.11 compares the y and u responses for a unit set-point change. The two MPC controllers provide superior output responses with very small settling times, but their initial MV changes are larger than those for the PID controller. (Note the expanded time scale for u .)

- (c) For the step disturbance, the output responses for the MPC controllers in Fig. 20.12 have relatively small maximum deviations and are nonoscillatory. By comparison, the PID controller results in the largest maximum deviation and an oscillatory response. Of the two MPC controllers, the one designed using $P = 3$ and $M = 1$ provides a slightly more conservative response.

20.6.1 MPC Application: Distillation Column Model

In order to illustrate the effects of the MPC design parameters (M , P , \mathbf{Q} , and \mathbf{R}) for an MIMO problem, consider the Wood-Berry model that was introduced in Example 16.1:

$$\begin{bmatrix} X_D(s) \\ X_B(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s + 1} & \frac{-18.9e^{-3s}}{21s + 1} \\ \frac{6.6e^{-7s}}{10.9s + 1} & \frac{-19.4e^{-3s}}{14.4s + 1} \end{bmatrix} \begin{bmatrix} R(s) \\ S(s) \end{bmatrix} + \begin{bmatrix} 3.8e^{-8.1s} \\ 14.9s + 1 \\ 4.9e^{-3.4s} \\ 13.2s + 1 \end{bmatrix} F(s) \quad (20-74)$$

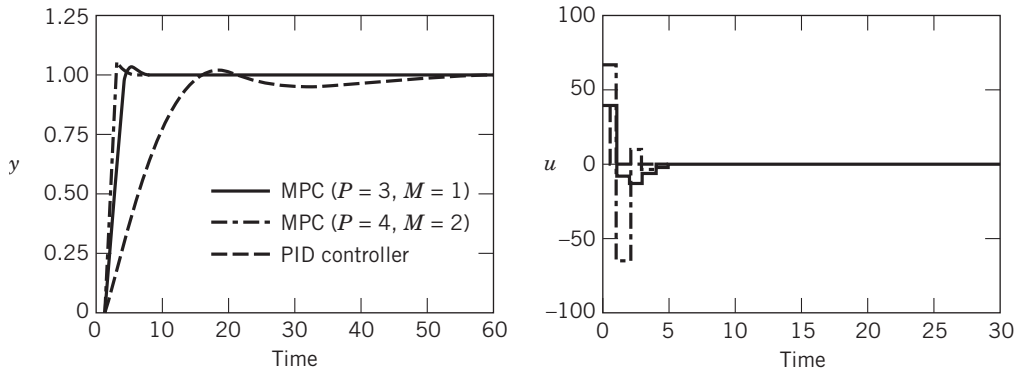


Figure 20.11 Set-point responses for Example 20.5.

The controlled variables (outputs) are the distillate and bottoms compositions (X_D and X_B); the manipulated variables (inputs) are the reflux flow rate and the steam flow rate to the reboiler (R and S); and feed flow rate F is an unmeasured disturbance variable.

Next, we compare a variety of MPC controllers and a multiloop control system, based on simulations performed using the MATLAB Model Predictive Control

Toolbox (Bemporad et al., 2009).¹ For each simulation the sampling period was $\Delta t = 1$ min, and saturation limits of ± 0.15 were imposed on each input. Unconstrained MPC controllers were designed using Eq. 20-55, while the constrained MPC controllers were based on the input constraints in Eq. 20-59. Some constrained MPC controllers were designed using an additional hard-output constraint of $|y_i| \leq 1.8$. In order to compare MPC and a standard multiloop control system, two PI controllers were simulated using the X_D-R/X_B-S control configuration from Example 16.1 and the controller settings in Table 20.3 reported by Lee et al. (1998).

Figures 20.13 and 20.14 compare the performance of the MPC and multiloop control systems for a +1% set-point change in X_B at $t = 0$, followed by two feed flow rate disturbances: a +30% increase at $t = 50$ min and a return to the original value at $t = 100$ min. The input and output variables are displayed as deviation variables. The numerical values of the integral of the absolute error (IAE) performance index (Chapter 11) are included for each output.

A comparison of Cases A and B in Fig. 20.13 indicates that unconstrained MPC is superior to the multiloop control system, because its output variables exhibit faster set-point responses, less oscillation, and smaller IAE values. In addition, the changes in the input variables are

