

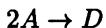
9. Application Case Studies

In this chapter, five application case studies are considered, consisting of various chemical and biochemical processes. These examples are chosen to reflect the experience of the authors in the identification and control of Volterra models, and they also reflect a range of challenging nonlinear dynamical systems. The tools from the preceding eight chapters will be explored in these case studies, highlighting their relevance and effectiveness for the particular problem. These include identification, analysis (stability), and controller synthesis.

9.1 Case study I – Van de Vusse CSTR

9.1.1 Process description

The first application considered involves an isothermal reactor in which the Van de Vusse reaction kinetic scheme:



is carried out. The mass balances for components A and B are given by:

$$\dot{C}_A = -k_1 C_A - k_3 C_A^2 + \frac{F}{V} (C_{Af} - C_A)$$

$$\dot{C}_B = k_1 C_A - k_2 C_B - \frac{F}{V} C_B$$

$$y = C_B.$$

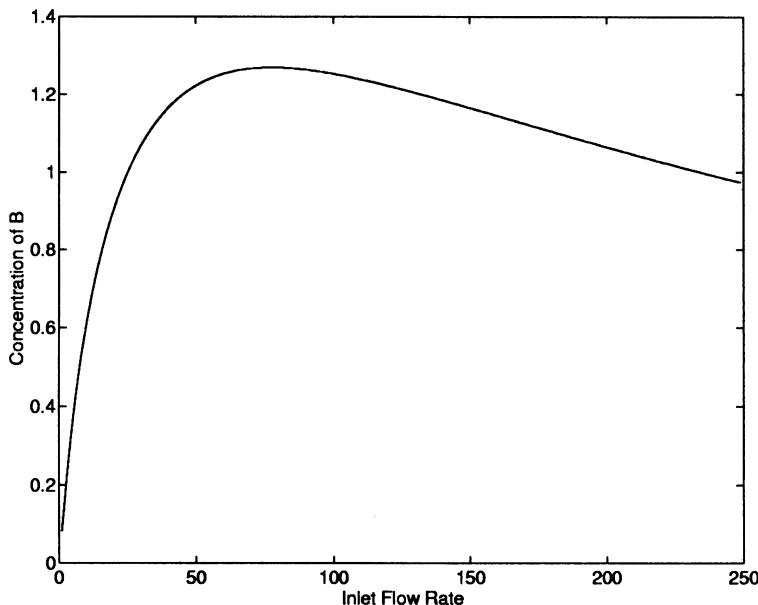
The control problem focuses on regulating the concentration of component B by manipulating the inlet flow rate. This example has been considered by a number of researchers as a benchmark problem for evaluating nonlinear process control algorithms (Van de Vusse, 1964; Kantor, 1986; Doyle III, 1991). A plot of the system's operating locus (Figure 9.1) reveals some of the interesting behavior displayed by this simple system. The reactor exhibits a change in steady-state gain at peak conversion level, nonminimum phase behavior to the left of this peak, and minimum phase behavior on the right.

By normalizing the process variables around the following operating point:

F. J. Doyle III et al., *Identification and Control Using Volterra Models*

Doyle, F. J., et al. (*Identification and Control Using Volterra Models*). Springer London, Limited, 2002. ProQuest Ebook Central,
<http://ebookcentral.proquest.com/lib/newcastle/detail.action?docID=3075019>.

Created from newcastle on 2019-04-30 18:17:20.

**Fig. 9.1.** Operating locus of Van de Vusse reactor**Table 9.1.** Kinetic parameters

| | |
|-----------|--|
| k_1 | 50 h^{-1} |
| k_2 | 100 h^{-1} |
| k_1 | $10 \text{ l mol}^{-1} \text{ h}^{-1}$ |
| C_{A_f} | 10 mol l^{-1} |
| V | 1 l |

$$C_{A0} = 3.0$$

$$C_{B0} = 1.12$$

$$\frac{F_0}{V} = 34.3$$

and substituting the values for the physical constants (Table 9.1), the process model becomes:

$$\dot{x}_1 = -50x_1 - 10x_1^2 + u(10 - x_1)$$

$$\dot{x}_2 = 50x_1 - 100x_2 + u(-x_2)$$

$$y = x_2$$

where the deviation variable for the concentration of component A is denoted by x_1 , the concentration of component B by x_2 , and the inlet flow rate by u .

9.1.2 Volterra model-based IMC design

The Carleman linearization (Rugh, 1981) of this nonlinear model yields the following bilinear approximation, where $\tilde{\mathbf{z}} = (z_1 \ z_2 \ z_1 z_2 \ z_1^2 \ z_2^2)^T$:

$$\dot{\tilde{\mathbf{z}}} = \mathbf{A}\tilde{\mathbf{z}} + \mathbf{N}\tilde{\mathbf{z}}u + \mathbf{bu}$$

$$\tilde{y} = \mathbf{c}^T \tilde{\mathbf{z}}$$

The matrices in this equation are given by:

$$\mathbf{A} = \begin{bmatrix} -144.3 & 0 & 0 & -10 & 0 \\ 50 & -134.3 & 0 & 0 & 0 \\ 0 & 0 & -278.6 & 50 & 0 \\ 0 & 0 & 0 & -288.6 & 0 \\ 0 & 0 & 100 & 0 & -268.6 \end{bmatrix}$$

$$\mathbf{N} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ -1.12 & 7 & -2 & 0 & 0 \\ 14 & 0 & 0 & -2 & 0 \\ 0 & -2.24 & 0 & 0 & -2 \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} 7 \\ -1.12 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{c}^T = [0 \ 1 \ 0 \ 0 \ 0].$$

The control synthesis procedure from Chapter 6 is employed, using this bilinear approximation. An IMC controller is designed, with the following realization of the plant inverse augmented with a first-order filter (time constant = 0.01 h):

$$\mathbf{Q}_1 = \frac{89.28s^2 + 24880s + 1.730 \times 10^6}{s^2 + 268.2s + 16820}$$

For comparison purposes, a linear model-based control scheme is also considered. In this case, however, a decomposition of the original linear system approximation into allpass and minimum phase factors must first be done. The linear controller is designed for the minimum phase factor, leaving the allpass factor and the filter in the nominal design.

Simulation results—open-loop. Figures 9.2 and 9.3 show the open-loop responses of the three different system models to a step change of +15 and -20 in the inlet flow rate respectively. The three models show a range of responses for the concentration of component B ; however, the second-order Volterra model shows a closer correspondence to the true nonlinear response than does the linear model. All three models display the inverse response associated with a nonminimum phase system, and all three models display the same sign of the process gain.

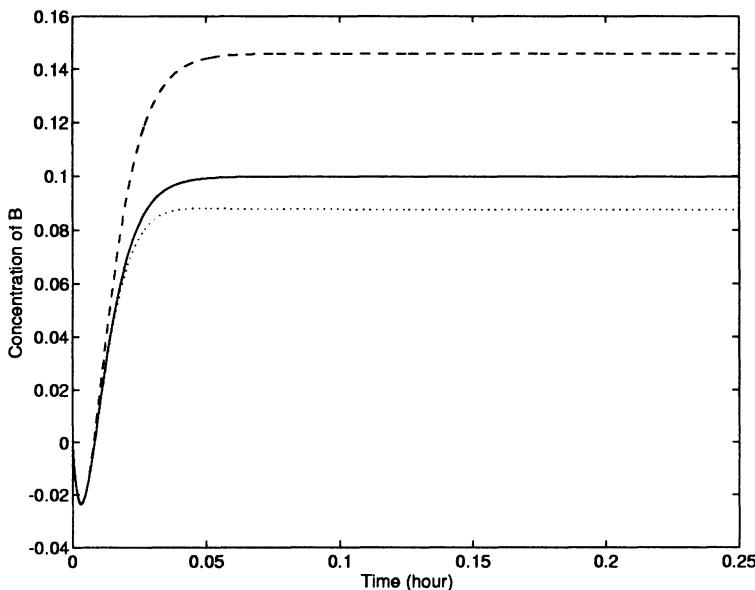


Fig. 9.2. Open-loop simulations of Van de Vusse reactor for step inputs of small magnitude (linear [dashed], second-order Volterra [dotted], and original nonlinear [solid])

Simulation results—closed-loop. Figure 9.4 shows the closed-loop response of the system to a 0.1 step increase in the setpoint of C_B . The solid line is the *ideal* system response, i.e. the response of the product of the linear all-pass factor and the first-order filter, to a step input. The dashed line is the response under linear model-based control and the dotted line is the response under second-order Volterra model-based control. The Volterra model-based system is clearly closer to the reference signal, whereas the linear model-based system is more sluggish.

Figure 9.5 shows the closed loop response to a -0.5 change in the set-point of C_B , further highlighting the difference between the two controllers. The Volterra model-based controller shows slight undershoot before return-

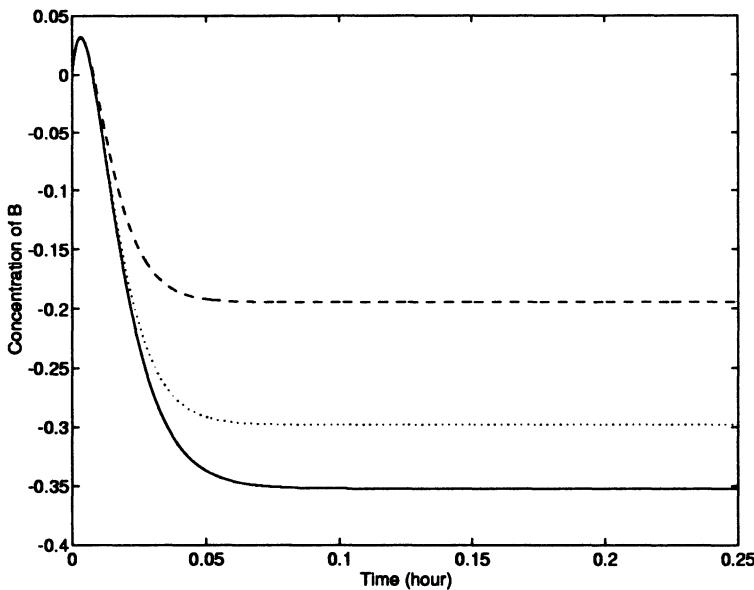


Fig. 9.3. Open-loop simulations of Van de Vusse reactor for step inputs of large magnitude (linear [dashed], second-order Volterra [dotted], and original nonlinear [solid])

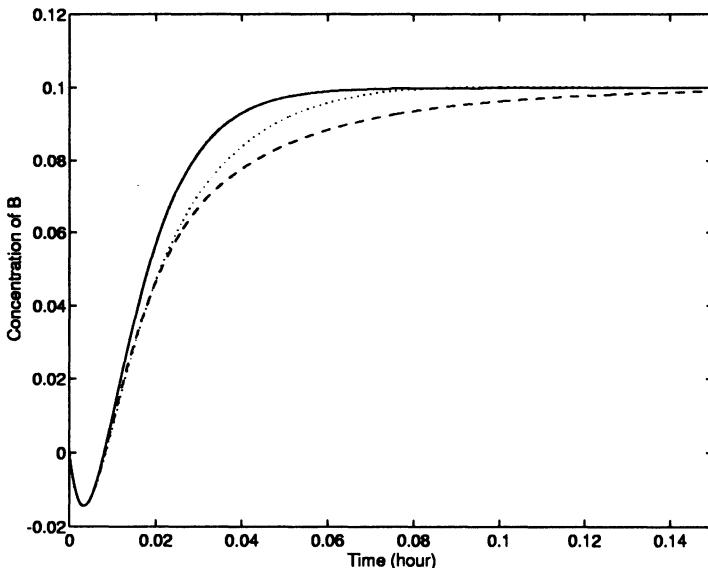


Fig. 9.4. Closed-loop simulation of Van de Vusse reactor for a +10% step in the setpoint (linear [dashed], second-order Volterra [dotted], and ideal response [solid])

ing quickly to the new setpoint; the linear model-based system shows greater undershoot and a slowly oscillating return to the setpoint.

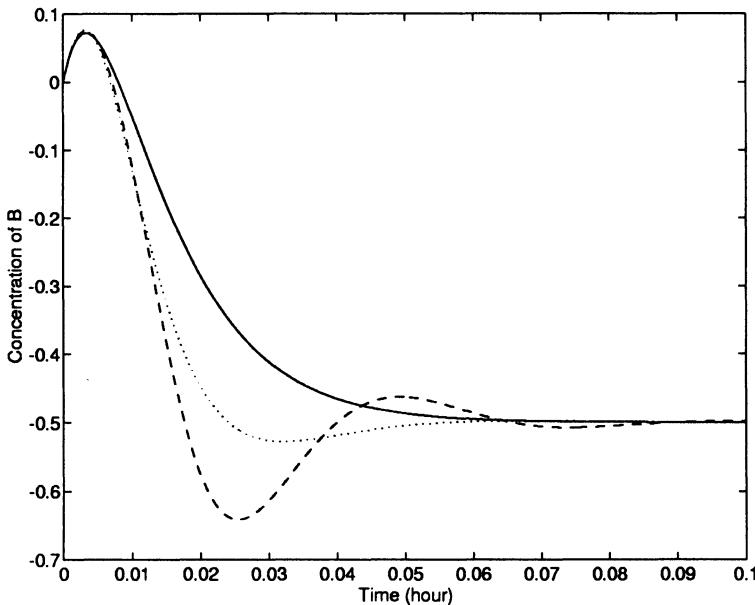


Fig. 9.5. Closed-loop simulation of Van de Vusse reactor for a -50% step in the setpoint (linear [dashed], second-order Volterra [dotted], and ideal response [solid])

9.2 Case study II – isothermal polymerization reactor

9.2.1 Process description

In this section we consider the application of the proposed controller design strategy to a jacketed CSTR in which a polymerization reaction is taking place. The reaction under consideration is the free-radical polymerization of MMA with azo-bis-isobutyronitrile (AIBN) as initiator and toluene as solvent. Details of the rate laws for free-radical polymerization were obtained from Congalidis et al. (1989), Daoutidis et al. (1990), and Ray (1972). For this particular case study, we make the following simplifying assumptions:

- isothermal operation
- perfect mixing
- constant heat capacity
- no polymer in the inlet stream
- no gel effect

- constant reactor volume
- negligible initiator flow rate (cf. monomer flow rate)
- quasi-steady state and long-chain hypothesis.

Under these assumptions, the polymer reactor dynamics can be described by the following model:

$$\begin{aligned}\dot{x}_1 &= - \left[Z_P \exp \left(\frac{-E_P}{RT} \right) + Z_{f_m} \exp \left(\frac{-E_{f_m}}{RT} \right) \right] x_1 P_0(x_2, T) - \frac{Fx_1}{V} + \frac{FC_{m_{in}}}{V} \\ \dot{x}_2 &= -Z_I \exp \left(\frac{-E_I}{RT} \right) x_2 - \frac{Fx_2}{V} + \frac{F_I C_{I_{in}}}{V} \\ \dot{x}_3 &= \left[0.5 Z_{T_c} \exp \left(\frac{-E_{T_c}}{RT} \right) + Z_{T_d} \exp \left(\frac{-E_{T_d}}{RT} \right) \right] P_0^2(x_2, T) \\ &\quad + Z_{f_m} \exp \left(\frac{-E_{f_m}}{RT} \right) x_1 P_0(x_2, T) - \frac{Fx_3}{V} \\ \dot{x}_4 &= M_m \left[Z_P \exp \left(\frac{-E_P}{RT} \right) + Z_{f_m} \exp \left(\frac{-E_{f_m}}{RT} \right) \right] x_1 P_0(x_2, T) - \frac{Fx_4}{V}\end{aligned}$$

where:

$$P_0(x_2, T) = \left[\frac{2f^* x_2 Z_I \exp \left(\frac{-E_I}{RT} \right)}{Z_{T_d} \exp \left(\frac{-E_{T_d}}{RT} \right) + Z_{T_c} \exp \left(\frac{-E_{T_c}}{RT} \right)} \right]^{0.5}$$

The dimensionless state variables in the above equations are defined as follows: $x_1 = C_m$, $x_2 = C_I$, $x_3 = D_0$, $x_4 = D_1$, where x_1 is the monomer concentration C_m , x_2 is the initiator concentration C_I , x_3 and x_4 are the first and zeroth moments of the molecular weight distribution respectively, and $\frac{D_1}{D_0}$ is the NAMW (ratio of the first and zeroth moments). The control problem focuses on manipulating the volumetric flowrate of the initiator ($u = F_I$) to regulate the NAMW ($y = \frac{D_1}{D_0}$). The operating conditions chosen for this study are as follows: $x_{10} = 5.50677$, $x_{20} = 0.132906$, $x_{30} = 0.0019752$, $x_{40} = 49.3818$, $u_0 = 0.016783$, $y_0 = 25,000$.

Introducing the values in Tables 9.2 and 9.3 for the physical constants in the modeling equations (and assuming a constant $C_{m_{in}}$) yields:

$$\begin{aligned}\dot{x}_1 &= 10(6 - x_1) - 2.4568x_1\sqrt{x_2} \\ \dot{x}_2 &= 80u - 10.1022x_2 \\ \dot{x}_3 &= 0.0024121x_1\sqrt{x_2} + 0.112191x_2 - 10x_3 \\ \dot{x}_4 &= 245.978x_1\sqrt{x_2} - 10x_4 \\ y &= \frac{x_4}{x_3}. \tag{9.1}\end{aligned}$$

The overall process is depicted schematically in Figure 9.6.

Table 9.2. Kinetic parameters

| i | Z_i | E_i |
|--------------|---|--------------------------------|
| T_c | 3.8223E10 $\text{m}^3 \text{ kmol}^{-1} \text{ h}^{-1}$ | 2.9442E3 kJ kmol^{-1} |
| T_d | 3.1457E11 $\text{m}^3 \text{ kmol}^{-1} \text{ h}^{-1}$ | 2.9442E3 kJ kmol^{-1} |
| I | 3.7920E18 h^{-1} | 1.2550E5 kJ kmol^{-1} |
| P | 1.7700E9 $\text{m}^3 \text{ kmol}^{-1} \text{ h}^{-1}$ | 1.8283E4 kJ kmol^{-1} |
| f_m | 1.0067E15 $\text{m}^3 \text{ kmol}^{-1} \text{ h}^{-1}$ | 7.4478E4 kJ kmol^{-1} |
| $f^* = 0.58$ | | |

Table 9.3. System parameters

| | |
|-------------------|--------------------------------------|
| $F = 1.00$ | m^3 |
| $V = 0.1$ | m^3 |
| $C_{I,in} = 8.0$ | kmol m^{-3} |
| $y_{sp} = 25,000$ | kg kmol^{-1} |
| $F_I = 0.016783$ | $\text{m}^3 \text{ h}^{-1}$ |
| $R = 8.314$ | $\text{kJ kmol}^{-1} \text{ K}^{-1}$ |
| $M_m = 100.12$ | kg kmol^{-1} |
| $C_{m,in} = 6.0$ | kmol m^{-3} |
| $T = 335$ | K |

9.2.2 Volterra IMC design

From this first principles reactor model, we now proceed to obtain an approximate Volterra model via the Carleman linearization approach. By defining the normalized variables $z_i = \frac{x_i - x_{i0}}{x_{i0}}$, where x_{i0} is the nominal operating value for x_i , we obtain a model with variables that are zero at the operating condition and have roughly equivalent ranges. Now, if we take a Taylor series approximation of this normalized model, and retain up to the second-order terms, we obtain:

$$\begin{aligned}\dot{z}_1 &= -10.8957z_1 - 0.447837z_2 - 0.447837z_1z_2 + 0.111959z_2^2 \\ \dot{z}_2 &= -10.1022z_2 + 10.1022\tilde{u} \\ \dot{z}_3 &= 2.45162z_1 + 8.7744z_2 - 10z_3 + 1.22581z_1z_2 - 0.306453z_2^2 \\ \dot{z}_4 &= 10z_1 + 5.00001z_2 - 10z_4 + 5.00001z_1z_2 - 1.25z_2^2 \\ \tilde{y} &= -z_3 + z_4 - z_3z_4 + z_3^2.\end{aligned}$$

The Carleman linearization of this nonlinear system yields the following bilinear approximation where $\tilde{\mathbf{z}} = (z_1 \ z_2 \ z_3 \ z_4 \ z_1^2 \ z_1z_2 \ z_1z_3 \ z_1z_4 \ z_2^2 \ z_2z_3 \ z_2z_4 \ z_3^2 \ z_3z_4)^T$:

$$\dot{\tilde{\mathbf{z}}} = \mathbf{A}\tilde{\mathbf{z}} + \mathbf{N}\tilde{\mathbf{z}}u + \mathbf{b}u$$

$$\tilde{\mathbf{y}} = \mathbf{c}^T \tilde{\mathbf{z}}$$

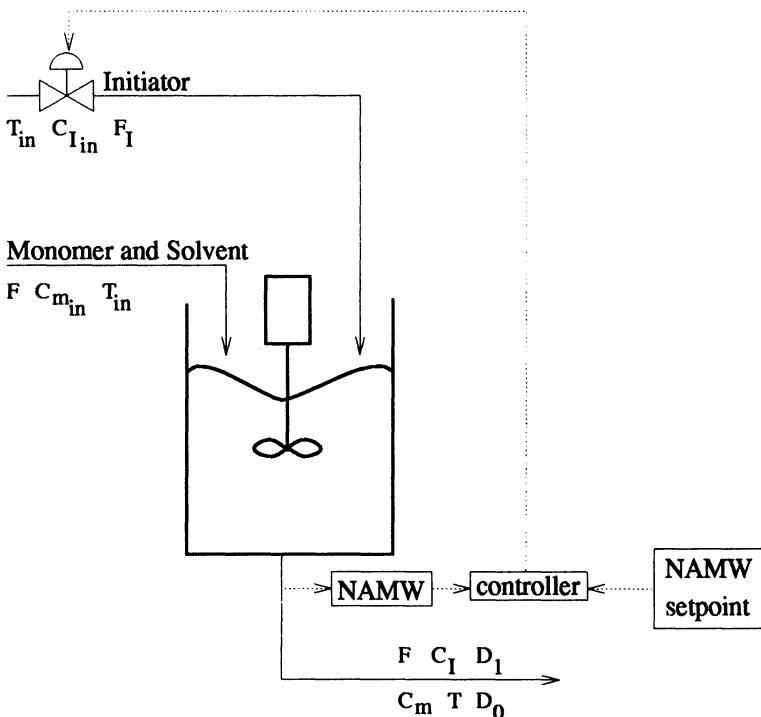


Fig. 9.6. Process schematic for case study II

The matrices in this equation are detailed in Doyle III et al. (1995).

Controller synthesis. From the bilinear model of the polymerization reactor, it is relatively straightforward to derive the linear process controller as well as the second-order correction described in Chapter 6.

Using the matrices given above, the linear reactor model can be described by the following frequency domain Volterra kernel (Laplace transfer function):

$$P_1(s) = c_1^T (sI - A_{11})^{-1} b_1 \quad (9.2)$$

where A_{11}, b_1, c_1 is the subsystem of the bilinear model corresponding to the first four (linear) states in \tilde{z} . The second order portion of the model is described by the following frequency domain Volterra kernels:

$$P_2(s_1, s_2) = c^T ((s_1 + s_2)I - A)^{-1} N(s_1 - A)^{-1} b$$

which can be realized in the time domain by the following expression (Al-Baiyat, 1986):

$$\begin{aligned} \dot{x}_1 &= Ax_1 + bu \\ \dot{x}_2 &= Ax_2 + N(x_1 * u) \\ y &= cx_2 \end{aligned}$$

From this information, a standard controller, denoted \mathbf{Q}_1 , can be designed for \mathbf{P}_1 using any appropriate method. In this study, we employ an IMC controller where \mathbf{Q}_1 is chosen as the inverse of \mathbf{P}_1 , and augmented by a linear filter

$$F(s) = \frac{1}{(0.5s + 1)^2}$$

to yield the causal controller:

$$\frac{-0.1056s^4 - 4.324s^3 - 66.33s^2 - 452.0s - 1155}{s^4 + 25.79s^3 + 209.1s^2 + 5588s + 471.6}.$$

The choice of filter reflects the guidelines in Morari and Zafiriou (1989) that recommend a single tuning knob, and a realizable control structure.

Following the procedure outlined in Chapter 6, a second-order controller is realized as shown in Figure 6.3. Note that the linear filter F must be implemented *before* the second-order correction is fed back. Structurally this is equivalent to implementing a controller as in Figure 6.3 with the linear plant inverse augmented with a linear filter, and the second-order correction P_2 augmented with the inverse of the linear filter. The causality of this latter operator is not strictly guaranteed, as the relative degree or pole-zero excess of the second-order Volterra kernel is not always equal to the relative degree of the linear Volterra kernel. It is always possible to correct for this by introducing a second linear filter F_2 cascaded with the second-order plant model term.

Observe that the controller has been synthesized using only an optimal linear controller (generated from the linear inverse) and the second-order Volterra kernel for the reactor model. The continuous-time controller is discretized numerically in solving for the closed-loop responses that are described below.

Simulation results—closed-loop. The nonlinear nature of the polymerization reactor under investigation is revealed in the open-loop simulations in Figures 9.7 and 9.8. Figure 9.7 shows the response of the full nonlinear model to step changes in the initiator flow rate that range $\pm 20\%$. As can be seen in the figure, all three curves overlap for small input changes, but as the magnitude of the change increases the nonlinear curve separates from the other two curves. However, the second-order Volterra model tracks the true nonlinear response more accurately than the linear model.

Results from a similar simulation are displayed in Figure 9.8, where now the magnitude of the input step change has been increased to $\pm 90\%$. Although these changes may appear unrealistic, they do highlight the relative accuracy of the linear and second-order model approximations. In this case, there is a significantly larger difference between the true nonlinear response and both the linear and second-order model responses. And as before, the second-order response tracks the full nonlinear response more closely than the linear response.

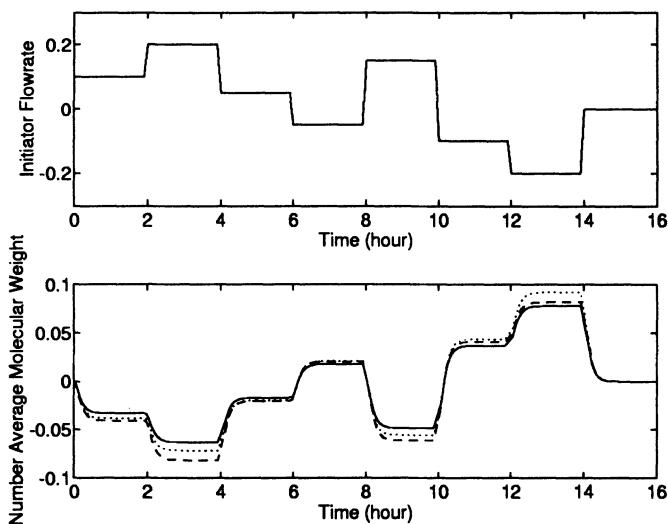


Fig. 9.7. Open-loop simulations of polymerizer reactor for step inputs of small magnitude (solid line—Equation (9.1)), the second-order bilinear approximation (dotted line—Equation (9.2)), and the linear approximation (dashed line—Equation (9.2))

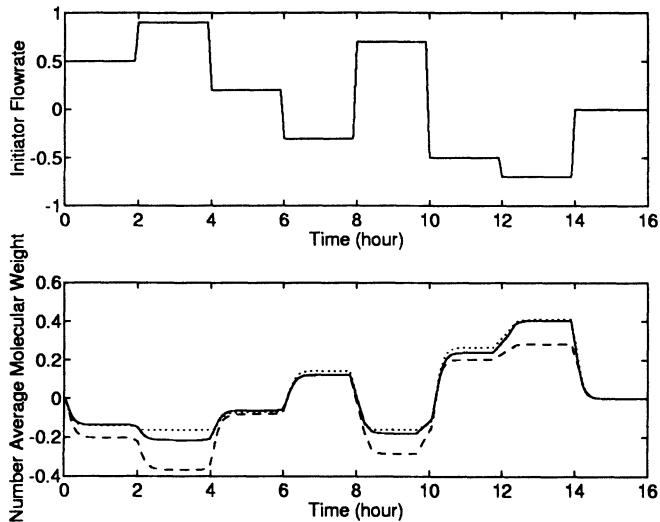


Fig. 9.8. Open-loop simulations of polymerizer reactor for step inputs of large magnitude (solid line—Equation (9.1)), the second-order bilinear approximation (dotted line—Equation (9.2)), and the linear approximation (dashed line—Equation (9.2))

Figure 9.9 shows the response of the closed-loop system to a -10% step change in the setpoint. For this set-point change, the nonlinear nature of the plant is revealed as the closed-loop response deviates from the ideal linear trajectory. Though both controllers produce qualitatively similar responses, the second-order controller has reduced the integral squared tracking error, relative to the linear controller, by approximately 95%.

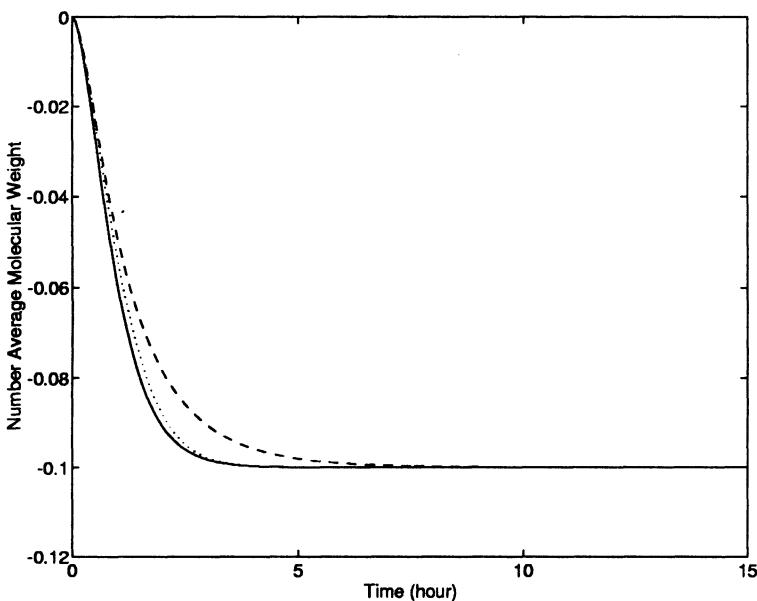


Fig. 9.9. Closed-loop simulation of polymerizer reactor for a -10% step in the set-point (linear controller [dashed line]; linear controller plus second-order correction (Volterra controller) [dotted line]; and reference trajectory (linear filter response) [solid line])

The same simulation is repeated for a larger step change ($r = \frac{0.5}{s}$) and the results are displayed in Figure 9.10. The same three curves are plotted and similar trends are evident. In this case, the larger input magnitude has led to a greater separation between the linear and second-order responses, with the second-order response tracking the reference trajectory more closely.

A quantitative summary of the response to a number of setpoint changes for three different reactor plant descriptions is given in Table 9.4. Shown here are the magnitudes of the ISE between the true output and the reference trajectory. In this case, the reactor plant employed in the simulation is the full nonlinear plant model given in Equation (9.1).

Another set of informative simulations is summarized in Table 9.5. In this case we examine the robust performance of the two closed-loop systems (linear controller, second-order controller). The specific uncertainty considered

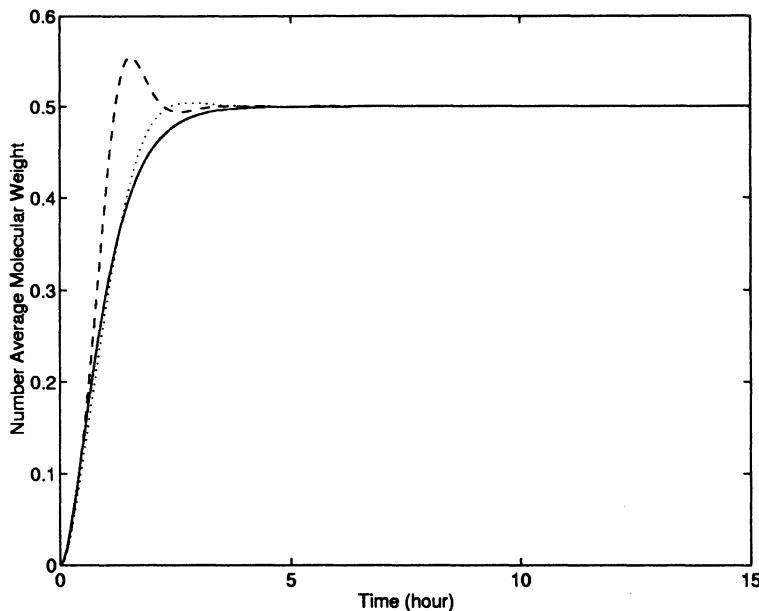


Fig. 9.10. Closed-loop simulation of polymerizer reactor for a 10% step in the set-point (linear controller [dashed line]; linear controller plus second-order correction (Volterra controller) [dotted line]; and reference trajectory (linear filter response) [solid line])

Table 9.4. Closed-loop performance—nominal nonlinear plant

| Input Signal | Tracking Error | |
|---------------|-------------------|---------------------|
| | Linear Controller | Volterra Controller |
| $r = 0.1/s$ | 6.7759E-5 | 0.1398E-5 |
| $r = 0.25/s$ | 2.4136E-3 | 0.1988E-3 |
| $r = 0.5/s$ | 2.6864E-2 | 2.1912E-2 |
| $r = -0.05/s$ | 4.6581E-6 | 0.0457E-6 |
| $r = -0.1/s$ | 7.7548E-5 | 0.4076E-5 |
| $r = -0.15/s$ | 4.0764E-4 | 0.7313E-4 |

was a -15% error in the process gain. The ISE between the output and the reference trajectory is compiled. These results confirm the superior performance of the second-order controller under a situation of nonlinear model mismatch.

In both cases considered above (nominal nonlinear plant and perturbed nonlinear plant) it can be seen that the second-order controller consistently outperforms the linear controller by a margin of approximately 20 to 95%.

Table 9.5. Closed-loop performance-perturbed nonlinear plant

| Input Signal | Tracking Error | |
|---------------|-------------------|---------------------|
| | Linear Controller | Volterra Controller |
| $r = 0.1/s$ | 1.0566E-5 | 9.3754E-5 |
| $r = 0.25/s$ | 8.2126E-4 | 3.0921E-4 |
| $r = 0.5/s$ | 2.0392E-2 | 0.1098E-2 |
| $r = -0.1/s$ | 4.7520E-5 | 1.7597E-5 |
| $r = -0.25/s$ | 3.3347E-4 | 0.3865E-4 |
| $r = -0.5/s$ | 1.1838E-3 | 0.0071E-3 |

9.2.3 Volterra-Laguerre IMC design

In discrete state-space, the Volterra-Laguerre model has the form:

$$l(k+1) = Al(k) + Bu(k)$$

$$\hat{y}(k) = C^T l(k) + l(k)^T D l(k) + \sum_{p=1}^L (l(k)^T E_p l(k)) l_p(k) + \dots$$

Let

$$\hat{y} = \tilde{H}[u]$$

denote this representation. Note that this model can be partitioned as:

$$\tilde{H} = \tilde{H}_1 + \tilde{H}_2 + \tilde{H}_3 + \dots$$

where \tilde{H}_1 is the linear operator, \tilde{H}_2 is the quadratic operator, and \tilde{H}_3 is the cubic operator, which are given by:

$$\tilde{H}_1[u] = \begin{cases} l(k+1) = Al(k) + Bu(k) \\ y(k) = C^T l(k) \end{cases}$$

$$\tilde{H}_2[u] = \begin{cases} l(k+1) = Al(k) + Bu(k) \\ \hat{y}(k) = l(k)^T D l(k) \end{cases}$$

$$\tilde{H}_3[u] = \begin{cases} l(k+1) = Al(k) + Bu(k) \\ \hat{y}(k) = \sum_{p=1}^L (l(k)^T E_p l(k)) l_p(k) \end{cases}$$

For the reactor process currently under consideration, the linear portion of the Volterra-Laguerre model is given in the following equation. With the **A**, **B**, and **C** matrices defined earlier in this section, the z-transform of the linear portion of the Volterra-Laguerre model is given by:

$$G(z) = \frac{-0.1952z^2 + 0.1327z + 0.007303}{z^3 - 1.8974z^2 + 1.2000z - 0.2530}.$$

which can be factored into zero-pole-gain form as:

$$G(z) = \frac{-0.1952(z - 0.7312)(z + 0.05118)}{(z - 0.6325)^3},$$

Multiplying the numerator and denominator by z^{-3} and simplifying yields:

$$G(z) = \frac{-0.1952z^{-1}(1 - 0.7312z^{-1})(1 + 0.05118z^{-1})}{(1 - 0.6325z^{-1})^3}.$$

Factoring out the time delay gives:

$$G_-(z) = \frac{-0.1952(1 - 0.7312z^{-1})(1 + 0.05118z^{-1})}{(1 - 0.6325z^{-1})^3}.$$

Then

$$\begin{aligned}\tilde{Q}(z) &= \frac{1}{G_-(z)} \\ &= \frac{(1 - 0.6325z^{-1})^3}{-0.1952(1 - 0.7312z^{-1})(1 + 0.05118z^{-1})}.\end{aligned}$$

To improve robustness, this controller should be augmented with a filter:

$$F(z) = \frac{1 - \alpha}{1 - \alpha z^{-1}}.$$

Here, α is the filter constant, which ranges from zero to one. Thus, the final controller is given by:

$$\begin{aligned}Q(z) &= F(z)\tilde{Q}(z) \\ &= \frac{(1 - \alpha)(1 - 0.6325z^{-1})^3}{-0.1952(1 - \alpha z^{-1})(1 - 0.7312z^{-1})(1 + 0.05118z^{-1})}.\end{aligned}$$

Note that the filter constant α is the lone tuning parameter. A setting closer to zero yields a more aggressive controller, whereas a setting closer to one yields a less aggressive controller.

Simulation results—open-loop. The open-loop response of the third-order Volterra-Laguerre model, compared with that of the full nonlinear model (labeled “Actual”) and the linear and second-order Volterra-Laguerre models, is seen in Figures 9.11 through 9.13. Observe that the third-order model has improved predictive performance over both the linear and second-order models. The sum total of the individual step ISEs for the third-order Volterra-Laguerre model is given in Table 9.6. The third-order model performs 75% better than the linear model and 51% better than the second-order model.

The NLIMC scheme incorporating a third-order Volterra-Laguerre is implemented as a logical extension of the second-order development in Chapter 6. The closed-loop responses to $\pm 20\%$ step changes in setpoint (with $\alpha = 0.7$) are shown in Figures 9.14 and 9.15. Observe that the third-order controller yields better reference tracking than both the linear and second-order controllers. Tables 9.7 and 9.8 show the performance of the third-order controller in response to $\pm 20\%$ changes in setpoint for different values of α .