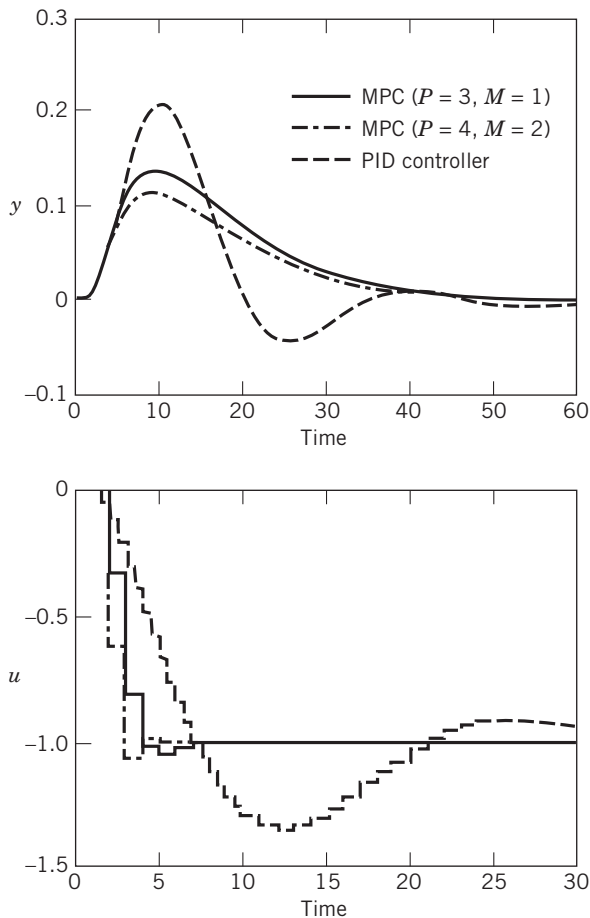


Figure 20.12 Disturbance responses for Example 20.5.

Table 20.3 PI Controller Settings for the Wood-Berry Model

Control Loop	K_c	τ_I (min)
$X_D - R$	0.85	7.21
$X_B - S$	-0.089	8.86

¹The code for the Wood-Berry example is available in this MATLAB Toolbox. A modified version of the code is included with Exercise 20.9. A newer version of the MPC Toolbox is also available, but without this example (Bemporad et al., 2009).

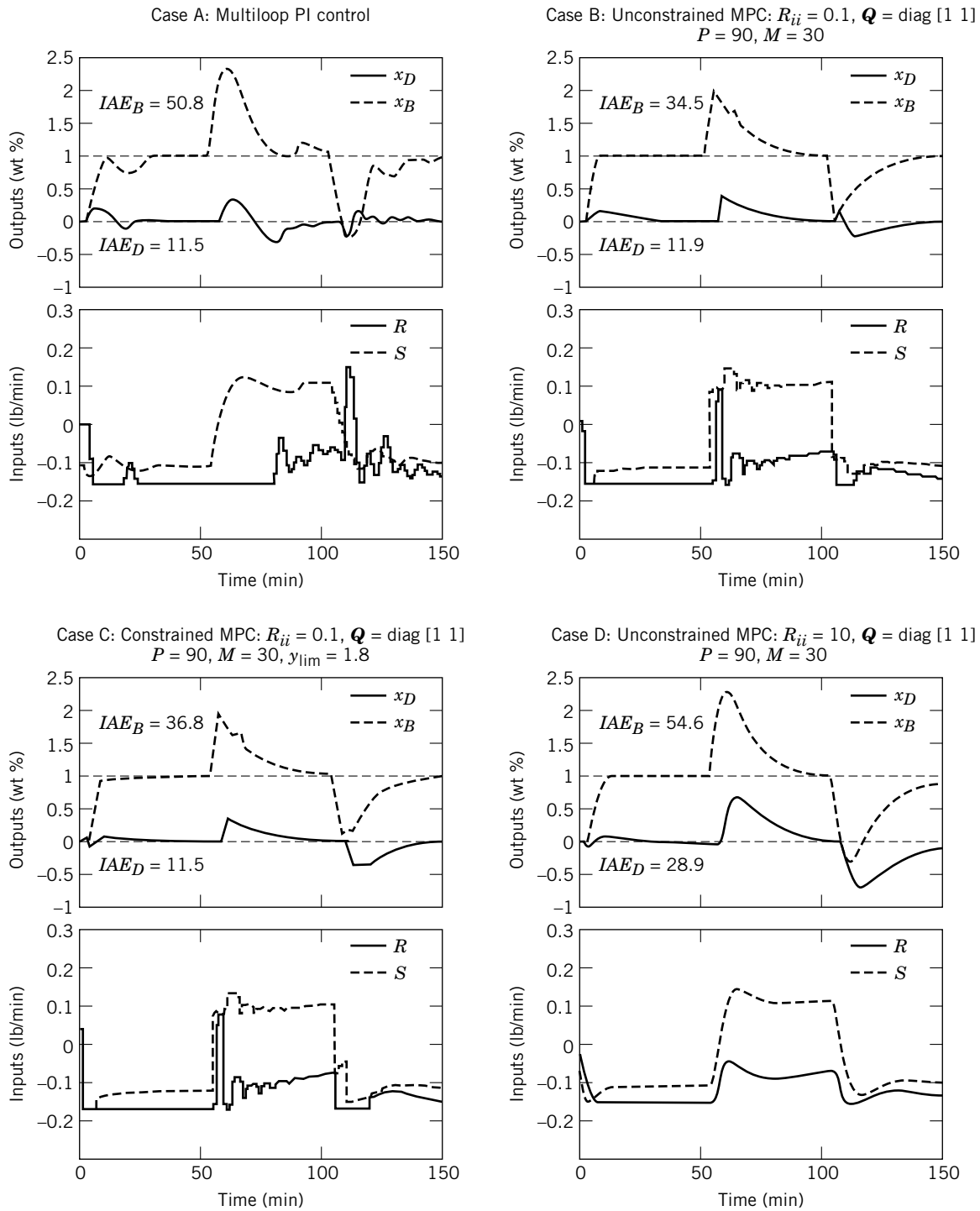


Figure 20.13 Comparison of multiloop PI control and MPC for the Wood-Berry model.

smoother for MPC. Case B is used as a “base case” for the comparisons in Figs. 20.13 and 20.14. Its MPC design parameters are shown in Fig. 20.13 and were selected according to the guidelines presented earlier.

Cases B and C in Fig. 20.13 provide a comparison of constrained and unconstrained MPC. These responses are very similar, with only small differences occurring,

mainly for the second disturbance. This somewhat surprising result can be interpreted as follows. The responses for constrained and unconstrained MPC are very similar because the inputs are saturated much of the time for both controllers. When one input saturates, the MPC controller only has a single degree of freedom left, the other input. By contrast, for larger control

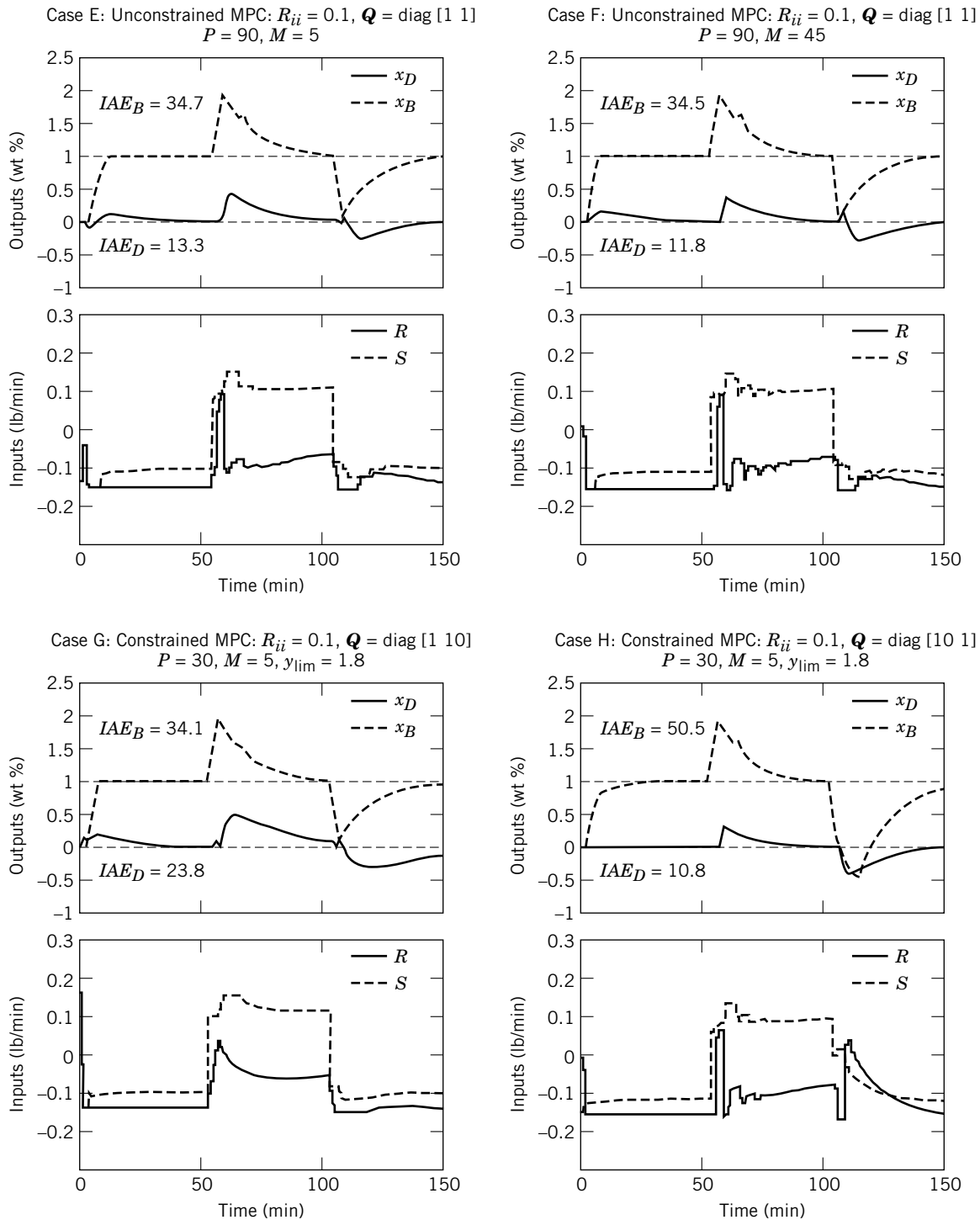


Figure 20.14 Effects of MPC design parameters for the Wood-Berry model.

problems (for example, 10×10), constrained MPC will have many more degrees of freedom. For these larger problems, constrained MPC tends to provide improved control due to the extra degrees of freedom and its awareness of the constraints and process interactions.

The effect of a diagonal move suppression matrix R is apparent from a comparison of Cases B and D.