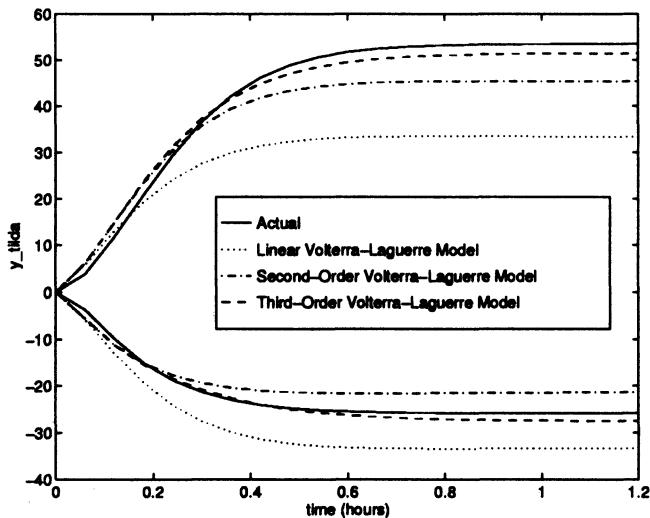


Fig. 9.11. Response to step inputs of magnitude ± 30.0

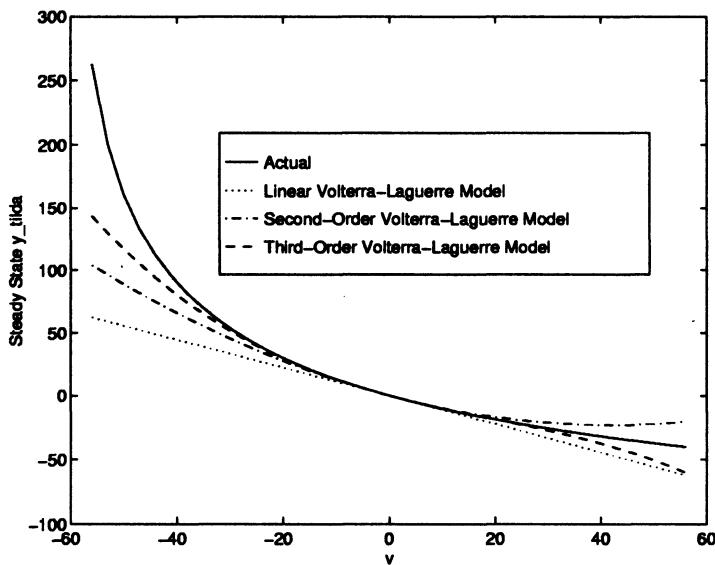


Fig. 9.12. Step response steady-state gain loci

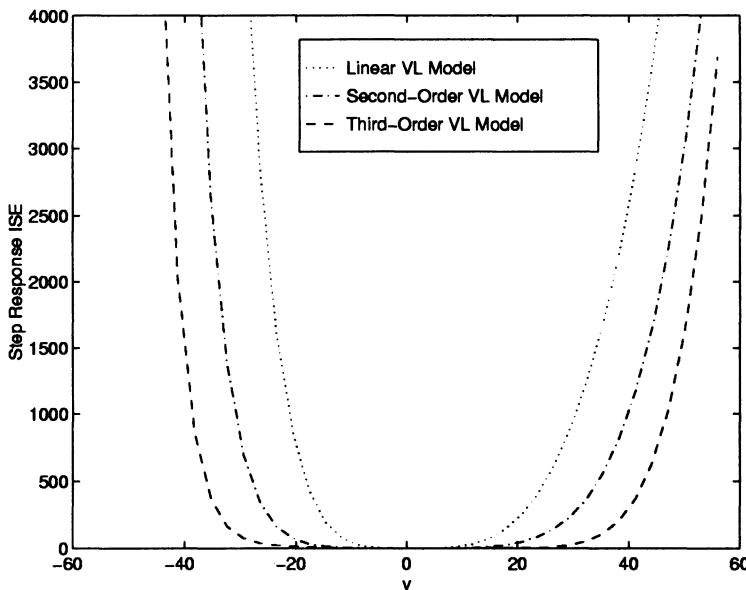


Fig. 9.13. Step response ISE loci

Table 9.6. Volterra-Laguerre model performance

| Model | Unique Parameters | Total Step ISE |
|--------------------------------|-------------------|----------------|
| Linear Volterra-Laguerre | 3 | 1.12e+06 |
| Second-Order Volterra-Laguerre | 9 | 5.63e+05 |
| Third-Order Volterra-Laguerre | 19 | 2.77e+05 |

The “Percent Improvement” column denotes the increase in accuracy for a third-order controller versus a second-order controller. As in the second-order case, as α increases, the superior performance of the third-order model becomes more evident. As α decreases, the lack of high frequency information in the third-order model causes degradation in performance. Tables 9.9 and 9.10 show the performance of the third-order controller to setpoint changes of various magnitudes with $\alpha = 0.7$. Clearly, the third-order controller is superior to the second-order controller and the performance improvement increases with larger step changes. This is due to the third-order controller’s ability to compensate for more of the system nonlinearity than the second-order model. Figures 9.16 and 9.17 show the disturbance rejection capabilities of the third-order controller, compared with the linear and second-order controllers, for $\pm 20\%$ step-changes in $C_{m_{in}}$ (with $\alpha = 0.7$). As can be seen, all three perform equally well, primarily because a model relating $C_{m_{in}}$ to the output is available for feedforward controller design.

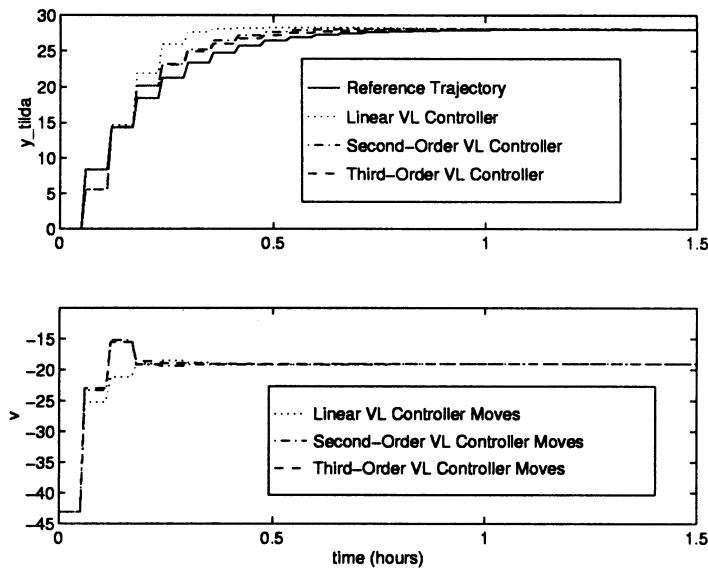


Fig. 9.14. Closed-loop response for a 20% step increase in setpoint with $\alpha = 0.7$

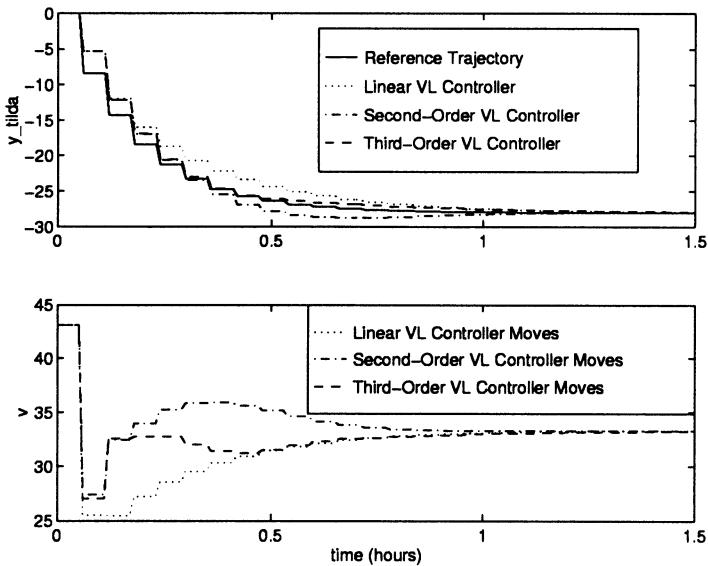


Fig. 9.15. Closed-loop response for a 20% step decrease in setpoint with $\alpha = 0.7$

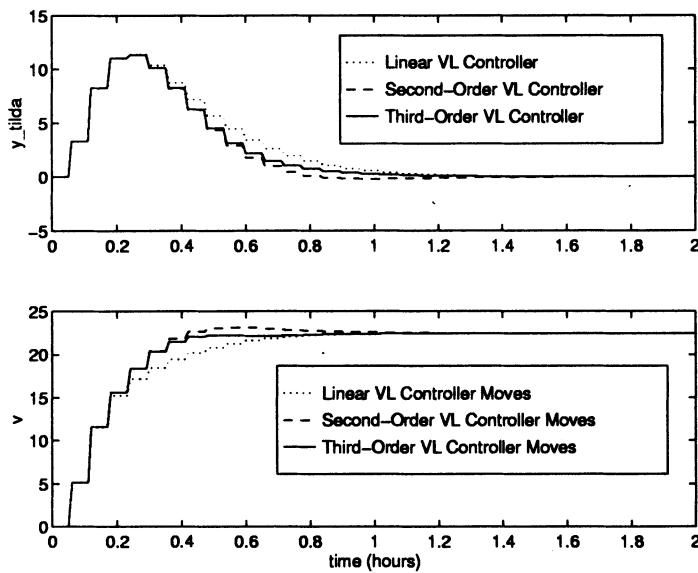


Fig. 9.16. Closed-loop response for a 20% step increase in $C_{m_{in}}$ with $\alpha = 0.7$

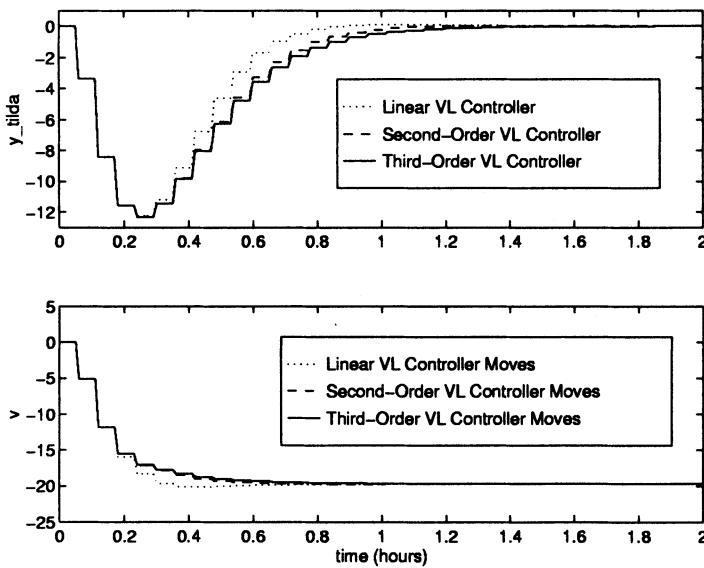


Fig. 9.17. Closed-loop response for a 20% step decrease in $C_{m_{in}}$ with $\alpha = 0.7$

Table 9.7. ISE as a function of α for a 20% step increase in setpoint

| α | Third-Order VL ISE | Percent Improvement |
|----------|--------------------|---------------------|
| 0.1 | 4.15E+02 | 0.7 |
| 0.2 | 2.52E+02 | 1.9 |
| 0.3 | 1.56E+02 | 2.5 |
| 0.4 | 9.76E+01 | 3.4 |
| 0.5 | 6.03E+01 | 5.8 |
| 0.6 | 3.63E+01 | 11.5 |
| 0.7 | 2.10E+01 | 23.1 |
| 0.8 | 1.16E+01 | 44.0 |
| 0.9 | 6.15E+00 | 70.3 |

Table 9.8. ISE as a function of α for a 20% step decrease in setpoint

| α | Third-Order VL ISE | Percent Improvement |
|----------|--------------------|---------------------|
| 0.1 | 1.10E+02 | None |
| 0.2 | 8.77E+02 | None |
| 0.3 | 6.92E+01 | 0.7 |
| 0.4 | 5.35E+01 | 2.4 |
| 0.5 | 4.02E+01 | 6.1 |
| 0.6 | 2.90E+01 | 15.2 |
| 0.7 | 2.05E+01 | 31.7 |
| 0.8 | 1.52E+01 | 57.4 |
| 0.9 | 1.65E+01 | 77.8 |

Table 9.9. ISE for positive step changes in setpoint of various magnitude with $\alpha = 0.7$

| Setpoint Change | Third-Order VL ISE | Percent Improvement |
|-----------------|--------------------|---------------------|
| +5% | 8.60E-01 | 3.0 |
| +10% | 3.82E+00 | 9.5 |
| +15% | 9.92E+00 | 16.6 |
| +20% | 2.10E+01 | 23.1 |
| +25% | 3.97E+01 | 28.5 |
| +30% | 6.97E+01 | 32.3 |
| +35% | 1.16E+02 | 34.8 |
| +40% | 1.83E+02 | 37.3 |
| +45% | 2.83E+02 | 38.5 |

9.2.4 Volterra MPC design

In order to formulate the model predictive controller, the following procedure is adopted. The variables u , x , and y are cast in deviation form and scaled by their nominal values ($\tilde{u} = \frac{u-u_0}{u_0}$, $\tilde{x}_i = \frac{x_i-x_{i0}}{x_{i0}}$, and $\tilde{y} = \frac{y-y_0}{y_0}$),

Table 9.10. ISE for negative step changes in setpoint of various magnitude with $\alpha = 0.7$

| Setpoint Change | Third-Order VL ISE | Percent Improvement |
|-----------------|--------------------|---------------------|
| -5% | 8.10E-01 | 3.6 |
| -10% | 3.32E+00 | 13.1 |
| -15% | 8.25E+00 | 25.0 |
| -20% | 2.05E+01 | 31.7 |
| -25% | 6.31E+01 | 43.7 |

and a bilinear representation of the nonlinear model was obtained using Carleman linearization (Rugh, 1981). The continuous-time bilinear system was then discretized using the explicit fourth-order Runge-Kutta method given by Equations (4.35) through (4.38) with a sampling time of 0.03 h (1.8 min). The largest negative eigenvalue of the system is -10.9 h^{-1} . The corresponding small system time constant arises from the relatively small reactor volume of 100ℓ . The model parameters are calculated using Equations (4.39) through (4.41). Linear MPC and the Volterra controller described in Section 8.4.1 were used with $m = 1$, $p = 25$, and $\lambda = 0.95$ as tuning parameters. The tuning parameters were the same for both controllers because the underlying linear models are identical. The truncation order of the model was $N = 30$. This value is chosen to obtain a model with a memory approximately equal to three times the system time constant. The plant is represented by the actual nonlinear model integrated with a fourth-order Runge-Kutta method.

Simulation results—open-loop. Open-loop responses of the nonlinear, linear, and second-order Volterra models to step changes of $\pm 0.008392 \text{ m}^3 \text{ h}^{-1}$ in F_I (from its nominal value of $0.016783 \text{ m}^3 \text{ h}^{-1}$) are shown in Figure 9.18. In both simulations, the second-order Volterra model output more closely tracks the actual nonlinear plant output. The output of the Volterra model is asymmetric for steps of the same magnitude but opposite sign, while that of the linear model is symmetric.

The result of a closed-loop simulation for a setpoint change from 25,000 to 38,000 kg kmol $^{-1}$ is shown in Figure 9.19. The reference trajectory corresponds to the closed-loop response of a nominal linear plant; consequently, deviations from this reference are the result of uncompensated *nonlinear* behavior. The Volterra controller effectively cancels the process nonlinearity to produce a linear response. In this simulation, the response obtained by the Volterra controller tracks the reference trajectory more closely than the response obtained by the linear controller. Linear MPC leads to an overshoot in the setpoint response, and this may be undesirable if product specifications require the molecular weight to be less than 38,000 kg kmol $^{-1}$. The cause of the overshoot can be attributed to the aggressive manipulated variable profile for linear MPC, which is evident in Figure 9.19. Although linear MPC could be detuned to yield an overdamped response for this setpoint

change, detuning would cause performance deterioration for setpoints below the nominal value of $25,000 \text{ kg kmol}^{-1}$. Hence, the nonlinear behavior of this process requires a compromise in the tuning of a linear model-based controller. Results for other setpoint changes are listed in Table 9.11, in terms of the relative performance of the linear controller and the second-order controller. The unstable responses are attributed to the second-order Volterra controller.

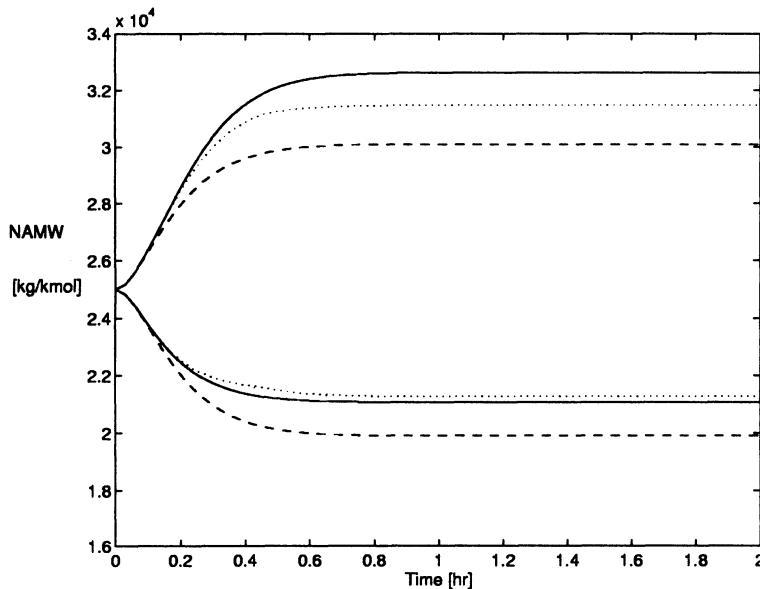


Fig. 9.18. Open-loop simulations for step changes of $\pm 0.008392 \text{ m}^3 \text{ h}^{-1}$ in F_1 from its nominal value of $0.016783 \text{ m}^3 \text{ h}^{-1}$. Solid: nonlinear; dashed: linear; dotted: Volterra

Table 9.11. Performance comparison of the linear and second-order Volterra controllers for case study II

| Setpoint kg kmol^{-1} | Percent Improvement in ISE | Maximum # of iterations |
|-----------------------------------|-------------------------------|----------------------------|
| 38,000 | 84.79 | 27 |
| 32,000 | 93.23 | 8 |
| 28,000 | 97.29 | 4 |
| 22,000 | 74.41 | 4 |
| 18,000 | unstable | NA |
| 12,000 | unstable | NA |

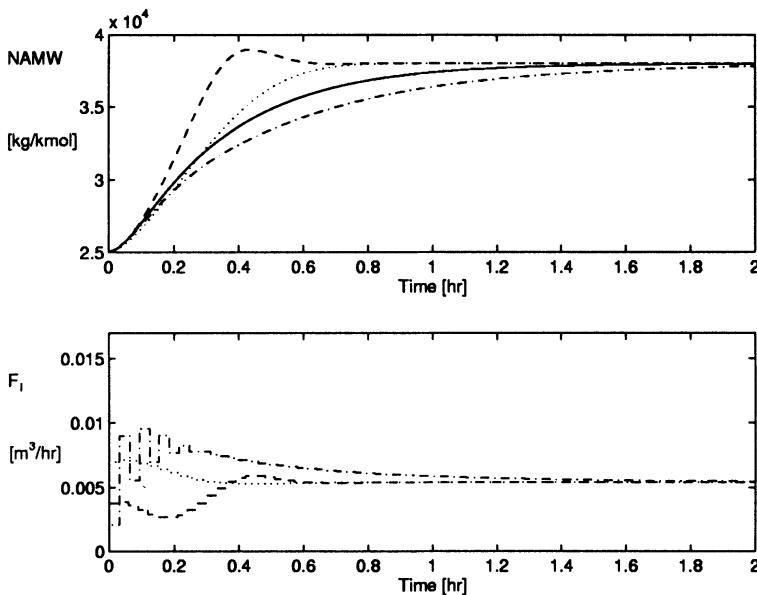


Fig. 9.19. Closed-loop simulation for a filtered step setpoint change from 25,000 to 38,000 kg kmol⁻¹. Solid: reference; dashed: linear MPC; dotted: Volterra MPC; dash dot: nonlinear QDMC

For a comparison with another nonlinear MPC strategy, the dash-dot lines in Figure 9.19 correspond to input and output profiles obtained using nonlinear QDMC (García, 1984) with $m = 1$, $p = 5$, and a first-order filter in the feedback path with filter time constant 0.94. The tuning parameters for nonlinear QDMC are different from those used for linear MPC and Volterra MPC, because the underlying linear models used by nonlinear QDMC were different from those of the other two controllers. The tuning parameters used in nonlinear QDMC resulted in an improvement in closed-loop performance compared with implementing the tuning parameters used in linear and Volterra MPC. In this implementation, the actual nonlinear model is linearized at each sampling interval, i.e. the only plant-model mismatch arises from local linearization errors, which are much smaller than the Volterra approximation errors. The output profile is much more sluggish than the output trajectories obtained using linear and Volterra MPC. Note that the initial chattering observed in the initiator flow rate (using nonlinear QDMC) indicates that this controller is tuned quite aggressively.

It is interesting to note the effect of reducing the number of second-order coefficients on the closed-loop performance, in the spirit of the model pruning strategy introduced in Section 3.6.1. Plots of the first- and second-order parameters are shown in Figures 9.20 and 9.21 respectively. Since the $b_{i,j}$ coefficients are one order of magnitude smaller than the a_i coefficients, and

\tilde{u} is on the order of 10^{-3} , the later $b_{i,j}$ terms in Figure 9.21 make a very small contribution to the model. Since $N = 30$, there are 465 $b_{i,j}$ terms. If the truncation order for the second-order parameters is reduced from $N = 30$ to $N = 15$, there would be 120 $b_{i,j}$ terms. The closed-loop performance using this reduced model is shown in Figure 9.22. As expected, the performance is very nearly the same as that shown in Figure 9.19 for the full second-order Volterra model. This model structure pruning approach corresponds to reducing the nonlinear memory of the system as noted in Section 3.6.1.

For setpoints below $18,000 \text{ kg kmol}^{-1}$ the closed-loop performance of the Volterra controller is unstable, whereas the linear model-based controller successfully brings the nonlinear plant to the new setpoint. An explanation for this behavior can be found from a plot of the steady-state gain loci (Figure 9.23). The gain of the bilinear model changes sign in the region of $y = 18,000 \text{ kg kmol}^{-1}$. This error is the result of deriving the bilinear model using local expansion that results from Carleman linearization. This technique is accurate for capturing local nonlinearities around an operating point, but can be erroneous in describing global nonlinear behavior (Doyle III et al., 1995). Since the discrete-time Volterra model was obtained from the continuous-time model, it also erroneously predicts a sign change in gain near $y = 18,000 \text{ kg kmol}^{-1}$. The Volterra series is a time-invariant operator and would need parameter adaptation if it is to be used over a larger region of operation. Several researchers (Hernández and Arkun, 1993; Ricker and Lee, 1995) have incorporated on-line parameter adaptation into nonlinear MPC schemes, and this is a promising direction for future research.