When the diagonal elements, R_{ii} , are increased from 0.1 to 10, the MPC inputs become smoother and the output responses have larger deviations, higher IAE values, and longer settling times.

The effect of changing control horizon, M , is shown in Cases B, E, and F. The y responses and IAE values are quite similar for all three values of

M : 5, 30, and 45. However, the u responses are smoother for $M = 5$.

Cases G and H demonstrate that improved control of a designated output variable can be achieved by adjusting the elements of the Q matrix in Eq. 20-54. For Case G, x_B is weighted 10 times more heavily than x_D , in contrast to Case H, where the reverse situation occurs. Control of the more heavily weighted output improves at the expense of the other output, as indicated by smaller maximum deviations, IAE values, and settling times. For Cases G and H, $P = 30$, and the results are similar to other cases where $P = 90$.

20.7 IMPLEMENTATION OF MPC

This section provides an overview of the activities that are involved in designing and implementing a model predictive control system. For a new MPC application, a cost/benefit analysis is usually performed prior to project approval. Then the steps involved in the implementation of MPC can be summarized as follows (Hokanson and Gerstle, 1992; Qin and Badgwell, 2003):

1. Initial controller design
2. Pretest activity
3. Plant tests
4. Model development
5. Control system design and simulation
6. Operator interface design and operator training
7. Installation and commissioning
8. Measuring results and monitoring performance

Step 1: Initial Controller Design

The first step in MPC design is to select the controlled, manipulated, and measured disturbance variables. These choices determine the structure of the MPC control system and should be based on process knowledge and control objectives. In typical applications the number of controlled variables is less than or equal to 40, and the number of manipulated (input) variables is less than or equal to 20. These preliminary selections are reviewed in Step 5 and revised, if necessary. The input and measured disturbance variables that are varied during the plant tests of Step 3 should be chosen carefully. For example, if it is decided to add a new input variable later during Step 5, additional plant tests would be required, a nontrivial task. By contrast, additional output variables can be added to the MPC control structure later, if necessary, provided that these measurements were recorded during the plant tests.

Step 2: Pretest Activity

During the pretest activity (or *pretest*, for short), the plant instrumentation is checked to ensure that it is working properly. Remedial action may be required for faulty sensors, sticking control valves, and the like. Also, a decision may be made to install sensors for

some process variables that are not currently measured. The pretest also includes preliminary experimental tests to estimate the steady-state gains and approximate settling times for each input-output pair. This information is used to plan the full plant tests of Step 3.

As mentioned earlier, the results of the MPC control calculations are input moves that are implemented as set points for regulatory control loops. For example, if a cooling water flow rate is an MPC input variable, the MPC controller calculates the set point for the corresponding DCS control loop. Consequently, it is important to thoroughly check the performance of the DCS control system during the pretest, and to retune or reconfigure control loops if necessary.

These evaluation and maintenance activities are very important. If the basic instrumentation and DCS control system do not function properly, the MPC strategy will be ineffective, and the success of the MPC application will be jeopardized.

In the pretest experiments, each manipulated variable (MV) is *bumped* at least once by making a small step change. Steady-state gains and settling times are estimated from the step-response data using the techniques described in Chapter 6. Each measured disturbance variable (DV) should also be bumped, if possible. If not, the gains and settling times can be estimated from historical data for periods during which the disturbance variables changed significantly. During these bump tests, any existing DCS control loops for the output variables should be placed in manual. Thus, the pretest experiments are open-loop step tests (see Chapter 11). However, the MV and DV moves are usually implemented as set-point changes to the DCS loops for the DVs and MVs.

As part of the pretest, it is desirable to benchmark the performance of the existing control system for later comparison with MPC performance (Step 8). For example, the closed-loop responses for representative set-point changes and measured disturbances could be characterized using the performance criteria of Chapter 11. A baseline for the economic performance of the control system should also be established, although it is not always easy to do so.

Step 3: Plant Tests

The dynamic model for the MPC calculations is developed from data collected during special plant tests. The plant testing can be very time-consuming, typically requiring days, or even weeks, of around-the-clock experiments. The required test duration depends on the settling times of the outputs and the numbers of MVs and DVs. The excitation for the plant tests usually consists of changing an input variable or a disturbance variable (if possible) from one value to another, using either a series of step changes with different durations

or the pseudorandom-binary sequence (PRBS) that was introduced in Chapter 6. The plant test experiments are implemented in the same manner as the pretest experiments of Step 2.

It is traditional industrial practice to move each MV and DV individually. The magnitudes of the moves should be carefully chosen, because too small a move may result in the step responses being obscured by normal process fluctuations and measurement noise. On the other hand, too large a change may result in an output constraint violation or nonlinear process behavior that cannot be accurately described by a linear model.

The magnitude of the maximum allowable input changes can be estimated from knowledge of the output constraints and the estimated steady-state gains from the pretest. For example, suppose that $(\Delta u_j)_{\max}$ denotes the maximum change that can be made in u_j without violating a constraint for y_i . It can be estimated from the expression,

$$(\Delta u_j)_{\max} = \frac{(\Delta y_i)_{\max}}{\hat{K}_{ij}} \quad (20-75)$$

where $(\Delta y_i)_{\max}$ is the maximum allowable change in y_i and \hat{K}_{ij} is the estimated steady-state gain between y_i and u_j . However, this steady-state analysis does not guarantee that each y_i satisfies its constraints during transient responses.

The duration of the longest step test is equal to t_{\max} , the longest settling time that was observed during the pretest. Shorter step changes are also made, with the durations typically varying from $t_{\max}/8$ to $t_{\max}/2$. In order to ensure that sufficient data are obtained for model identification, each input variable is typically moved 8–15 times (Qin and Badgwell, 2003).

Some MPC vendors recommend a total plant testing period of $t_{\text{test}} = 6(r + p)t_{\max}$ where r is the number of input variables and p is the number of measured disturbance variables. In principle, t_{test} can be reduced by making simultaneous changes to several input (or disturbance) variables rather than the traditional sequential (“one-at-a-time”) approach. Also, it can be very difficult to identify poorly conditioned process models using the sequential approach. However, because of a number of practical considerations, input moves are traditionally made sequentially. In particular, simultaneous input moves tend to complicate the test management and make it more difficult to identify periods of abnormal operation by visual inspection of the test data. It is also more difficult to ensure that output constraints will not be violated. Because of similar practical considerations, step changes have been traditionally preferred over the pseudorandom binary sequence (PRBS) of Chapter 6.

Step 4: Model Development

The dynamic model is developed from the plant test data by selecting a model form (for example, a step-response

model) and then estimating the model parameters. However, first it is important to eliminate periods of test data during which plant upsets or other abnormal situations have occurred, such as control valve saturation or a DCS control loop having been placed in manual. Decisions to omit portions of the test data are based on visual inspection of the data, knowledge of the process, and experience. Parameter estimation is usually based on least-squares estimation (Chapter 6).

As part of the model development step, the model accuracy should be characterized, because this information is useful for subsequent system design and tuning. The characterization can include confidence intervals for the model predictions and/or model parameters. The confidence intervals can be calculated using standard statistical techniques (Ljung, 1999).

Step 5: Control System Design and Simulation

The MPC design is based on the control and optimization objectives, process constraints, and the dynamic model of the process. The preliminary control system design from Step 1 is critically evaluated and modified, if necessary. Then the MPC design parameters in Section 20.6 are selected, including the sampling periods, weighting factors, and control and prediction horizons. Next, the closed-loop system is simulated using the identified process model and a wide variety of process conditions to evaluate control system performance. The MPC design parameters are adjusted, if necessary, to obtain satisfactory control system performance and robustness over the specified range of operating conditions.

Step 6: Operator Interface Design and Operator Training

Because plant operators play a key role in manufacturing plants, it is important that the MPC operator interface meet their needs. Operator training is also important, because MPC concepts such as predictive control, multivariable interactions, and constraint handling are very different from conventional regulatory control concepts. For a standard multiloop control system, each input is adjusted based on measurements of a single output. By contrast, in MPC each input depends on all of the outputs. Thus, understanding why the MPC system responds the way that it does, especially in unusual operating conditions, can be very challenging for both operators and engineers.

Step 7: Installation and Commissioning

After a MPC control system is installed, it is first evaluated in a “prediction mode.” Model predictions are compared with measurements, but the process continues to be controlled by the existing control system (e.g., DCS). After the output predictions are judged to be satisfactory, the calculated MPC control moves are evaluated to see if they are reasonable. Finally, the MPC software is evaluated during closed-loop

operation with the calculated control moves implemented as set points to the DCS control loops. The MPC design parameters are tuned, if necessary. The commissioning period typically requires some troubleshooting and can take as long as, or even longer than, the plant tests of Step 3.

Step 8: Measuring Results and Monitoring Performance

The evaluation of MPC system performance is not easy, and widely accepted metrics and monitoring strategies are not available. However, useful diagnostic information is provided by basic statistics, such as the means and standard deviations for both measured variables, and calculated quantities, such as control errors and model residuals. Another useful statistic is the relative amount of time that an input is saturated or a constraint is violated, expressed as a percentage of the total time the MPC system is in service. These types of routine monitoring activities are considered in more detail in Chapter 21.

In Chapter 11, we considered a number of classical metrics for characterizing control system performance,

SUMMARY

Model predictive control is an important model-based control strategy devised for large multiple-input, multiple-output control problems with inequality constraints on the inputs and/or outputs. This chapter has considered both the theoretical and practical aspects of MPC. Applications typically involve two types of calculations: (1) a steady-state optimization to determine the optimum set points for the control calculations, and (2) control calculations to determine the MV changes that will drive the process to the set points. The success of model-based control strategies such as MPC depends strongly on the availability of a reasonably accurate process model. Consequently, model development is the most critical step in applying MPC. As Rawlings (2000) has noted, “feedback can overcome some effects

such as the IAE index, overshoot, settling time, and degree of oscillation. Though helpful, these metrics provide an incomplete picture of overall MPC performance. An important motivation for MPC is that it facilitates process operation closer to targets and limiting constraints. Thus, an evaluation of MPC performance should include measures of whether these objectives have been realized. If so, a noticeable improvement in process operation should be reflected in economically meaningful measures such as product quality, throughput, or energy costs. The various performance metrics should be calculated before and after the MPC system is installed.