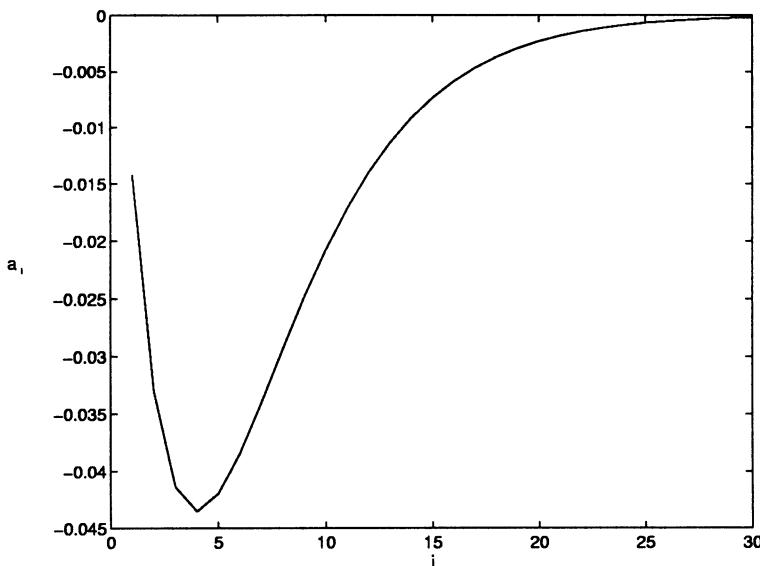


Fig. 9.20. First-order coefficients for case study II

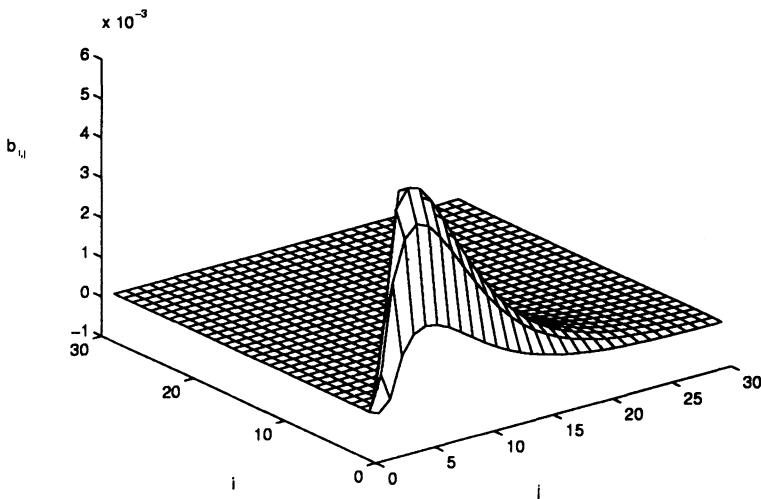


Fig. 9.21. Second-order coefficients for case study II

9.2.5 AR-Volterra MPC design

Model identification. For this case study, 500 points of input-output data were generated using a sample time of $T_s = 0.03$ h and random steps with a switching probability of $P_s = 0.05$ with the values at each transition drawn from a uniform distribution in the range $u = [0.0046, 0.028966] \text{ m}^3 \text{ h}^{-1}$. The inputs and outputs were placed in scaled, deviation form, i.e. $\tilde{u} = \frac{u-u_0}{u_0}$ and $\tilde{y} = \frac{y-y_0}{y_0}$. The regressors and model parameters were obtained using a stepwise model building algorithm (Kortmann et al., 1988). This approach employed the Akaike information criterion (Akaike, 1972) to determine the significance of previously added regressors, and to terminate structure selection. Significant regressors were detected using linear correlation coefficients as in the algorithm used by Hernández (1992). The order of the model and the number of lags on the input and output were specified to be $n = 3$, $n_u = 5$, and $n_y = 5$ respectively.

A linear model was identified using the same switching probability with values drawn from $u = [0.0151047, 0.0184613] \text{ m}^3 \text{ h}^{-1}$ which corresponded to $\pm 10\%$ of the nominal value of the input. The inputs and outputs are also cast in scaled deviation form. The regressors and model parameters are obtained using the stepwise model-building algorithm with $n = 1$, $n_u = 5$, and $n_y = 5$. The models obtained are shown in Table 9.12.

Figure 9.24 shows the change in the NAMW in response to initiator flow rates of $F_I = 0.016783 \pm 0.012183 \text{ m}^3 \text{ h}^{-1}$. These step inputs have predominantly low-frequency characteristics and demonstrate the long-term predic-

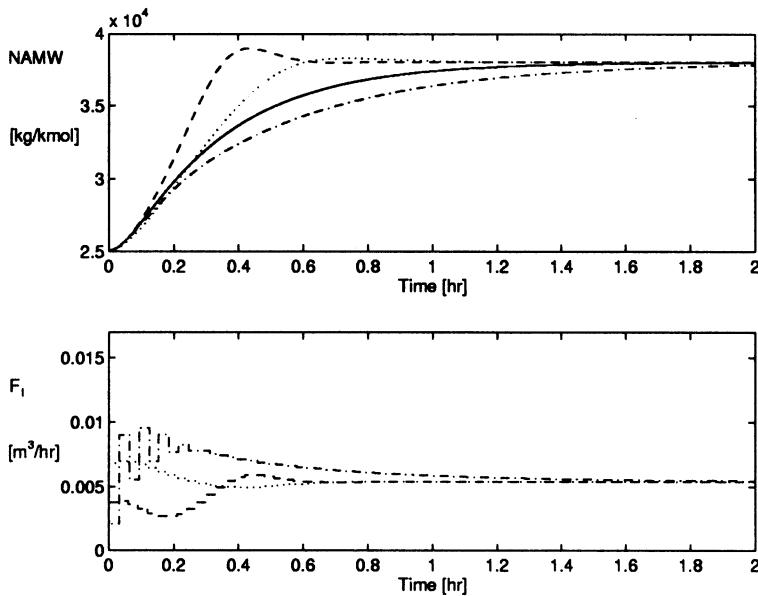


Fig. 9.22. Closed-loop simulation for a filtered step setpoint change from 25,000 to 38,000 kg kmol⁻¹ using the reduced model. Solid: reference; dashed: linear MPC; dotted: Volterra MPC; dash dot: nonlinear QDMC

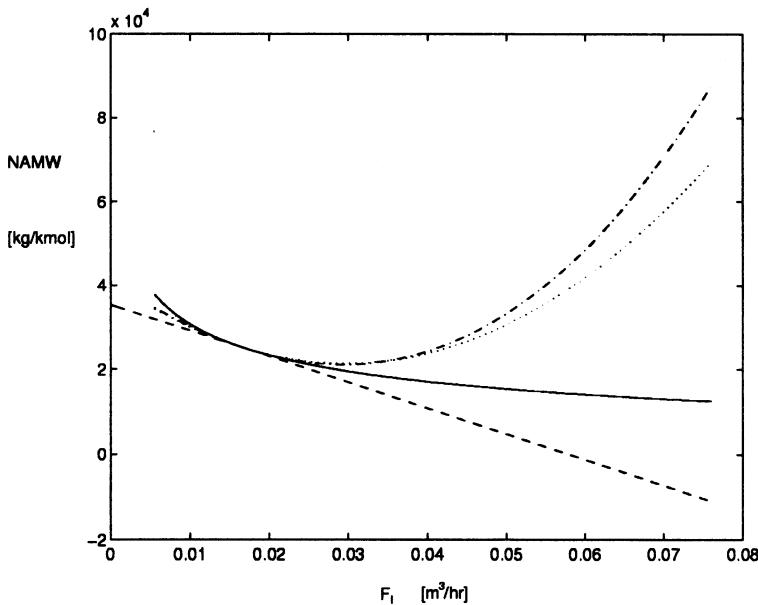


Fig. 9.23. Steady-state loci for various models. Solid: nonlinear; dashed: linear; dotted: Volterra; dash dot: bilinear

Table 9.12. Models for case study II

| Linear model | | Nonlinear model | |
|--------------|------------|-----------------|----------------------------|
| Coefficients | Regressors | Coefficients | Regressors |
| 0.0000 | 1 | 0.0000 | 1 |
| 0.9504 | $y(k - 1)$ | 1.3072 | $y(k - 1)$ |
| -0.0120 | $u(k - 1)$ | -0.0141 | $u(k - 1)$ |
| -0.0105 | $u(k - 3)$ | 0.0186 | $u(k - 5)u(k - 1)$ |
| -0.0204 | $u(k - 2)$ | 0.0359 | $y(k - 5)$ |
| -0.0694 | $y(k - 5)$ | -0.0040 | $u^2(k - 5)$ |
| -0.0039 | $u(k - 4)$ | -0.0203 | $u(k - 2)$ |
| | | 0.0200 | $u(k - 4)u(k - 2)$ |
| | | 0.0059 | $u(k - 5)$ |
| | | -0.0169 | $u(k - 5)u(k - 4)u(k - 1)$ |
| | | 0.0051 | $u(k - 3)$ |
| | | -0.0164 | $u^2(k - 4)$ |
| | | -0.0183 | $u(k - 4)u^2(k - 2)$ |
| | | 0.0090 | $u(k - 3)u(k - 2)$ |
| | | -0.0036 | $u(k - 5)u(k - 3)$ |
| | | 0.0189 | $u^3(k - 4)$ |
| | | -0.3860 | $y(k - 3)$ |
| | | 0.0060 | $u(k - 4)$ |
| | | -0.0097 | $u(k - 5)u(k - 2)$ |
| | | 0.0062 | $u^2(k - 2)u(k - 1)$ |

tion accuracy of the models identified. The asymmetric behavior of the actual process is evident. Since the linear model has a constant gain, it predicts equal magnitude changes in the output in response to equal changes in the input. However, the nonlinear input-output model captures the asymmetric behavior of this process. Both the gains and time constants for these two changes are modeled well. Figure 9.25 shows the improved modeling performance of the AR-Volterra model for a validation input sequence with $P_s = 0.05$.

Simulation results—closed-loop setpoint tracking. For the closed-loop simulations, a performance specification of $\pm 1000 \text{ kg kmol}^{-1}$ around the target value for the required product is considered acceptable. Hence, for a grade change, it is desirable to enter quickly and remain within the new product specification bounds. Nonlinear MPC, linear MPC, and a proportional integral (PI) controller are tuned to yield comparable responses for a setpoint change from $25,000 \text{ kg kmol}^{-1}$ to $38,000 \text{ kg kmol}^{-1}$. The PI controller tuning parameters were obtained using IMC tuning rules (Morari and Zafiriou, 1989) and are $K_c = -1.371$ and $\tau_I = 0.225 \text{ h}$. The corresponding IMC filter time constant is 0.40 h . The tuning parameters for linear MPC are $m = 1$, $p = 20$, $\gamma = 1$, $\lambda = 0$, $\Phi_{r2} = 0.94$, $\Phi_{r1} = 0.94$, and $L = 0$. The tuning parameters for nonlinear MPC are $m = 1$, $p = 10$, $\gamma = 1$, $\lambda = 0$, $\Phi_{r2} = 0.8$, $\Phi_{r1} = 0.7$, and $L = 0$. The linear and nonlinear MPC schemes employ different parameters because the underlying linear models used by the two controllers are

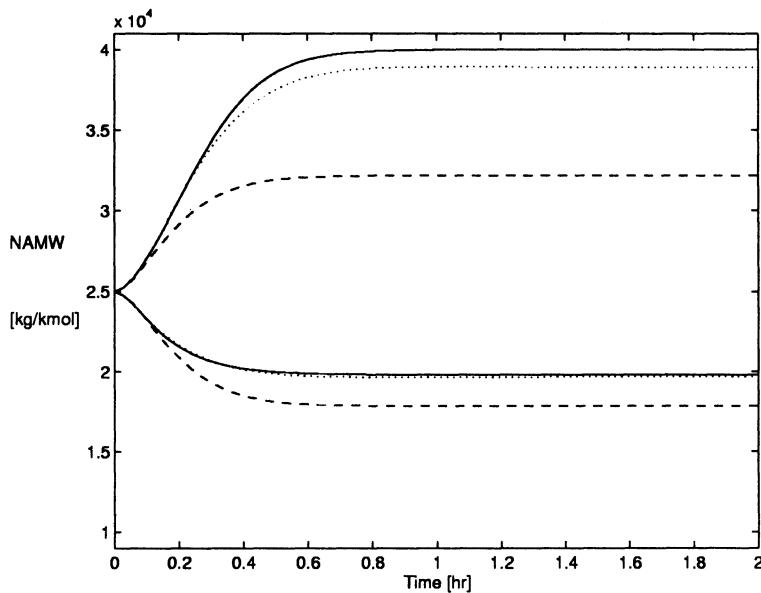


Fig. 9.24. Open-loop simulations for step changes of $\pm 0.012183 \text{ m}^3 \text{ h}^{-1}$ in F_l from its nominal value of $\pm 0.016783 \text{ m}^3 \text{ h}^{-1}$. Solid: nonlinear; dashed: linear; dotted: AR-Volterra

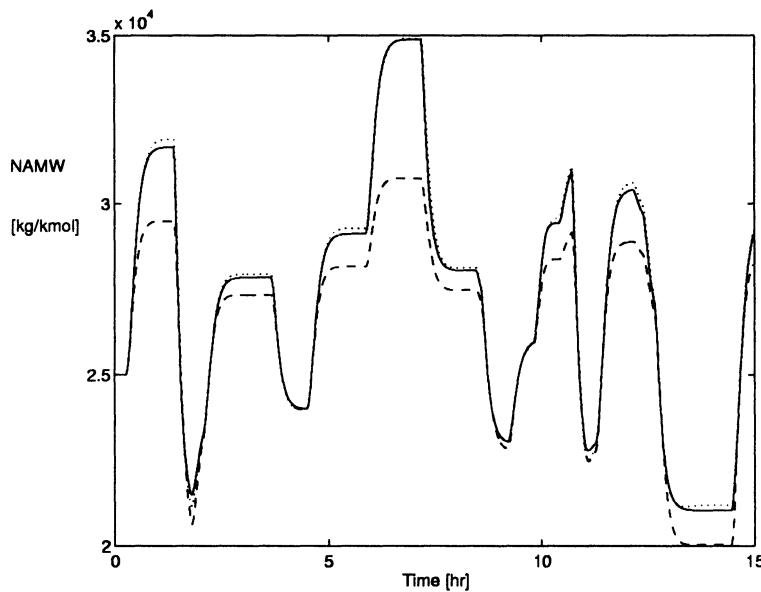


Fig. 9.25. Validation sequence for random steps in initiator flow rate. Solid: nonlinear; dashed: linear; dotted: AR-Volterra

different. The three controllers were tuned to yield comparable performance for one setpoint change to enable a fair performance comparison for setpoint tracking and disturbance rejection simulations. The model predictive controllers are tuned so that the manipulated variable did not reach its lower constraint. This is done so that improved performance can be primarily attributed to model accuracy and not influenced by actuator constraints. The closed-loop responses obtained by these three controllers are nearly indistinguishable for this setpoint change (Figure 9.26). The NAMW enters into the new product specification bounds after 19.8 min using the linear MPC and PI control schemes denoted by the dashed and dash-dot lines, respectively. The response obtained with nonlinear MPC, denoted by the dotted line, actually enters in the bounds after 21.6 min. However, all three controllers bring the reactor to steady state at nearly the same time. For practical purposes, the closed-loop performance of all three controllers is essentially the same. The closed-loop responses for a new setpoint of $20,000 \text{ kg kmol}^{-1}$ are shown in Figure 9.27. Here the performance of linear MPC and a PI controller are again nearly the same. This is not surprising given the unconstrained, single-input-single-output problem. Both linear MPC and PI control bring the process into the new operating region after 54 min. The nonlinear MPC scheme, however, brings the reactor to the new operating bounds after only 16.2 min, a reduction in grade transition time of approximately 70%. In addition, the nonlinear controller reaches the new setpoint after 1.32 h, whereas the other two controllers take 7.05 h, an improvement of approximately 80%.

Simulation results—closed-loop disturbance rejection. Another important measure of control system performance is the ability to reject unmeasured disturbances. In response to an unmeasured disturbance, it is desired to re-enter the product specification bounds very quickly. Morari and Zafiriou (1989) proved that for the case of a linear plant and an arbitrary linear model, the two degrees of freedom structure in Figure 8.1 allows the designer to tune independently for both setpoint tracking and disturbance rejection. However, in the case of a nonlinear plant, the $F_1(z)$ filter in Figure 8.1 acts to filter both unmeasured disturbances and plant-model mismatch, both of which are nonlinear. In the case of severe nonlinearity, $F_1(z)$ may need to be detuned to the degree that $F_2(z) = F_1(z)$, effectively preventing the controller from being a two degrees of freedom controller.

Controlling the nonlinear reactor with linear MPC using the two degrees of freedom controller with $F_2(z) \neq F_1(z)$ involves an inevitable performance tradeoff. Disturbance rejection can be improved by tuning $F_1(z)$ very aggressively. However, the resulting filter parameter for $F_1(z)$ results in significantly worse performance for the $38,000 \text{ kg kmol}^{-1}$ setpoint change. For this case study, the filter time constant for $F_1(z)$ cannot be reduced below that of $F_2(z)$ without incurring performance deterioration for the $38,000 \text{ kg kmol}^{-1}$ setpoint change. Hence, linear MPC has to be detuned for robustness to such a degree that it is not able to treat setpoints and disturbances differently.

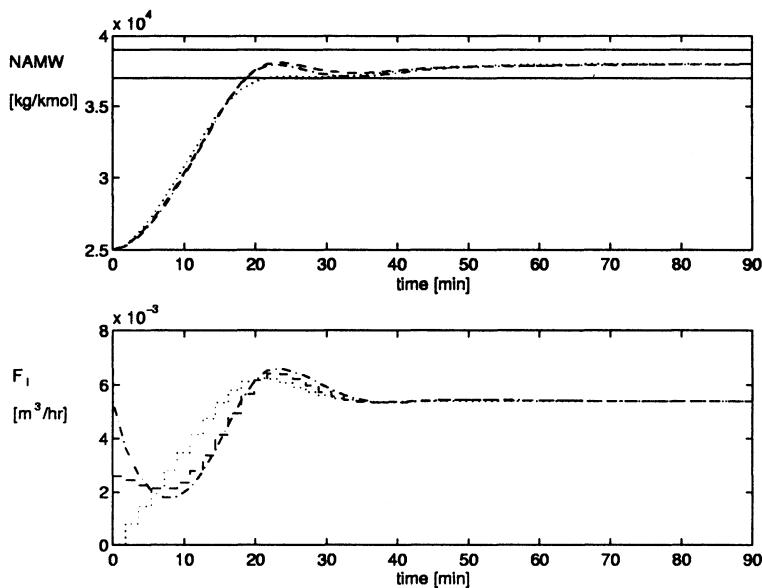


Fig. 9.26. Closed-loop simulation for a filtered step setpoint change from 25,000 kg kmol⁻¹ to 38,000 kg/kmol⁻¹ in NAMW. PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted

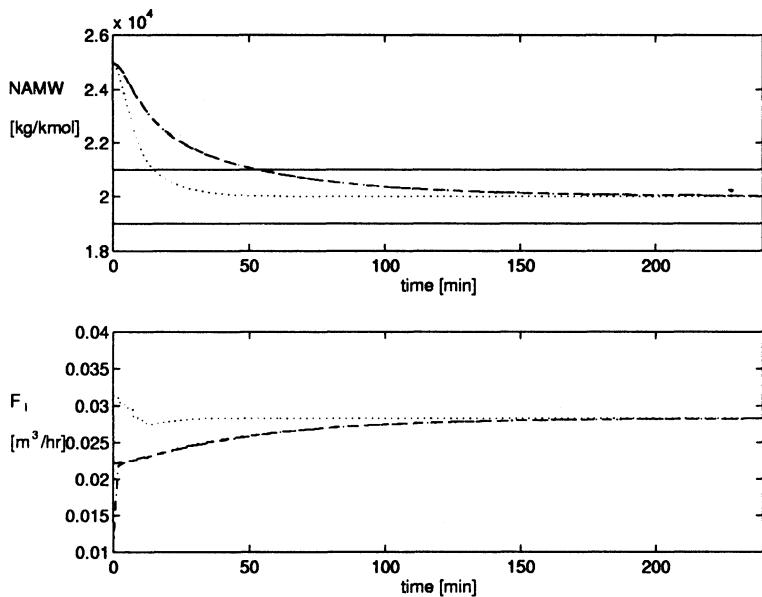


Fig. 9.27. Closed-loop simulation for a filtered setpoint step change from 25,000 kg kmol⁻¹ to 20,000 kg kmol⁻¹ in NAMW. PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted

This observation is in agreement with the observation in Prett and García (1988) that $F_1(z)$ must be detuned to guarantee stability in the face of model errors. Hence, the performance of the PI controller and linear MPC are nearly the same for disturbance rejection simulations in this example as well.