MPC system performance should be monitored on a regular basis to ensure that performance does not degrade owing to changes in the process, instrumentation, or process conditions, including disturbances. If performance becomes significantly worse, retuning the controller or reidentifying all (or part) of the process model may be required. The development of MPC monitoring strategies is an active research area (Kozub, 2002; Jelali, 2006; McIntosh and Canney, 2008; Badwe et al., 2009).

of poor models, but starting with a poor process model is akin to driving a car at night without headlights.” Finally, the MPC design parameters should be chosen carefully, as illustrated by two simulation examples.

Model predictive control has had a major impact on industrial practice, with thousands of applications worldwide. MPC has become the method of choice for difficult control problems in the oil refining and petrochemical industries. However, it is not a panacea for all difficult control problems (Shinskey, 1994; Hugo, 2000; McIntosh and Canney, 2008). Furthermore, MPC has had much less impact in the other process industries. Performance monitoring of MPC systems is an important topic of current research interest.

REFERENCES

- Badgwell, T. A., and S. J. Qin, A Review of Nonlinear Model Predictive Control Applications, in *Nonlinear Predictive Control: Theory and Practice*, B. Kouvaritakis and M. Cannon (Eds.), Inst. Electrical Eng., London, 2001, Chapter 1.
- Badwe, A. S., R. D. Gudi, R. S. Patwardhan, S. L. Shah, and S. C. Patwardhan, Detection of Model-Plant Mismatch in MPC Applications, *J. Process Control*, **19**, 1305 (2009).
- Bemporad, A., M. Morari, and N. L. Ricker, *Model Predictive Control Toolbox 3*, User's Guide, Mathworks, Inc., Natick, MA, 2009.
- Camacho, E. F., and C. Bordons, *Model Predictive Control* 2nd ed., Springer-Verlag, New York, 2003.
- Canney, W. M., The Future of Advanced Process Control Promises More Benefits and Sustained Value, *Oil & Gas Journal*, **101** (16), 48 (2003).
- Clarke, D. W., C. Mohtadi, and P. S. Tufts, Generalized Predictive Control—Part I. The Basic Algorithm, *Automatica*, **23**, 137 (1987).
- Cutler, C. R., and B. L. Ramaker, Dynamic Matrix Control—A Computer Control Algorithm, *Proc. Joint Auto. Control Conf., Paper WP5-B*, San Francisco (1980).
- Edgar, T. F., D. M. Himmelblau, and L. Lasdon, *Optimization of Chemical Processes*, 2d ed., McGraw-Hill, New York, 2001.
- Hokanson, D. A., and J. G. Gerstle, Dynamic Matrix Control Multivariable Controllers, in *Practical Distillation Control*, W. L. Luyben (Ed.) Van Nostrand Reinhold, New York, 1992, p. 248.
- Hugo, A., Limitations of Model Predictive Controllers, *Hydrocarbon Process.*, **79** (1), 83 (2000).
- Jelali M., An Overview of Control Performance Assessment Technology and Industrial Applications, *Control Eng. Practice*, **14**, 441 (2006).
- Kano, M., and M. Ogawa, The State of the Art in Advanced Chemical Process Control in Japan, *Proc. IFAC Internat. Sympos on Advanced Control of Chemical Processes (ADCHEM 2009)*, Paper 240, Istanbul, Turkey (July 2009).

Kassman, D. E., T. A. Badgwell, and R. B. Hawkins, Robust Steady-State Target Calculations for Model Predictive Control, *AIChE J.*, **46**, 1007 (2000).

Kozub, D. J. Controller Performance Monitoring and Diagnosis. Industrial Perspective, *Preprints of the 15th Triennial World IFAC Congress*, Barcelona, Spain (July 2002).

Lee, J., W. Cho, and T. F. Edgar, Multiloop PI Controller Tuning for Interacting Multivariable Processes, *Computers and Chem. Engng.*, **22**, 1711 (1998).

Ljung, L., *System Identification*, 2d ed., Prentice Hall, Upper Saddle River, NJ, 1999.

Lundström, P., J. H. Lee, M. Morari, and S. Skogestad, Limitations of Dynamic Matrix Control, *Computers and Chem. Engng.*, **19**, 409 (1995).

Maciejowski, J. M., *Predictive Control with Constraints*, Prentice Hall, Upper Saddle River, NJ, 2002.

Maurath, P. R. D. A. Mellichamp, and D. E. Seborg, Predictive Controller Design for SISO Systems, *IEC Research*, **27**, 956 (1988).

McIntosh, A. R. and W. M. Canney, The Dirty Secrets of Model Predictive Controller Sustained Value, *Proc. Internat. Sympos. on Advanced Control of Industrial Processes (ADCONIP 2008)*, Paper MoB1.4, Jasper, Alberta, Canada (May 2008).

Morari, M., and J. H. Lee, Model Predictive Control: Past, Present, and Future, *Comput. and Chem. Eng.*, **23**, 667 (1999).