Ricker (1990) proposed a structure where $F_2(z) = F_1(z)$ in Figure 8.1 that, combined with state estimation, acts as a two degrees of freedom controller for control of a linear plant. Ricker noted that if the model is poor, large estimator gains may lead to performance deterioration and in some cases closed-loop instability. Incorporating state estimation into the linear MPC scheme leads to the same performance tradeoff as in the two-filter arrangement discussed earlier. A nonzero estimator gain yielded improved performance for disturbance rejection at the expense of poorer performance for the $38,000 \text{ kg kmol}^{-1}$ setpoint change.

As evidenced by Figures 9.24 and 9.25, the AR-Volterra model is a very accurate model of the true nonlinear process. Hence, the plant-model mismatch is quite small when the AR-Volterra model is used in a control scheme, and the value being filtered by $F_1(z)$ in Figure 8.1 is due primarily to unmeasured disturbances and not plant-model mismatch. Although the nonlinearity of the process prevents complete decoupling of the tuning problem for set-point tracking and disturbance rejection, the more accurate model enables the tuning problem to be decoupled to some degree, enabling the designer to tune for both setpoint tracking and disturbance rejection. Hence, the parameters for the two filters for nonlinear MPC are not equal ($\Phi_{r2} = 0.8$, $\Phi_{r1} = 0.7$).

Figure 9.28 shows the responses of the three controllers for a change in monomer feed concentration from $C_{m_{in}} = 6 \text{ kmol m}^{-3}$ to $C_{m_{in}} = 5 \text{ kmol m}^{-3}$. A decrease in monomer concentration reduces the degree of polymerization lowering the molecular weight (Hill, Jr., 1977). Linear MPC and a PI controller return the molecular weight to its specification range after 37.8 min, whereas nonlinear MPC achieves this goal after only 25.2 min, a 33.3% improvement. The times needed to return the molecular weight to within 1% of its setpoint are 68.4 min, 66.6 min, and 39.6 min, for a PI controller, linear MPC, and nonlinear MPC respectively. Hence, if the performance specification bounds are more stringent than $\pm 1000 \text{ kg kmol}^{-1}$, the performance improvement obtained with a nonlinear model-based controller is even greater for both the servo and regulator problems.

Stability analysis. Consider the nominal case ($G(z) = \tilde{G}(z)$) where the AR-Volterra model in Table 9.12 is used as the plant. The setpoint filter parameter is $\Phi_{r2} = 0.8$. Hence, the corresponding pole lies inside the unit circle and $F_2(z)$ is stable. The five poles of the linear portion of the plant are computed to be $z = 0.8254, 0.8254, 0.3746, 0.3746$, and 0.3755 . All poles lie inside the unit circle, guaranteeing that $G(z)$ is BIBO stable. In addition, a μ -analysis for the model resulted in $\mu < 1$ for the region over which the

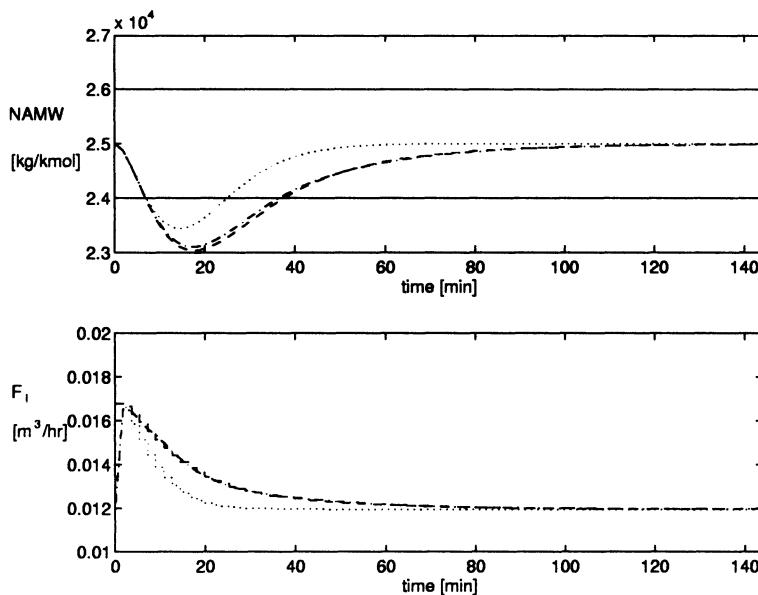


Fig. 9.28. Closed-loop simulation for an unmeasured step disturbance in monomer feed composition from its nominal value of $C_{min} = 6 \text{ kmol m}^{-3}$ to $C_{min} = 5 \text{ kmol m}^{-3}$. PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted

model is valid ($\tilde{u} \in [-0.7259, 0.7259]$). Hence, the model is also exponentially stable for its region of validity.

The zeros of the linear part of the AR-Volterra model are $z=1.6015$, 0.7554 , 0.5881 , and 0.5881 . The zero outside the unit circle results in unstable zero dynamics and renders the inverse-based controller unstable. The stability of the p -inverse controller is considered with $p > 1$. Employing the uncertainty description in Hernández (1992) leads to an uncertain system with five uncertainties. This approach uses “lumped” uncertainty and is a conservative uncertainty description. Stability of the p -inverse controller could be guaranteed for the input range $\tilde{u} \in [-0.05, 0.05]$ and for $p = 2, 3, 5$, and 10 .

A more structured uncertainty formulation was employed that contained six uncertain parameters. The less conservative uncertainty description was obtained by first defining states as:

$$\begin{aligned}
 x_1(k) &= \tilde{y}(k) \\
 x_2(k) &= \tilde{y}(k-1) \\
 x_3(k) &= \tilde{y}(k-2) \\
 x_4(k) &= \tilde{y}(k-3) \\
 x_5(k) &= \tilde{y}(k-4) \\
 x_6(k) &= \tilde{u}(k-1) \\
 x_7(k) &= \tilde{u}(k-2) \\
 x_8(k) &= \tilde{u}(k-3) \\
 x_9(k) &= \tilde{u}(k-4)
 \end{aligned}$$

and noting that the AR-Volterra model in Table 9.12 may be recast as a linear-plus-uncertainty system as follows:

$$\begin{aligned}
 \underline{x}(k+1) = [A \ b] \begin{bmatrix} \underline{x}(k) \\ \tilde{u}(k) \end{bmatrix} + \delta_1[r_{B1}x_8] + \\
 \delta_2[r_{B2}x_8 + r_{B4}x_6] + \delta_3[r_{B3}\tilde{u} + r_{B5}x_9 + r_{B6}x_6] + \\
 \delta_4[r_{B7}\tilde{u} + r_{B8}x_8] + \delta_5[r_{B9}\tilde{u}] + \delta_6[r_{B10}x_6 + r_{B11}x_9]
 \end{aligned}$$

where $\delta_i \in [-1, 1]$ and $r_{B1}, r_{B2}, \dots, r_{B11}$ denote the largest bounds on selected terms. Table 9.13 contains the variables for the r_B values and numerical values for the input range $\tilde{u} \in [-0.14, 0.14]$. Although the number of uncertain parameters increases, the uncertainty is focused in a more structured manner rather than using a “lumped” uncertainty description. The sizes of the uncertain parameters decrease significantly, and the smaller uncertainty values more than offset the effect of adding one additional uncertainty parameter. The analysis of the stability of the p -inverse controller is given in Table 9.14 for the input range $\tilde{u} \in [-0.14, 0.14]$. It is evident from Table 9.14 that the p -inverse controller is guaranteed to be stable for a significantly larger operating region than before. In addition, the less conservative uncertainty description is less sensitive to larger p -values. One might expect that the upper bound for μ should increase with increasing values of p due to the increase in the number of uncertainty parameters. However, as p is increased, the p -inverse controller is detuned, which has the effect of lowering the upper bound on μ .

The closed-loop stability analysis presented is only valid for a relatively small operating region ($\tilde{u} \in [-0.14, 0.14]$) using a p -inverse controller. In addition, it assumes that there is no plant-model mismatch. However, the closed-loop simulations in Figures 9.26, 9.27, and 9.28 show that the closed-loop system using a more aggressively tuned general nonlinear MPC scheme based on the AR-Volterra model is stable over a large operating region and is robust in the presence of unmeasured disturbances and plant-model mismatch.

Although performance improvements were shown in using the AR-Volterra model in a nonlinear MPC scheme, this case study was a single-input-single-output problem. Most problems encountered in the chemical industry are multivariable. However, the proposed model structure can be implemented

in multivariable control problems as well, as will be demonstrated in the next case study.

Table 9.13. Bounds using the less conservative uncertainty description

| r_B | Variable | Value for $\tilde{u} \in [-0.14, 0.14]$ |
|-----------|--------------------|---|
| r_{B1} | $\theta_{15}x_8^2$ | 3.70×10^{-4} |
| r_{B2} | $\theta_{11}x_8$ | 2.30×10^{-4} |
| r_{B3} | θ_3x_9 | 2.60×10^{-3} |
| r_{B4} | θ_7x_8 | 2.80×10^{-3} |
| r_{B5} | θ_5x_9 | 5.60×10^{-4} |
| r_{B6} | $\theta_{18}x_9$ | 1.36×10^{-3} |
| r_{B7} | $\theta_{19}x_6^2$ | 1.22×10^{-4} |
| r_{B8} | $\theta_{12}x_6^2$ | 3.59×10^{-4} |
| r_{B9} | $\theta_9x_9x_8$ | 3.31×10^{-4} |
| r_{B10} | $\theta_{13}x_7$ | 1.26×10^{-3} |
| r_{B11} | $\theta_{14}x_7$ | 5.04×10^{-4} |

Table 9.14. Stability analysis of the p -inverse model for $\tilde{u} \in [-0.14, 0.14]$ using less conservative uncertainty formulation

| p | $\min_D \bar{\sigma}(DM_{11}^T D^{-1})$ | Number of uncertainties |
|-----|---|-------------------------|
| 1 | [1.1929, 1.9220] | 6 |
| 2 | [0.9348, 0.9843] | 12 |
| 3 | [0.9339, 0.9861] | 18 |
| 5 | [0.9299, 0.9815] | 30 |
| 10 | [0.9063, 0.9837] | 60 |

9.3 Case study III – multivariable polymerization reactor

9.3.1 Process description

The third case study considers the control of the free-radical solution polymerization of styrene in a jacketed CSTR. Hidalgo and Brosilow (1990) proposed a nonlinear MPC scheme to control the reactor temperature at the unstable steady-state by manipulating the cooling water and monomer flow rates. For this example, two changes are made to the original problem. First, it is desired to control the NAMW in addition to the reactor temperature to

obtain a multiple-input-multiple-output (MIMO) control problem. To accomplish these objectives, the initiator and cooling water flow rates are selected as manipulated variables. A schematic of this process is shown in Figure 9.29. The monomer flow rate is held constant. The second modification to the original problem is that the reactor is controlled around the low conversion stable steady-state point. Derivation of a bilinear model at a stable steady-state, using Carleman linearization (Rugh, 1981), enabled calculation of the model parameters using Equations (4.39) through (4.42). The nonlinear model was obtained by augmenting the original four-state model with two additional equations that are used to determine the NAMW:

$$\begin{aligned}\frac{d[I]}{dt} &= \frac{(Q_i[I_f] - Q_t[I])}{V} - k_d[I] \\ \frac{d[M]}{dt} &= \frac{(Q_m[M_f] - Q_t[M])}{V} - k_p[M][P] \\ \frac{dT}{dt} &= \frac{Q_t(T_f - T)}{V} + \frac{(-\Delta H_r)}{\rho C_p} k_p[M][P] - \frac{hA}{\rho C_p V}(T - T_c) \\ \frac{dT_c}{dt} &= \frac{Q_c(T_{cf} - T_c)}{V_c} + \frac{hA}{\rho_c C_{pc} V_c}(T - T_c) \\ \frac{dD_0}{dt} &= 0.5k_t[P]^2 - \frac{Q_t D_0}{V} \quad (9.3)\end{aligned}$$

$$\frac{dD_1}{dt} = M_m k_p[M][P] - \frac{Q_t D_1}{V} \quad (9.4)$$

$$y_1 = \frac{D_1}{D_0}$$

$$y_2 = T$$

where:

$$\begin{aligned}[P] &= \left[\frac{2f k_d[I]}{k_t} \right]^{0.5} \\ k_i &= A_i \exp(-E_i/T), \quad i = d, p, t \\ Q_t &= Q_i + Q_s + Q_m\end{aligned}$$

Two assumptions are made in deriving Equations (9.3) and (9.4). First, it is assumed that the rate of disappearance of monomer is primarily due to propagation. Hence, disappearance of monomer due to chain transfer to monomer is not included in the model. This assumption is also made in Hidalgo and Brosilow (1990) and Jaisinghani and Ray (1977). The second assumption is that the overall chain termination rate constant k_t is composed of both combination k_{T_C} and disproportionation k_{T_d} contributions (Schmidt and Ray, 1981) where:

$$k_t = k_{T_c} + k_{T_d} \quad (9.5)$$

For styrene, chain termination has been experimentally determined to occur solely by combination in bulk (Bevington et al., 1954) and solution (Timm

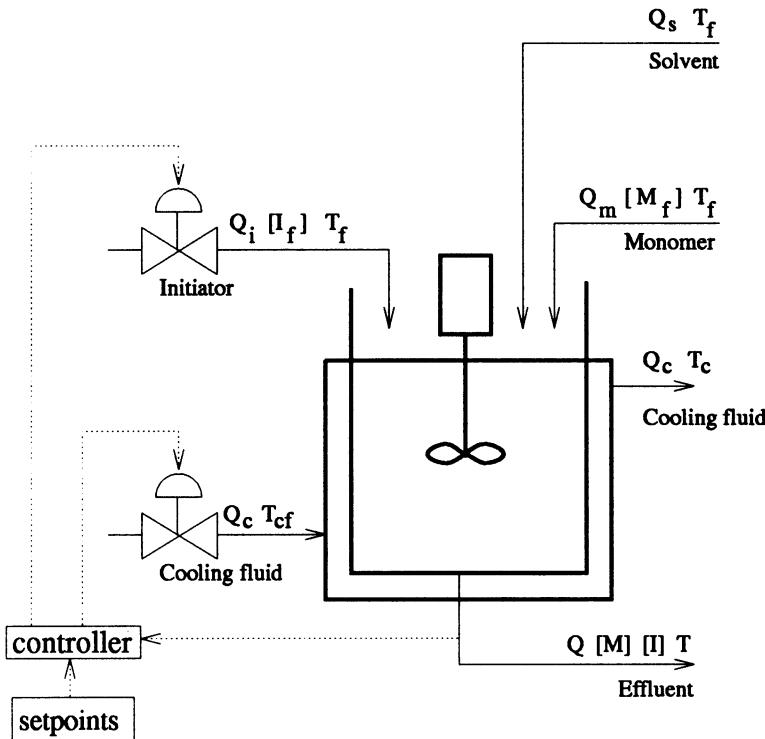


Fig. 9.29. Control configuration for case study III

and Rachow, 1974) polymerization. Hence, $k_t = k_{T_c}$ in Equation (9.5). The kinetic and thermodynamic parameters (which correspond to Process 1 in the original paper of Hidalgo and Brosilow (1990)), model parameters, and steady-state operating conditions are listed in Tables 9.15, 9.16, and 9.17 respectively. The model assumptions affecting the parameter selection are the same as those in the original paper by Hidalgo and Brosilow (1990). The first assumption is that the solvent volume fraction is maintained at 0.6 so that the gel effect may be neglected (Choi, 1986). Hence, the solvent flow rate was manipulated according to the following equation:

$$Q_s = 1.5Q_m - Q_i$$

A second model assumption is that the temperature must be less than 423 K. This is justified by the fact that the rate of thermal initiation becomes significant at temperatures greater than 373 K, and probably dominates catalytic initiation at temperatures above 423 K (Biesenberger and Sebastian, 1983).

The goal of the control system is to drive the polymerization system to a new state to produce polymers with different NAMWs while keeping the temperature at its setpoint.

9.3.2 Volterra MPC design

The vectors \mathbf{u} , \mathbf{x} , and \mathbf{y} are placed in deviation form, and a bilinear representation of the nonlinear model is obtained using Carleman linearization (Rugh, 1981) as detailed in Section 4.7. The model matrices are scaled with the following matrices:

$$\mathbf{N}_u = \begin{bmatrix} 108 & 0 \\ 0 & 471.6 \end{bmatrix}$$

$$\mathbf{N}_v = \begin{bmatrix} 2500 & 0 \\ 0 & 0.5 \end{bmatrix}$$

The scaled, continuous-time bilinear system is discretized using the explicit fourth-order Runge-Kutta method given by Equations (4.35) through (4.38) with a sampling time of 1 h. The open-loop and closed-loop responses are on the same order as those observed in the control of a 1,000 ℓ polymerization reactor in Congalidis et al. (1989). The largest negative eigenvalue of the Hidalgo and Brosilow system is -0.74 h^{-1} at the low conversion steady state. The large time constant for this system arises from the large reactor volume of 3,000 ℓ and operation of the reactor at the low temperature, low conversion steady-state. The model parameters are calculated using Equations (4.39) through (4.42). The Volterra controller described in Section 8.4.2

Table 9.15. Kinetic and thermodynamic parameters for case study III

| | | | |
|---------------|---|-----------------------|--|
| f | = | 0.6 | |
| A_d | = | 5.95×10^{13} | 1 s^{-1} |
| E_d | = | 14,897 | K |
| A_t | = | 1.25×10^9 | $\ell \text{ mol}^{-1} \text{ s}^{-1}$ |
| E_t | = | 843 | K |
| A_p | = | 1.06×10^7 | $\ell \text{ mol}^{-1} \text{ s}^{-1}$ |
| E_p | = | 3,557 | K |
| $-\Delta H_r$ | = | 16700 | cal mol^{-1} |
| hA | = | 70 | cal $\text{K}^{-1} \text{ s}^{-1}$ |
| ρC_p | = | 360 | cal $\text{K}^{-1} \ell^{-1}$ |
| ρC_{pc} | = | 966.3 | cal $\text{K}^{-1} \ell^{-1}$ |

Table 9.16. Parameters for case study III

| | | | |
|----------|---|--------|-----------------------|
| Q_s | = | 0.1275 | $\ell \text{ s}^{-1}$ |
| Q_m | = | 0.105 | $\ell \text{ s}^{-1}$ |
| V | = | 3000 | ℓ |
| V_c | = | 3312.4 | ℓ |
| $[I_f]$ | = | 0.5888 | mol ℓ^{-1} |
| $[M_f]$ | = | 8.6981 | mol ℓ^{-1} |
| T_f | = | 330 | K |
| T_{cf} | = | 295 | K |
| M_m | = | 104.14 | g mol $^{-1}$ |

Table 9.17. Steady-state operating conditions for case study III

| | | | |
|-------------|---|-------------------------|-------------------------|
| $x_1 = [I]$ | = | 6.6832×10^{-2} | $\text{mol } \ell^{-1}$ |
| $x_2 = [M]$ | = | 3.3245 | $\text{mol } \ell^{-1}$ |
| $x_3 = T$ | = | 323.56 | K |
| $x_4 = T_c$ | = | 305.17 | K |
| $x_5 = D_0$ | = | 2.7547×10^{-4} | $\text{mol } \ell^{-1}$ |
| $x_6 = D_1$ | = | 16.110 | $\text{g } \ell^{-1}$ |
| $u_1 = Q_i$ | = | 0.03 | $\ell \text{ s}^{-1}$ |
| $u_2 = Q_c$ | = | 0.131 | $\ell \text{ s}^{-1}$ |
| y_1 | = | 58,481 | g mol^{-1} |
| y_2 | = | 323.56 | K |

is used in this case study, i.e. a fourth-order nonlinear program is solved at each sampling interval. The tuning parameters for linear and Volterra MPC were $m = 1$, $p = 20$, $[\gamma_1, \gamma_2] = [2, 1]$, and $[\lambda_1, \lambda_2] = [1000, 1]$. The truncation order of the model was $N = 35$.

simulation results—open-loop. Open-loop responses of the nonlinear, linear, and second-order Volterra models to step changes of $\pm 27\ell \text{ h}^{-1}$ in Q_i from its nominal value of $108\ell \text{ h}^{-1}$ are shown in Figure 9.30. In both simulations, the output of the second-order Volterra model more closely tracks the actual nonlinear plant output for the NAMW (the curves are indistinguishable for one of the step changes). Although these step changes correspond to $\pm 25\%$ changes in the initiator flow rate, the temperature does not deviate significantly from the nominal value. Both the linear and second-order Volterra models accurately predict the moderate temperature changes of the plant.

Simulation results—closed-loop. The result of a closed-loop simulation for a setpoint change from $58,481$ to $80,000 \text{ g mol}^{-1}$ is shown in Figure 9.31. As in the preceding case study, the Volterra model predictive controller outperforms the linear model predictive controller, as the number average molecular weight more closely tracks the reference trajectory. Conversely, linear MPC results in significant overshoot. The overshoot is caused by the more aggressive manipulated variable profiles in Figure 9.31. Linear MPC could be detuned to yield an overdamped response for the NAMW. However, improved performance for positive setpoint changes would be achieved at the expense of performance deterioration for negative setpoint changes.

The dash-dot lines in Figure 9.31 correspond to input and output profiles obtained using nonlinear QDMC with $M = 1$, $P = 12$, $[\gamma_1, \gamma_2] = [1, 2000]$, and $[\lambda_1, \lambda_2] = [0, 0]$. Different tuning parameters were used for nonlinear QDMC compared with those used for linear and Volterra MPC, because the underlying linear models were different from those used in the other two controllers. The tuning parameters used in nonlinear QDMC resulted in an improvement in closed-loop performance compared with incorporating the tuning parameters used in linear and Volterra MPC.

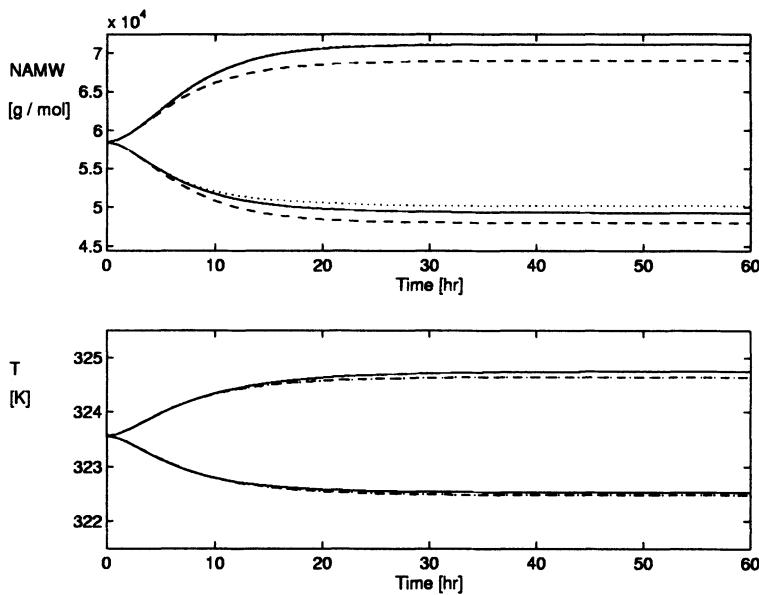


Fig. 9.30. Open-loop simulations for step changes of $\pm 27 \text{ l h}^{-1}$ in Q_i from its nominal value of 108 l h^{-1} . Solid: nonlinear, dashed: linear, dotted: Volterra

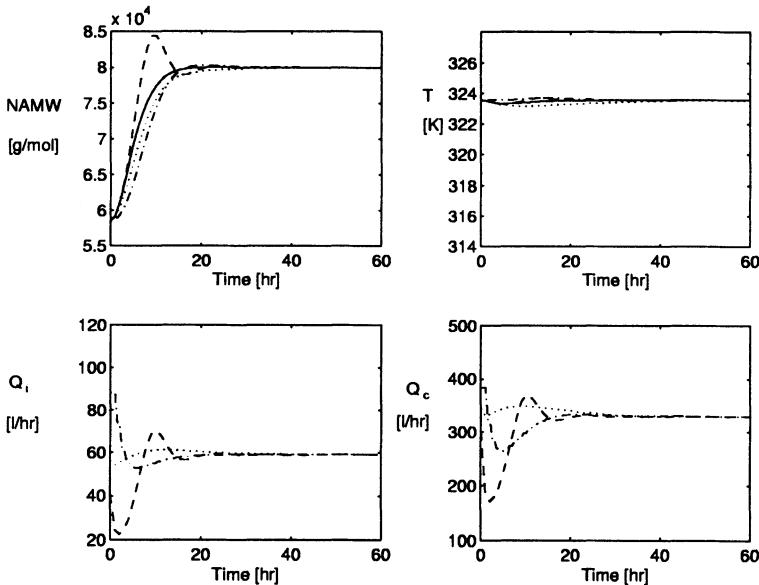


Fig. 9.31. Closed-loop simulation for a step setpoint change from $58,481$ to $80,000$ g/mol. Solid: reference, dashed: linear MPC, dotted: Volterra MPC, dash dot: non-linear QDMC

The performance obtained using nonlinear QDMC is comparable to that of using Volterra MPC. The response of the NAMW is approximately first-order, and the temperature is maintained close to its setpoint. In addition, the input profiles for both flow rates are very smooth. An attractive feature of nonlinear QDMC is that a single quadratic program needs to be solved on-line, whereas the second-order Volterra controller requires the solution of a fourth-order nonlinear program. However, the nonlinear QDMC algorithm requires the availability of a fundamental model.

The simulation results in Figure 9.31 employ a controller based on a triangular second-order Volterra model that did not include cross terms ($u_1(k-1)u_2(k-1)$, ... etc.) in the inputs. For a truncation order of $N = 35$, this resulted in 2,520 second-order coefficients. The first- and second-order parameters for this case study are shown in Figures 9.32 and 9.33 respectively. The first-order parameters are approximately one order of magnitude larger than the second-order terms, and the scaled manipulated variables were on the order of 10^{-1} . Hence, the later $b_{l,j,l,n}^i$ terms in Figure 9.33 make a very small contribution to the model. Although the reactor temperature is affected by the heat of polymerization and the Arrhenius temperature dependence of the kinetic parameters, the large reactor acts as a heat sink and prevents a large change in temperature provided there is always a nominal amount of cooling water flowing through the cooling jacket. Hence, linear models can be used to describe the relationship between the two inputs and the reactor temperature. In addition, the initiator flow rate has a much greater effect on the NAMW than the cooling water flow rate, and a linear model can also be used to relate the cooling water flow rate and the NAMW. In addition, if the truncation order of the second-order parameters in the remaining Volterra model is reduced from $N = 35$ to $N = 20$, the number of second-order coefficients is reduced to 210. Repeating the same setpoint change simulation yielded the closed-loop performance illustrated in Figure 9.34. The closed-loop performance does not degrade significantly with the elimination of over 90% of the second-order parameters in the second-order Volterra model. This is another application of the model pruning approach described in Section 3.6.1.

9.3.3 Remarks

One of the potential drawbacks of this approach is that second-order Volterra models require many parameters. In case studies II and III, it was shown that the number of second-order coefficients could be significantly reduced with no significant difference in closed-loop performance. Though it seems reasonable that the memory of the linear and second-order contributions to the system dynamics should be approximately the same, the $b_{l,j,l,n}^i$ parameters are one order of magnitude smaller than the $a_{l,j}^i$ coefficients, and act to correct the linear model for the effect of nonlinearities. Hence, the later $b_{l,j,l,n}^i$ terms are very small and may be omitted without significant performance degradation in modeling and control. Another disadvantage is associated with using

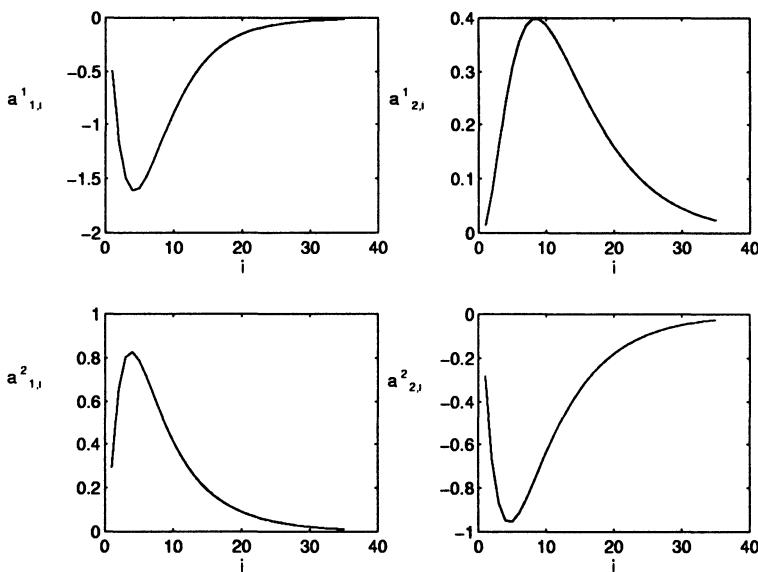


Fig. 9.32. First-order coefficients for case study III

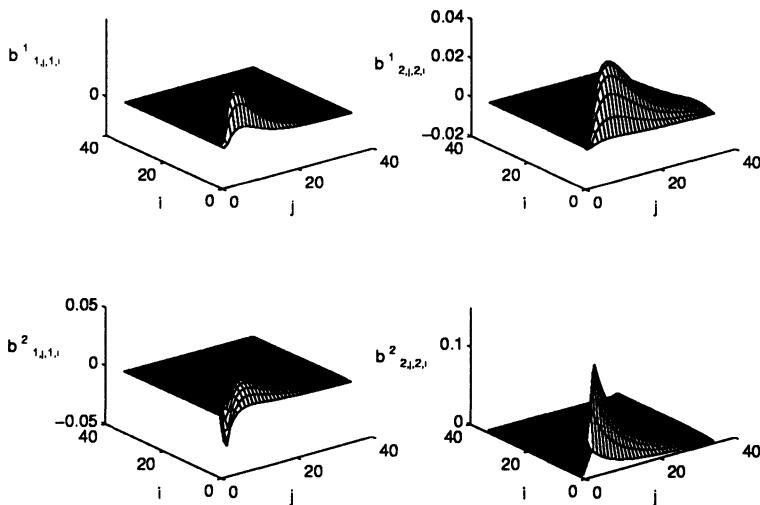


Fig. 9.33. Second-order coefficients for case study III

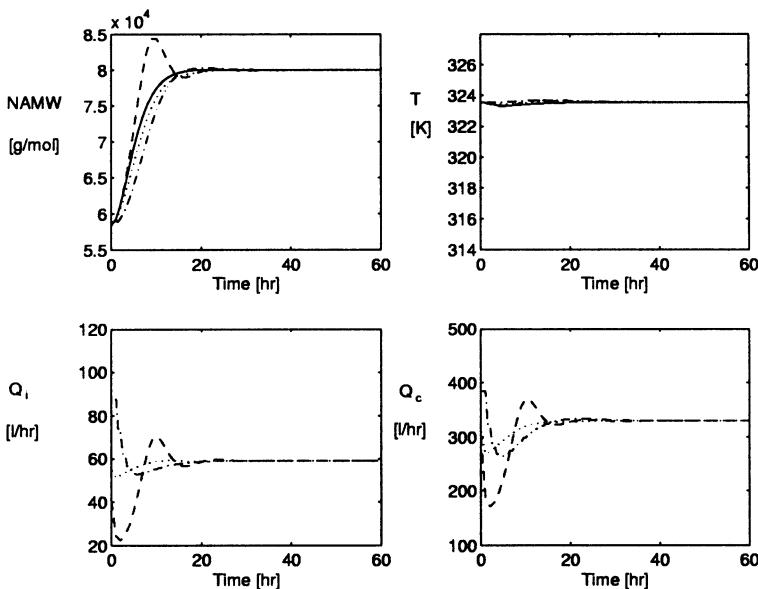


Fig. 9.34. Closed-loop simulation for a step setpoint change from 58,481 to 80,000 g mol⁻¹ using the reduced model. Solid: reference; dashed: linear MPC; dotted: Volterra MPC; dash-dot: nonlinear QDMC

a second-order nonlinear model. The performance of a linear model-based controller usually degrades as the system is moved away from the point at which the model was obtained. Although performance degradation occurs, the linear model-based controller is still stable over a fairly broad regime. The performance of a second-order nonlinear model-based controller also typically worsens as the plant is moved away from the point at which the model was obtained. However, owing to the inherent curvature of second-order models, they will always predict that the plant gain will change sign at some point. Hence, care should be taken to ensure that a second-order model-based controller is not used outside the region for which the model is valid. The ability of a second-order Volterra model derived from Carleman linearization to capture “local” and not global behavior was also demonstrated in Doyle III et al. (1995).

Despite the drawbacks associated with the second-order Volterra controller, there are also several advantages of this control scheme. One advantage is that improved performance over a linear model-based controller can be achieved, as illustrated the preceding case studies. A second advantage is that these models yield a fourth-order nonlinear program for the standard 2-norm MPC objective function if the controller is implemented as described in Section 8.4.2. Although this optimization problem is more difficult to solve than the quadratic program encountered using traditional linear MPC, it is less computationally burdensome than the nonlinear program that arises in

modified NLIMC or nonlinear MPC using polynomial ARMA models. A third advantage results from the fact that Volterra models are based only on past inputs. Hence, stability results are easier to obtain than for models based on past inputs *and* outputs.

Stability results for the continuous-time inverse Volterra series (Zheng and Zafiriou, 1993) and the 1-norm SISO MPC formulation (Genceli and Nikolaou, 1995) have been presented by other researchers. In addition, the ability to obtain the second-order Volterra model parameters from Carleman linearization and input-output data is another advantage.

9.4 Case study IV – industrial polymerization reactor

In this section, a more comprehensive benchmark problem is considered. The dynamic order is significantly higher than the first three examples, and the control structure is multivariable.

9.4.1 Process description

Figure 9.35 is a flowsheet of the copolymerization reactor with recycle loop considered by Congalidis et al. (1989). There are feed streams for monomers *A* and *B*, initiator, solvent, and chain transfer agent. In addition, an inhibitor may enter with the fresh feeds as an unmeasured disturbance. These feed streams are combined (stream 1) with the recycle stream (stream 2) and flow to the reactor (stream 3), which is assumed to be a jacketed, well-mixed tank. The heat of polymerization is removed by a coolant that flows through the jacket. Polymer, solvent, unreacted monomers, initiator, and chain transfer agent flow out of the reactor to the separator (stream 4). Polymer, residual initiator, and chain transfer agent are removed in this step. Unreacted monomers and solvent (stream 7) continue on to a purge point (stream 8) that represents venting and other losses. Purging is needed to prevent accumulation of inerts in the system. After the purge, the monomers and solvent (stream 9) are stored in the hold tank, which acts as a surge capacity to smooth out variations in the recycle flow and composition. The recycle stream (stream 2) is then added to the fresh feeds. The outputs to be controlled are production rate, composition, weight average molecular weight, and reactor temperature. The complete model description can be found in Maner and Doyle III (1997).

Monomer *A* is MMA, monomer *B* is vinyl acetate, the solvent is benzene, the initiator is AIBN, and the chain transfer agent is acetaldehyde. The monomer stream may also contain inhibitors such as *m*-dinitrobenzene (*m*-DNB). The steady-state operating conditions are listed in Table 9.18. Under these conditions, the reactor residence time is $\theta_r = 6$ h, and the overall reactor monomer conversion is 20%. These operating conditions ensure that the viscosity of the reaction medium remains moderate.

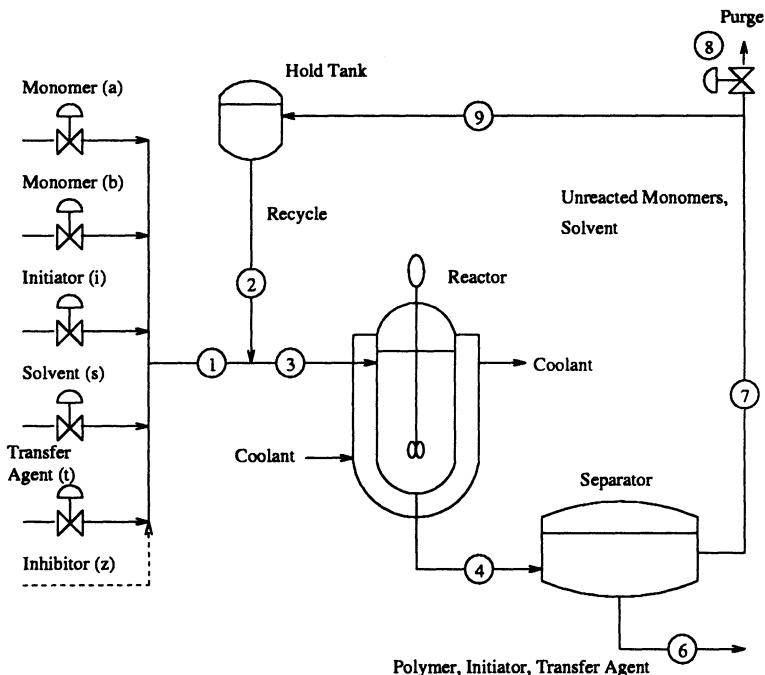


Fig. 9.35. Flow diagram of copolymerization reactor with recycle

Table 9.18. Steady-state operating conditions for case study IV

| <i>Inputs</i> | |
|---|--|
| Monomer A (MMA) feed rate | $G_{af} = 18 \text{ kg h}^{-1}$ |
| Monomer B (VAc) feed rate | $G_{bf} = 90 \text{ kg h}^{-1}$ |
| Initiator (AIBN) feed rate | $G_{if} = 0.18 \text{ kg h}^{-1}$ |
| Solvent (benzene) feed rate | $G_{sf} = 36 \text{ kg h}^{-1}$ |
| Chain transfer (acetaldehyde) feed rate | $G_{tf} = 2.7 \text{ kg h}^{-1}$ |
| Inhibitor (<i>m</i> -DNB) feed rate | $G_{zf} = 0 \text{ kg h}^{-1}$ |
| Reactor jacket temperature | $T_j = 336.0 \text{ K}$ |
| Reactor feed temperature | $T_{rf} = 353.0 \text{ K}$ |
| Purge ratio | $\xi = 0.05$ |
| <i>Reactor parameters</i> | |
| Reactor volume | $V_r = 1 \text{ m}^3$ |
| Reactor heat transfer area | $S_r = 4.6 \text{ m}^2$ |
| <i>Outputs</i> | |
| Polymer production rate | $G_{pi} = 23.327 \text{ kg h}^{-1}$ |
| Mole fraction of A in polymer | $Y_{ap} = 0.5591$ |
| Weight average molecular weight | $M_{pw} = 34,994.7 \text{ kg kmol}^{-1}$ |
| Reactor temperature | $T_r = 353.0161 \text{ K}$ |

9.4.2 AR-Volterra MPC controller design

Congalidis et al. (1989) implement feedforward control to compensate for the disturbances introduced by the recycle stream. This is accomplished by manipulating the fresh feeds in order to maintain constant feed composition and flow to the reactor. Feedforward control of the recycle stream enables the designer to separate the control of the reactor from the rest of the process.

The feedforward equations are obtained by writing component balances around the recycle addition point. For example, the mole balance for monomer *A* is:

$$F_{a3} = F_{a1} + y_{a2}F_2 \quad (9.6)$$

where the alphabetical subscript corresponds to the species, and the numerical subscript denotes the stream in Figure 9.35. Since it is desired to keep the flow of monomer *A* to the reactor (F_{a3}) constant, Equation (9.6) is solved for the fresh feed of monomer *A*:

$$F_{a1} = F_{a3} - y_{a2}F_2 \quad (9.7)$$

The corresponding feedforward control equations for fresh feeds of monomer *B* and solvent are:

$$F_{b1} = F_{b3} - y_{b2}F_2 \quad (9.8)$$

$$F_{s1} = F_{s3} - y_{s2}F_2 \quad (9.9)$$

The implementation of Equations (9.7), (9.8), and (9.9) in the overall control strategy can be seen in Figure 9.36. If any feedforward control equation causes a fresh feed to go negative, the value of that fresh feed is set to zero.

Congalidis et al. (1989) selected a control structure by ranking candidate structures according to the condition number (Morari, 1983), minimum singular value (Yu and Luyben, 1986), and the relative gain array (Bristol, 1966). Transfer functions were identified from step tests. The transfer function matrix for the selected control structure is given by Equation (9.10) and Table 9.19.

$$\begin{bmatrix} G_{pi}^+ \\ Y_{ap}^+ \\ M_{pw}^+ \\ T_r^+ \end{bmatrix} = [A \ B \ C \ D] \begin{bmatrix} G_{bf}^+ \\ (G_{af}/G_{bf})^+ \\ (G_{tf}/G_{bf})^+ \\ T_j^+ \end{bmatrix} \quad (9.10)$$

The time constants in the transfer functions in Table 9.19 are scaled by the residence time of the reactor. For a multivariable system, the condition number of the model is a measure of the difficulty of the control problem. The condition number of the transfer function matrix of the selected control structure is 91.50. A condition number on the order of 100 or larger indicates an ill-conditioned, difficult control problem, whereas a smaller condition number of approximately ten or smaller is representative of a well-conditioned, easier control problem.