Morari, M., and N. L. Ricker, *Model Predictive Control Toolbox*, The Mathworks, Inc., Natick, MA, 1994.

Qin, S. J., and T. A. Badgwell, A Survey of Industrial Model Predictive Control Technology, *Control Eng. Practice*, **11**, 733 (2003).

Rawlings, J. B., Tutorial Overview of Model Predictive Control, *IEEE Control Systems*, **20**(3), 38 (2000).

Richalet, J. and D. O'Donovan, *Predictive Functional Control: Principles and Industrial Applications*, Springer-Verlag Ltd., London, 2009.

Richalet, J., A. Rault, J. L. Testud, and J. Papon, Model Predictive Heuristic Control: Applications to Industrial Processes, *Automatica*, **14**, 413 (1978).

Rossiter, J. A., *Model-Based Predictive Control: A Practical Approach*, CRC Press, Boca Raton, FL, 2003.

Shinskey, F. G., *Feedback Controllers for the Process Industries*, McGraw-Hill, New York, 1994.

Sorensen, R. C., and C. R. Cutler, LP Integrates Economics into Dynamic Matrix Control, *Hydrocarbon Process.*, **77**(9), 57 (1998).

White, D. C., Multivariable Control of a Chemical Plant Incinerator: An Industrial Case Study of Co-Linearity and Non-Linearity, *Proc. Internat. Sympos. on Advanced Control of Industrial Processes (ADCONIP 2008)*, Paper MoB1.1, Jasper, Alberta, Canada (May 2008).

EXERCISES

20.1 For the transfer functions

$$G_P(s) = \frac{2e^{-s}}{(10s + 1)(5s + 1)} \quad G_v = G_m = 1$$

- (a) Derive an analytical expression for the step response to a unit step change. Evaluate the step-response coefficients, $\{S_i\}$, for a sampling period of $\Delta t = 1$.
- (b) What value of model horizon N should be specified in order to ensure that the step-response model covers a period of at least 99% of the open-loop settling time? (That is, we require that $N\Delta t \geq t_{99}$ where t_{99} is the 99% settling time.)

20.2 A process (including sensor and control valve) can be modeled by the transfer function,

$$G(s) = \frac{2(1 - 9s)}{(15s + 1)(3s + 1)}$$

- (a) Derive an analytical expression for the response to a unit step change in the input.
- (b) Suppose that the maximum allowable value for the model horizon is $N = 30$. What value of the sampling period Δt should be specified to ensure that the step-response model covers a period of at least 99% of the open-loop settling time? (That is, we require that $N\Delta t \geq t_{99}$ where t_{99} is the 99% settling time.)

Use the analytical solution and this value of Δt to obtain a step-response model in the form of Eq. 20-1.

20.3 Control calculations for a control horizon of $M = 1$ can be performed either analytically or numerically. For the process model in Exercise 20.1, derive \mathbf{K}_{cl} for $\Delta t = 1$, $N = 50$, and $P = 5$, $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = \mathbf{0}$, using Eq. 20-57. Compare your answer with the analytical result reported by Maurath et al. (1988).

$$\mathbf{K}_{cl} = \frac{1}{\sum_{i=1}^P S_i^2} [S_1 \ S_2 \ S_3 \ \dots \ S_P]$$

20.4 Consider the transfer function model of Exercise 20.1. For each of the four sets of design parameters shown below, design a model predictive controller. Then do the following:

- (a) Compare the controllers for a unit step change in set point. Consider both the y and u responses.
- (b) Repeat the comparison of (a) for a unit step change in disturbance, assuming that $G_d(s) = G(s)$.
- (c) Which controller provides the best performance? Justify your answer.

Set No.	Δt	N	M	P	\mathbf{R}
(i)	2	40	1	5	0
(ii)	2	40	20	20	0
(iii)	2	40	3	10	0.01
(iv)	2	40	3	10	0.1

20.5 For Exercise 20.1, suppose that a constraint is placed on the manipulated variable, $u_{k+j} \leq 0.2$ for $j = 1, 2, \dots, M - 1$. Let $\Delta t = 2$ and $N = 40$. Select values of M , P , and \mathbf{R} so that these constraints are not violated after a unit step disturbance occurs.

20.6 For Exercise 20.1, consider two sets of design parameters and simulate unit step changes in both the disturbance and the set point. Assume that the disturbance model is identical to the process model. The design parameters are

- (a) $M = 7$ $P = 10$ $\mathbf{R} = \mathbf{0}$
- (b) $M = 3$ $P = 10$ $\mathbf{R} = \mathbf{0}$

Which controller provides the best control? Justify your answer.

20.7 Consider the unconstrained, SISO version of MPC in Eq. 20-57. Suppose that the controller is designed so that the control horizon is $M = 1$ and the weighting matrices are $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = 1$. The prediction horizon P can be chosen arbitrarily.

Demonstrate that the resulting MPC controller has a simple analytical form.

20.8 A theoretical advantage of MPC for ideal conditions is that it guarantees that both controlled and manipulated variables satisfy specified inequality constraints. Briefly discuss why this theoretical advantage may not be realized in practical applications.