Table 9.19. 4×4 process transfer function matrix

| A | B | C | D |
|--|-----------------------------|-----------------------------|---|
| $\frac{0.98711(0.12011s+1)}{0.065948s^2+0.36662s+1}$ | $\frac{0.20527}{0.4195s+1}$ | 0 | $\frac{6.4595(0.89968s+1)}{0.068739s^2+0.29708s+1}$ |
| 0 | $\frac{0.66018}{1.5098s+1}$ | 0 | $\frac{-3.7235}{0.79590s+1}$ |
| $\frac{-0.10099}{0.908165s+1}$ | $\frac{0.49084}{1.5443s+1}$ | $\frac{-0.19626}{2.711s+1}$ | $\frac{-4.7145}{0.075213s^2+0.40798s+1}$ |
| 0 | 0 | 0 | $\frac{1.0252(0.22710s+1)}{0.072732s^2+0.30978s+1}$ |

The feedback control strategy implemented by Congalidis et al. (1989) consists of four SISO PI controllers. The controller tuning parameters are listed in Table 9.20. Controller reset time τ_I is reported as a fraction of the reactor residence time. The temperature loop is determined to be dominant using the interaction index proposed by Economou and Morari (1986). This loop interacts with the other loops and should be conservatively tuned. The interaction measure also showed the remaining three loops may be tuned tightly. The combined feedforward and feedback control strategy of Congalidis et al. (1989) is shown in the block diagram in Figure 9.36.

Table 9.20. Tuning constants for multiloop PI control structure

| Manipulated | Output | K_c | $1/\tau_I$ |
|---------------------|------------|-------|------------|
| G_{bf}^+ | G_{pi}^+ | 2 | 4 |
| $(G_{af}/G_{bf})^+$ | Y_{ap}^+ | 2 | 1 |
| $(G_{tf}/G_{bf})^+$ | M_{pw}^+ | -6 | 1 |
| T_j^+ | T_r^+ | 0.2 | 6 |

The transfer function matrix given by Equation (9.10) and Table 9.19 were discretized with a sampling time T_s of 0.25 h using a zero-order hold approximation, and the resulting models were implemented in a 4×4 MPC scheme. However, tuning parameters for the multivariable controller cannot be obtained that result in significantly better performance than that of the multiloop PI control strategy for control of this process. The closed-loop responses using MPC are characterized by significant interactions between the outputs. The reason for this difficulty was attributed to the large condition number of the transfer function matrix.

In their work for the control of a high-purity distillation column, Chien and Ogunnaike (1992) note that strongly ill-conditioned systems prevent the

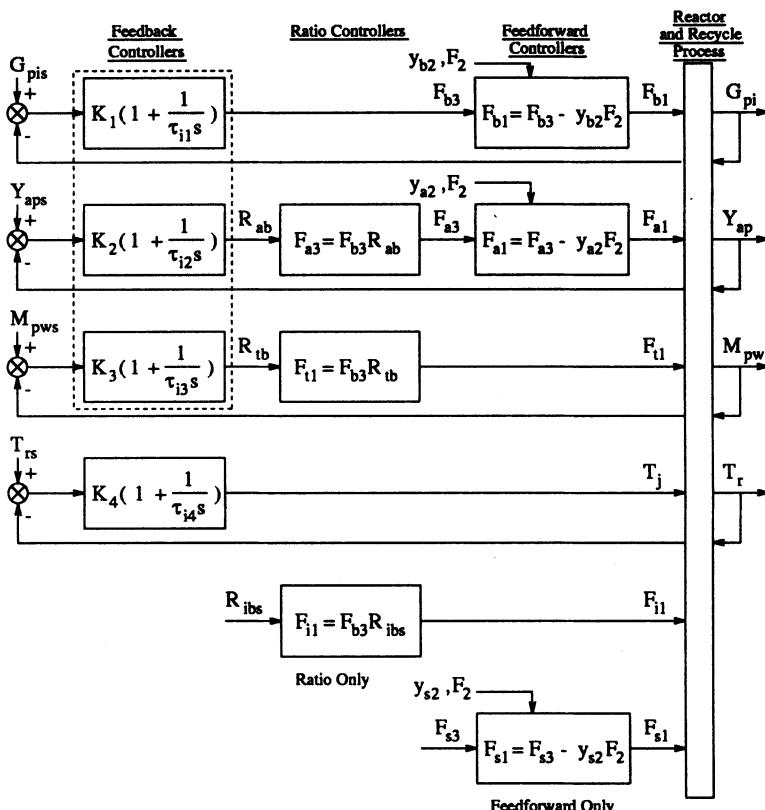


Fig. 9.36. Control structure for case study IV

full exploitation of the interaction compensation abilities of inverse-based controllers such as MPC. They observe that the performance of MPC is not better than that of a set of well-tuned, multiloop PID controllers for such systems. Chien and Ogunnaike (1992) conclude that “the customary situation is that with such systems, performance *must* be sacrificed for robustness; and sometimes this must be done to such a degree that PID controller performance becomes comparable.” Given the difficulties encountered in tuning the multivariable controller and the observations of Chien and Ogunnaike (1992), the temperature loop is closed with a PI controller with the same tuning parameters as used by Congalidis et al. (1989) to obtain a system that is better conditioned.

There are also several practical reasons for closing the temperature loop with a PI controller and designing a multivariable controller for the remaining three outputs. The temperature is typically sampled more frequently than the other three outputs. Consequently, the temperature is sampled continuously, and the remaining outputs are sampled every 0.25 h in this case study. A second reason for closing the temperature loop is safety. An industrial

company's safety regulations may require that reactor temperature be omitted from an advanced control strategy and controlled with a PI controller to prevent reactor runaway. The result is that performance is sacrificed for control of the temperature in exchange for tighter control of the other three outputs that are of primary interest to the customer (polymer composition and molecular weight) and plant manager (production rate).

With the temperature loop closed, transfer functions were identified for the 3×3 problem using step tests. The transfer function matrix is given by Equation (9.11) and Table 9.21.

$$\begin{bmatrix} G_{pi}^+ \\ Y_{ap}^+ \\ M_{pw}^+ \end{bmatrix} = [E \ F \ G] \begin{bmatrix} G_{bf}^+ \\ (G_{af}/G_{bf})^+ \\ (G_{tf}/G_{bf})^+ \end{bmatrix} \quad (9.11)$$

The time constants in the transfer functions in Table 9.21 are in units of hours. The transfer function matrix contains off-diagonal entries indicative of the interactions between inputs and outputs in this multivariable problem. The interactions raise the possibility of improved control using a multivariable controller that takes the interactions into account versus using a multiloop PI control strategy. Comparison with Equation (9.10) and Table 9.19 shows that the time constants are smaller with the temperature loop closed. The sampling time of $T_s = 0.25$ h is still in the four to ten samples per time constant range recommended by Åström and Wittenmark (1990). The condition number of the 3×3 transfer function matrix is 6.83. Hence, the 3×3 transfer function matrix is a well-conditioned transfer function matrix, and the interaction compensation ability of MPC can be exploited. The dotted box around the three PI controllers in Figure 9.36 indicates that these PI controllers are replaced by a 3×3 MPC scheme.

Table 9.21. 3×3 Process transfer function matrix

| E | F | G |
|-----------------------------------|----------------------------|------------------------------|
| $\frac{0.72137}{s^2 + 1.14s + 1}$ | $\frac{0.22246}{1.9s + 1}$ | 0 |
| $\frac{0.15963}{7.5s + 1}$ | $\frac{0.61361}{8.3s + 1}$ | 0 |
| 0 | $\frac{0.47417}{8.6s + 1}$ | $\frac{-0.18288}{12.8s + 1}$ |

Nonlinear model identification. AR-Volterra models are identified for each of the three outputs. The entries in Table 9.21 that are zero correspond to a negligible relationship between the corresponding output and input. In order to reduce the probability of the stepwise model-building algorithm from adding an erroneous term to the model, only inputs corresponding to

nonzero columns in Table 9.21 are manipulated to identify the corresponding output. Hence in the identification of outputs y_1 and y_2 , only u_1 and u_2 are manipulated. In the identification of y_3 , only u_2 and u_3 are allowed to change values. The input profiles consisted of random, four level sequences using $T_{cl} = 4$. Hence the sample time is $T_s = 0.25$ h, but an input is not allowed to change value more than once per hour. The number of input-output data points is 500. Figure 9.37 depicts the raw input-output data used to identify models for y_1 and y_2 . Figure 9.38 shows the raw input-output data used to identify a model for y_3 . The inputs and outputs are placed in scaled, deviation variables. The scale factors are chosen so that the maximum magnitude of a scaled input or output is 1.0, i.e. $\tilde{u}_i = \frac{u_i - u_{i0}}{\hat{u}_{i0}}$ and $\tilde{y}_i = \frac{y_i - y_{i0}}{\hat{y}_{i0}}$. The scale factors \hat{u}_{i0} and \hat{y}_{i0} correspond to the maximum deviations of the input and output values generated during identification. The linear models are also scaled using the same scale factors. The stepwise model-building algorithm described in Section 9.2.5 is used with $n = 3$, and a maximum value of 4 for n_y and n_u . The linear models are listed in Table 9.22 and the nonlinear models are given in Table 9.23.

Table 9.22. Linear models

| $y_1(k)$ | | $y_2(k)$ | | $y_3(k)$ | |
|-----------------|--------------|-----------------|--------------|-----------------|--------------|
| Coefficients | Regressors | Coefficients | Regressors | Coefficients | Regressors |
| $2.574e + 00$ | $y_1(k - 1)$ | $1.9375e + 00$ | $y_2(k - 1)$ | $1.9520e + 00$ | $y_3(k - 1)$ |
| $-2.2406e + 00$ | $y_1(k - 2)$ | $-9.3852e - 01$ | $y_2(k - 2)$ | $-9.5256e - 01$ | $y_3(k - 2)$ |
| $6.5930e - 01$ | $y_1(k - 3)$ | $5.0487e - 03$ | $u_1(k - 1)$ | $6.8067e - 2$ | $u_2(k - 1)$ |
| $1.6438e - 02$ | $u_1(k - 1)$ | $-4.8989e - 03$ | $u_1(k - 2)$ | $-6.8750e - 2$ | $u_2(k - 2)$ |
| $5.3480e - 04$ | $u_1(k - 2)$ | $4.9174e - 02$ | $u_2(k - 1)$ | $-1.2659e - 01$ | $u_3(k - 1)$ |
| $-1.3103e - 02$ | $u_1(k - 3)$ | $-4.7562e - 02$ | $u_2(k - 2)$ | $1.2296e - 01$ | $u_3(k - 2)$ |
| $6.1756e - 02$ | | | | | |
| $-1.0486e - 01$ | | | | | |
| $4.6441e - 02$ | | | | | |

Simulation results—setpoint tracking. As a first comparison of the performance of three control strategies (PI, linear MPC, Volterra MPC), a +10% setpoint change in polymer composition is considered. The tuning parameters for linear and nonlinear MPC are $m = 1$, $p = 6$, $\gamma = [1 \ 1 \ 1]$, $\lambda = [0 \ 0 \ 0]$, $\Phi_{r2} = \Phi_{r1} = [0.85 \ 0.95 \ 0.97]$, and $L = [0 \ 0 \ 0]$. The nonlinear MPC scheme consists of a nonlinear model for y_3 and the linear models used in linear MPC for y_1 and y_2 . This simulation corresponds to a grade change in polymer composition with specification bounds for the composition and molecular weight. Figure 9.39 shows a comparison of the closed-loop performance of the three control strategies. The solid lines pertain to a specification of $\pm 2.85\%$ of the setpoints for copolymer composition and molecular weight. The responses obtained with the PI strategy are given by the dash-dot lines, linear MPC is

Table 9.23. Nonlinear models

| $y_1(k)$ | | $y_2(k)$ | | $y_3(k)$ | |
|--------------|----------------------------|--------------|--------------------|--------------|----------------------------|
| Coefficients | Regressors | Coefficients | Regressors | Coefficients | Regressors |
| 6.73188e-03 | 1.0 | 3.20542e-03 | 1.0 | -2.11155e-03 | 1.0 |
| 1.35354e+00 | $y_1(k-1)$ | 9.91611e-01 | $y_2(k-1)$ | 1.25752e+00 | $y_3(k-1)$ |
| 6.95238e-02 | $u_1(k-1)$ | 1.14739e-02 | $u_2(k-3)$ | -1.08828e-03 | $u_3(k-4)$ |
| 6.24946e-02 | $u_2(k-1)$ | 2.76432e-03 | $u_1(k-3)$ | -6.02635e-03 | $u_3(k-2)$ |
| -2.81177e-02 | $y_2(k-1)$ | 4.61508e-03 | $u_2(k-3)u_1(k-2)$ | 1.05496e-02 | $u_2(k-2)$ |
| -3.81750e-01 | $y_1(k-3)$ | 2.31885e-03 | $u_2(k-1)$ | -2.62788e-01 | $y_3(k-4)$ |
| 1.83873e-02 | $u_2^2(k-1)u_1(k-1)$ | 2.49130e-03 | $y_1(k-3)$ | -8.30094e-02 | $u_2(k-1)$ |
| -1.94439e-02 | $u_1(k-2)$ | 1.48864e-03 | $u_1^3(k-1)$ | -3.53710e-03 | $u_3(k-3)$ |
| -2.73727e-02 | $u_1^2(k-2)u_2(k-3)$ | 3.56563e-03 | $u_2^3(k-2)$ | 6.44994e-03 | $u_2(k-1)$ |
| 9.83108e-03 | $u_1^2(k-1)u_1(k-1)$ | | | -3.52817e-03 | $u_3(k-1)$ |
| -1.00719e-02 | $u_2(k-2)$ | | | 5.97053e-03 | $u_2(k-3)$ |
| -3.83721e-02 | $u_1(k-3)$ | | | -1.31409e-03 | $u_2^2(k-4)u_3(k-2)$ |
| -2.45688e-02 | $u_2(k-3)$ | | | 3.15526e-03 | $u_2(k-4)u_2^2(k-1)$ |
| 1.947684e-02 | $u_1(k-2)u_1(k-1)u_2(k-3)$ | | | 1.44743e-03 | $u_3(k-4)u_3(k-2)u_2(k-4)$ |
| -2.00622e-02 | $u_2^2(k-2)u_1(k-2)$ | | | -2.03269e-03 | $u_3^2(k-4)u_2(k-1)$ |
| -9.57738e-03 | $u_2^3(k-3)$ | | | -2.46657e-03 | $u_2(k-4)u_2(k-1)u_3(k-1)$ |
| 1.05313e-02 | $u_1(k-2)u_2(k-1)$ | | | 7.52168e-02 | $y_2(k-4)$ |
| | | | | 1.42674e-03 | $u_3(k-4)u_3(k-1)u_2(k-1)$ |
| | | | | 8.49375e-04 | $u_2^2(k-4)u_3(k-4)$ |

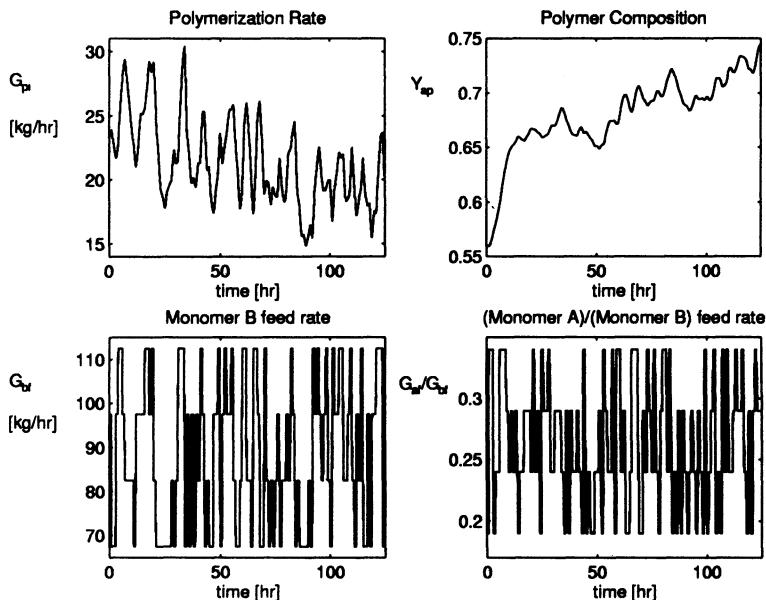


Fig. 9.37. Input-output data used to identify y_1 and y_2 (u_3 is held constant)

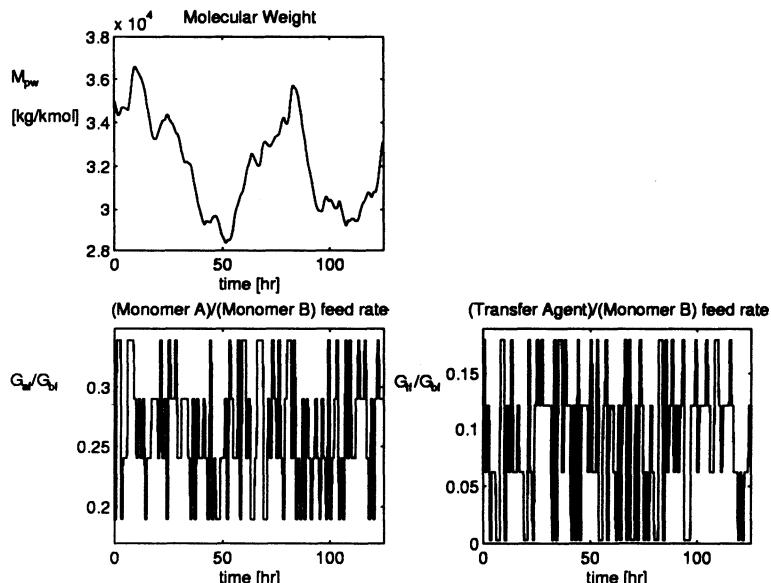


Fig. 9.38. Input-output data used to identify y_3 (u_1 is held constant)

denoted by the dashed lines, and nonlinear MPC is depicted by the dotted lines. Both MPC strategies bring the composition into its specification limits in 6 h while the PI control scheme requires 7.5 h. In addition, the MPC schemes keep the molecular weight within its specification bounds during the grade change, whereas the PI strategy does not return the molecular weight to this region for 16 h. The performance of the two MPC schemes is essentially the same for this moderate setpoint change. Hence, the performance improvement over the PI strategy is due to using multivariable compensation.

Figure 9.40 depicts the closed-loop performance comparison of the three control strategies for a +20% setpoint change in polymer composition. The PI, linear MPC, and nonlinear MPC schemes produce responses given by the dash-dot, dashed, and dotted lines respectively. The PI control strategy requires 13 h to reach the bounds for the copolymer composition. Linear MPC brings the polymer composition to its specification bounds after 10.5 h, whereas nonlinear MPC needs only 10.25 h. The difference in performance is most clearly shown in the plot for molecular weight in Figure 9.40. The PI scheme permits the molecular weight to reach $39,600 \text{ kg kmol}^{-1}$ which is far outside the product quality bounds. The molecular weight re-enters the specification region at 22.5 h but, it proceeds to violate the lower bound at 32.5 h. This output remains in the bounds after 38.75 h. The performance of linear MPC is significantly better than that of the PI scheme. The molecular weight violates the upper limit for several hours before bringing this output within specification after 13 h. The nonlinear MPC scheme results in additional performance improvement for control of the molecular weight. The AR-Volterra model accurately describes the molecular weight and enables the MPC scheme to take more appropriate control action that results in keeping this output within the product specification bounds.

It is interesting to note that in Figures 9.39 and 9.40 the multiloop PI control scheme results in tighter control of the temperature compared with the strategies using 3×3 MPC plus one PI controller. This observation is reasonable, since a tradeoff was made in designing the MPC schemes. This observation also highlights the fact that the control problem was not simplified. Performance was sacrificed in control of the temperature in exchange for improved control of the remaining outputs. For ill-conditioned systems, a multivariable controller could be detuned such that its performance is no better than that of a set of well-tuned PI controllers, or a tradeoff could be made in control of one output to achieve improved control of the remaining outputs. For the control of the copolymerization reactor, it is advantageous to make a tradeoff between control of polymer properties and temperature because the customer is concerned with the properties of the final product rather than the reactor operating temperature.