20.9 In Section 20.6.1, MPC was applied to the Wood-Berry distillation column model. A MATLAB program for this example and constrained MPC is shown in Table E20.9.

The design parameters have the base case values (Case B in Fig. 20.13) except for $P = 10$ and $M = 5$. The input constraints are the saturation limits for each input (-0.15 and $+0.15$). Evaluate the effects of control horizon M and input weighting matrix \mathbf{R} by simulating the set-point change and the first disturbance of Section 20.6.1 for the following parameter values:

- (a) Control horizon, $M = 2$ and $M = 5$
 (b) Input weighting matrix, $\mathbf{R} = 0.1\mathbf{I}$ and $\mathbf{R} = \mathbf{I}$

Table E20.9 MATLAB Program (Based on a program by Morari and Ricker (1994))

```
g11=poly2tfd(12.8,[16.7 1],0,1); % model
g21=poly2tfd(6.6,[10.9 1],0,7);
g12=poly2tfd(-18.9,[21.0 1],0,3);
g22=poly2tfd(-19.4,[14.4 1],0,3);
gd1=poly2tfd(3.8,[14.9 1],0,8,1);
gd2=poly2tfd(4.9,[13.2 1],0,3,4);
tfinal=120; % Model horizon, N
delt=1; % Sampling period
ny=2; % Number of outputs
model=tf2step(tfinal,delt,ny,g11,g21,g12,g22)
plant=model; % No plant/model mismatch
dmodel=[] % Default disturbance model
dplant=tf2step(tfinal,delt,ny,gd1,gd2)
P=10; M=5; % Horizons
ywt=[1 1]; uwt=[0.1 0.1]; % Q and R
tend=120; % Final time for simulation
r=[0 1]; % Set-point change in XB
a=zeros([1,tend]);
for i=51:tend
    a(i)=0.3*2.45; % 30 % step in F at t=50 min.
end
dstep=[a'];
ulim=[-.15 -.15 .15 .15 1000 1000]; % u limits
ylim=[]; % No y limits
tfilter=[];
[y1,u1]=cmpec(plant,model,ywt,uwt,M,P,tend,r,
ulim,ylim,tfilter,dplant,dmodel,dstep);
figure(1)
subplot(211)
plot(y1)
legend('XD','XB')
xlabel('Time (min)')
subplot(212)
stairs(u1) % Plot inputs as staircase functions
legend('R','S')
xlabel('Time (min)')
```

Consider plots of both inputs and outputs. Which choices of M and \mathbf{R} provide the best control? Do any of these MPC controllers provide significantly better control than the controllers shown in Figs. 20.13 and 20.14? Justify your answer.

20.10 Design a model predictive controller for the process

$$G_p(s) = \frac{e^{-6s}}{10s + 1} \quad G_v = G_m = 1$$

Select a value of N based on 95% completion of the step response and $\Delta t = 2$. Simulate the closed-loop response for a set-point change using the following design parameters:

- (a) $M = 1$ $P = 7$ $\mathbf{R} = \mathbf{0}$
 (b) $M = 1$ $P = 5$ $\mathbf{R} = \mathbf{0}$
 (c) $M = 4$ $P = 30$ $\mathbf{R} = \mathbf{0}$

20.11 Repeat Exercise 20.9 for the situation where the input constraints have been changed to -0.3 and $+0.3$.

20.12 Consider the PCM furnace module of Appendix E with the following variables (HC denotes hydrocarbon):

CVs: HC exit temperature T_{HC} and oxygen exit concentration c_{O_2}

MVs: fuel gas flow rate F_{FG} and air flow rate F_A

DVs: HC flow rate F_{HC}

Do the following, using the transfer function models given below:

- (a) Design an MPC system using the following design parameters: $\Delta t = 1$ min, $\mathbf{Q} = \text{diagonal } [0.1, 1]$, $\mathbf{R} = \text{diagonal } [0.1, 0.1]$, $P = 20$, and $M = 1$.
 (b) Repeat part (a) for the same design parameters, but where $\mathbf{R} = \text{diagonal } [0.5, 0.5]$.
 (c) Simulate the two MPC controllers for a step change in the c_{O_2} set point to 1.0143 mol/m^3 at $t = 10$ min.
 (d) Repeat part (c) for a step change in F_{HC} at $t = 10$ min to $0.035 \text{ m}^3/\text{min}$.
 (e) Based on your results for parts (c) and (d), which MPC controller is superior? Justify your answer.

Process transfer function matrix:

$$T_{HC} \begin{matrix} F_{FG} & F_A \\ \frac{220 e^{-2s}}{6.5 s + 1} & \frac{-13 e^{-2s}}{6.2 s + 1} \end{matrix}$$

$$c_{O_2} \begin{matrix} \frac{-2.0 e^{-4s}}{3.8 s + 1} & \frac{0.14 e^{-4s}}{4.2 s + 1} \end{matrix}$$

20.13 Repeat Exercise 20.12 for $\mathbf{R} = \text{diagonal } [0.1, 0.1]$ and:

- (i) $M = 1$ and (ii) $M = 4$.