Simulation results—disturbance rejection. In addition to smooth set-point tracking, another measure of control system performance is the manner in which unmeasured disturbances are rejected. The unmeasured disturbance

considered by Congalidis et al. (1989) was the presence of an inhibitor in the fresh feed. This disturbance inhibits the polymerization reaction, which lowers both the molecular weight and polymerization rate of the copolymer. Since the polymerization reaction is exothermic, less polymerization results in less heat being generated and the reactor temperature decreases as well. For an inhibitor disturbance of four parts per 1000 (mole basis) in the fresh feed, Figure 9.41 shows the open-loop behavior of this process. Figure 9.42 indicates the futility of solely controlling temperature in response to this disturbance. The polymer properties are still significantly affected. In fact, the molecular weight actually deviates from its nominal value more with the temperature loop closed than with the temperature loop open.

Figure 9.43 shows the closed-loop performance comparison of the three control strategies in response to this severe disturbance. All three strategies maintain the polymer composition within its specification bounds, although the PI strategy is close to violating the upper limit. The PI scheme is slowest to bring molecular weight back to its performance specification, taking 21.25 h. The linear MPC scheme returns the molecular weight to the product quality requirements after only 17.25 h. The nonlinear MPC scheme, however, is able to keep the molecular weight within its performance bounds.

Figure 9.44 depicts the manipulated variable profiles for all three control schemes for the inhibitor disturbance simulation and reveals additional insight into the performance of all three control strategies. From Table 9.21, it is evident that y_1 is the fastest of the three outputs in response to changes in the inputs. Figures 9.41, 9.42, and 9.43 indicate that y_1 is also the fastest of the three outputs in response to the inhibitor disturbance. Since the inhibitor disturbance is an unmeasured disturbance, none of the three control strategies takes corrective action until at least one output deviates from its setpoint. The PI strategy is at a disadvantage for two reasons. First, this scheme does not recognize that one input can affect more than one output. In addition, corrective action for y_2 , for example, is taken only after y_2 deviates from its setpoint. MPC, however, first recognizes the need for corrective action when the fastest output (y_1) differs from its setpoint. The multivariable controller takes action to bring y_1 back to its setpoint, recognizing that the inputs moved to accomplish this goal also affect other outputs as well. Hence, an advantage of a multivariable controller is that it may be able to reject disturbances in all of the outputs as fast as the disturbance can be rejected in the fastest output. The profiles for the fresh feed of chain transfer agent in Figure 9.44 saturate at some point throughout this simulation for all three control schemes. The PI controller is coded with anti-windup to deal with the input saturation. The profile for the PI controller, depicted by the dash-dot line, is sluggish because the PI controller does not move this manipulated variable until the molecular weight deviates from its setpoint. The flow rate of chain transfer agent in the fresh feed using linear MPC is more aggressive because the multivariable controller begins to take correc-

tive action once y_1 deviates from its setpoint. Nonlinear MPC also begins to take corrective action once y_1 differs from its setpoint value. However, the improved accuracy of the AR-Volterra model enabled the controller to take more appropriate control action, which resulted in keeping the molecular weight within its specification bounds in response to this disturbance. This case study showed performance improvement in going from a multiloop PI control strategy to a multivariable linear MPC scheme to a multivariable nonlinear MPC scheme.

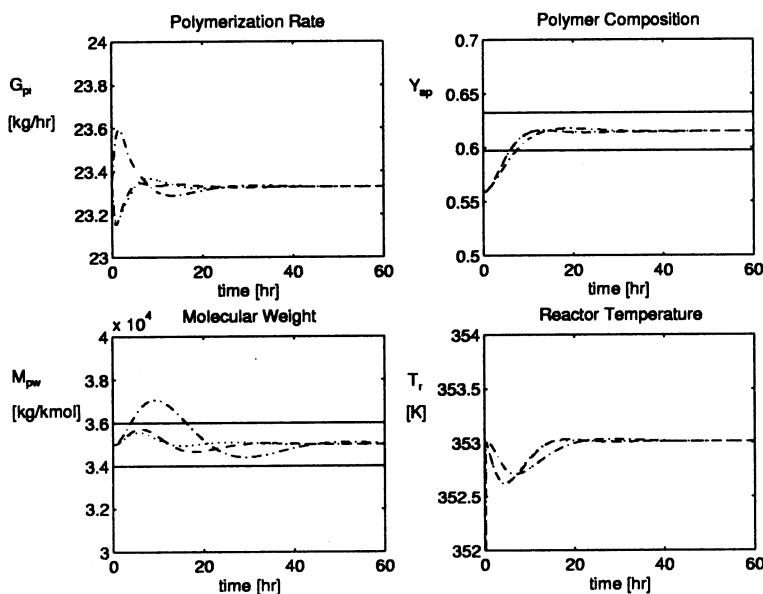


Fig. 9.39. Closed-loop simulation for a filtered step setpoint change of +10% in copolymer composition. Four PI controllers: dash-dot lines; one PI controller and 3 \times 3 linear MPC: dashed lines; one PI controller and 3 \times 3 MPC with nonlinear model for M_{pw} : dotted lines

9.5 Case study V – biochemical reactor

9.5.1 Process description

The process under consideration in this section is a continuous-flow bioreactor in which *Klebsiella pneumoniae* is grown on a glucose substrate. The mathematical model described by Baloo and Ramkrishna (1991a) and Baloo and Ramkrishna (1991b) is employed. The model falls into the class of cybernetic models, which are derived from the following principle (Kompala, 1984):

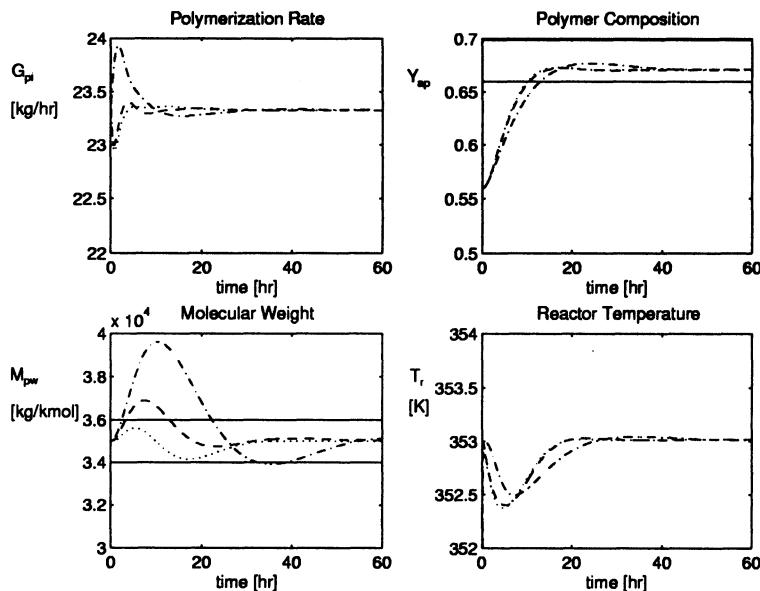


Fig. 9.40. Closed-loop simulation for a filtered step setpoint change of +20% in copolymer composition. Four PI controllers: dash-dot lines; one PI controller and 3×3 linear MPC: dashed lines; one PI controller and 3×3 MPC with nonlinear model for M_{pw} : dotted lines

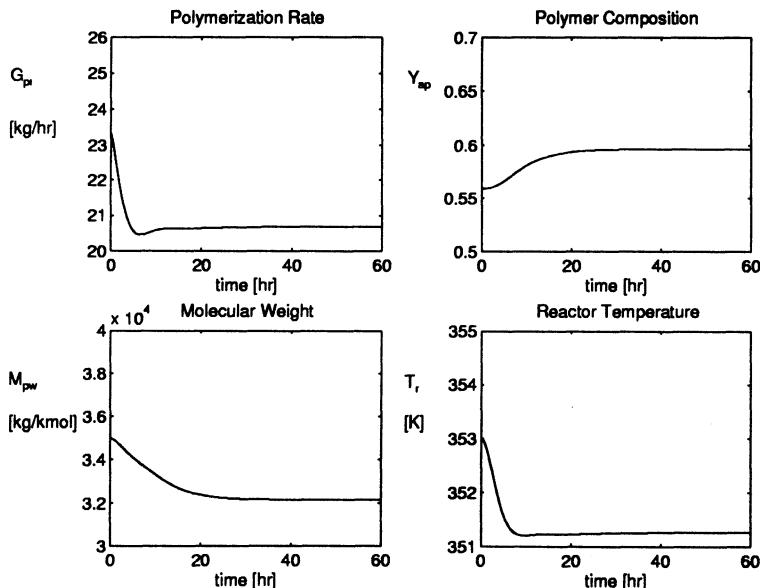


Fig. 9.41. Open-loop simulation for an inhibitor disturbance of four parts per 1000 (mole basis)

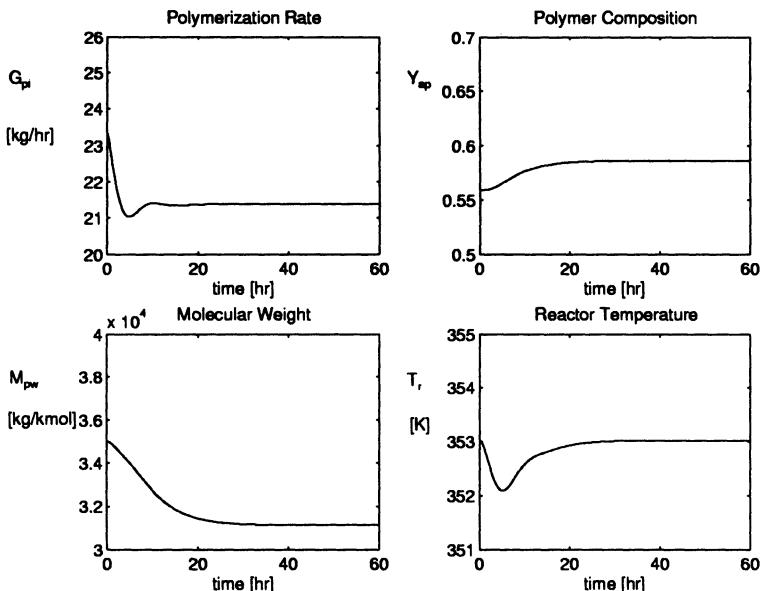


Fig. 9.42. Open-loop simulation for an inhibitor disturbance of four parts per 1000 (mole basis) with temperature feedback control

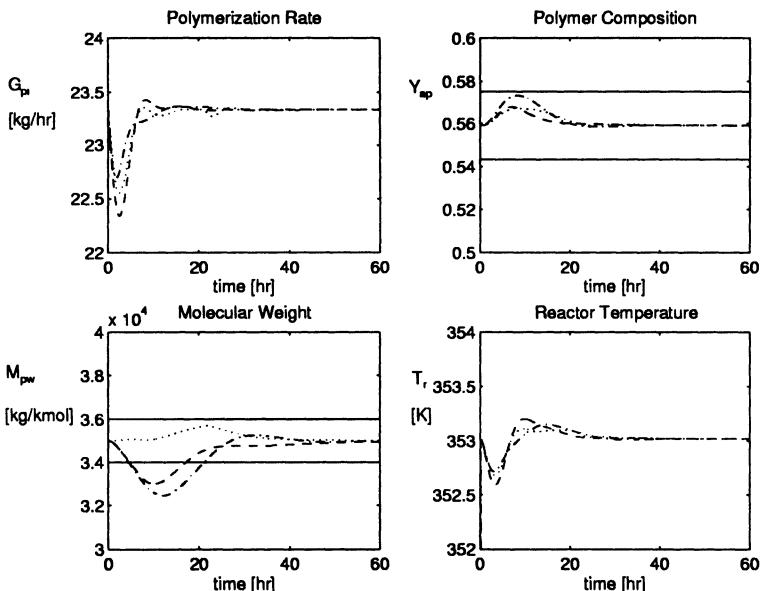


Fig. 9.43. Closed-loop simulation for an unmeasured step disturbance in inhibitor from its nominal value of zero to four parts per 1000 (mole basis). Four PI controllers: dash-dot lines; one PI controller and 3×3 linear MPC: dashed lines; one PI controller and 3×3 MPC with nonlinear model for M_{pw} : dotted lines

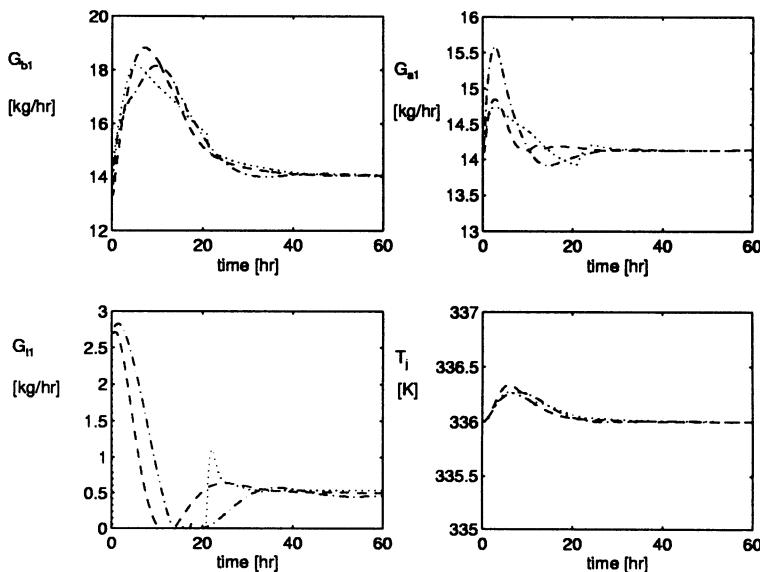
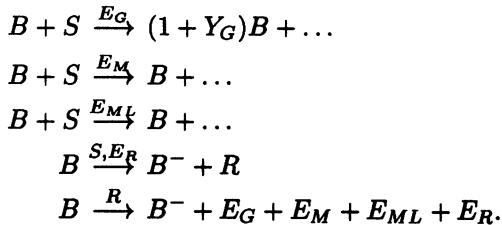


Fig. 9.44. Manipulated variable profiles for an unmeasured step disturbance in inhibitor from its nominal value of zero to four parts per 1000 (mole basis). Four PI controllers: dash-dot lines; one PI controller and 3×3 linear MPC: dashed lines; one PI controller and 3×3 MPC with nonlinear model for M_{pw} : dotted lines

Cybernetic modeling basic tenet: metabolic processes within cells are regulated in an optimal manner by the cell (developed through evolution), and an accurate description of the overall system can be formed if the net outcome of the process is captured, without knowing the full reaction mechanism.

Using this framework to derive a model of the bioreactor yields a pathway based on “key enzymes” and Monod-type rate expressions with parameters, such as the specific yield and specific growth rate, identified from actual process data (steady state-loci and dynamic responses). The cybernetic model reaction pathway for *K. pneumoniae* growth on glucose was given by Baloo and Ramkrishna (1991a):



Here B is biomass and S is the glucose substrate, which, in the presence of the growth enzyme E_G , yields additional biomass as a function of Y_G , the specific yield. Depending on the current glucose concentration within the bioreactor, either the maintenance (E_M) or low-glucose maintenance (E_{ML}) pathway

is followed to maintain cell biomass viability. Through biomass degradation (B^-), in the presence of the glucose substrate and resource enzyme E_R , a cellular resource (R) is produced. This resource, with further biomass reduction, is used to produce the necessary enzymes to catalyze the first four reactions. The resulting nonlinear differential equation model has the following form (Baloo and Ramkrishna, 1991a):

$$\begin{aligned}\dot{c} &= (r_G - D)c \\ \dot{e} &= r_E - e(\beta_E + r_G) \\ \dot{e}_{ML} &= r_{E_{ML}} - e_{ML}(\beta_{E_{ML}} + r_G) \\ \dot{R} &= r_R - R(\beta_R + r_G) \\ \dot{s} &= D(s_F - s) - \left(\frac{r_G}{Y_G} + r_{TM}v_M \right) c.\end{aligned}$$

The states of this model represent biomass concentration, growth/maintenance enzyme, low maintenance enzyme, cellular resource, and substrate (glucose). Expressions for the rates of enzyme synthesis (r_i) are given by:

$$\begin{aligned}r_G &= \mu_G^{\max} \frac{e_G}{e_G^{\max}} \frac{s}{K_s + s} \\ r_{TM} &= r_M + r_{ML} \\ r_M &= \mu_M^{\max} \frac{e_M}{e_M^{\max}} \frac{s}{K_M + s} \\ r_{ML} &= \mu_{ML}^{\max} \frac{e_{ML}}{e_{ML}^{\max}} \frac{s}{K_M + s} \\ r_E &= \alpha_E^* + \alpha_E \frac{s}{K_E + s} \frac{R}{K_R + R} u_G \\ r_{E_{ML}} &= \alpha_{E_{ML}}^* + \alpha_{E_{ML}} \frac{s}{K_{E_{ML}} + s} u_M \\ r_R &= \alpha_R \frac{e_R}{e_R^{\max}} \frac{s}{K_{RS} + s} v_R.\end{aligned}$$

Enzyme levels (e_i) have the mathematical form:

$$\begin{aligned}e &= e_M = e_G = e_R \\ e^{\max} &= \frac{\alpha_E + \alpha_E^*}{\beta_E + \mu_G^{\max}} \\ e_{ML}^{\max} &= \frac{\alpha_{E_{ML}}^* + \alpha_{E_{ML}}}{\beta_{E_{ML}}}.\end{aligned}$$

The cybernetic variables are governed by:

$$u_G = \frac{\frac{r_G}{Y_G}}{\frac{r_G}{Y_G} + r_{TM}v_M}$$

$$u_M = \frac{r_{TM}v_M}{\frac{r_G}{Y_G} + r_{TM}v_M}$$

$$v_M = 1 - \frac{r_G}{\mu_G^{max}}$$

$$v_R = \frac{r_G}{\mu_G^{max}}.$$

The remaining parameters in the model, defining Michaelis saturation, degradation, and other constants, have the values shown in Table 9.24. The cyber-

Table 9.24. Parameter values for the *K. pneumoniae* bioreactor

| | | |
|--|---|---|
| $\mu_G^{max} = 1.223 \text{ h}^{-1}$ | $K_s = 0.003 \text{ g l}^{-1}$ | |
| $Y_G = 0.51 \text{ gdw g}^{-1}$ | $K_{RS} = 0.003 \text{ g l}^{-1}$ | |
| $\mu_M^{max} = 0.17 \text{ h}^{-1}$ | $K_M = 1 \times 10^{-6} \text{ g l}^{-1}$ | |
| $\mu_{ML}^{max} = 0.39 \text{ h}^{-1}$ | | |
| $\beta_E = 0.05 \text{ h}^{-1}$ | $K_E = 0.003 \text{ g l}^{-1}$ | $\alpha_E = 0.001 \text{ h}^{-1}$ $\alpha'_E = 8 \times 10^{-5} \text{ h}^{-1}$ |
| $\beta_{EML} = 6.0 \text{ h}^{-1}$ | $K_{EML} = 1 \times 10^{-6} \text{ g l}^{-1}$ | $\alpha_{EML} = 0.05 \text{ h}^{-1}$ $\alpha'_{EML} = 8 \times 10^{-6} \text{ h}^{-1}$ |
| $\beta_R = 2.5 \text{ h}^{-1}$ | $K_R = 0.003 \text{ g l}^{-1}$ | $\alpha_R = 0.125 \text{ h}^{-1}$ |

netic modeling structure differs from most bioreactor and organism models in that it attempts to capture the inherent regulatory behaviors of the cell along with the necessary reaction kinetics. Regulation is governed by cybernetic variables representing rates of enzyme activity (v) and synthesis (u). The interested reader is referred to Kompala et al. (1986) or Baloo and Ramkrishna (1991a) for further details regarding cybernetic modeling.

The continuous process has a single controlled output (cell biomass exit concentration, g l^{-1}) and two inputs: the manipulated variable dilution rate (h^{-1}), and the disturbance substrate (glucose) feed concentration (g l^{-1}). A schematic of the process is shown in Figure 9.45.

The challenging control problem associated with this bioreactor is best seen by examining the steady-state and dynamic behaviors for the system. The steady-state operating locus displays a change in the sign of the gain at a dilution rate of 0.895 h^{-1} as shown in Figure 9.46. In addition to the critical point in the steady-state locus, Figure 9.47 demonstrates the nonlinear dynamic behavior of the process. This asymmetric response to symmetric inputs is clearly a further challenge for control algorithms. Not evident from the figures is the transition from minimum phase to nonminimum phase dynamics as the operating conditions change from the high dilution rate side of the optimum to dilution rates $< 0.895 \text{ h}^{-1}$. Given the above behaviors, the nominal process operating condition is on the high dilution rate portion of the curve

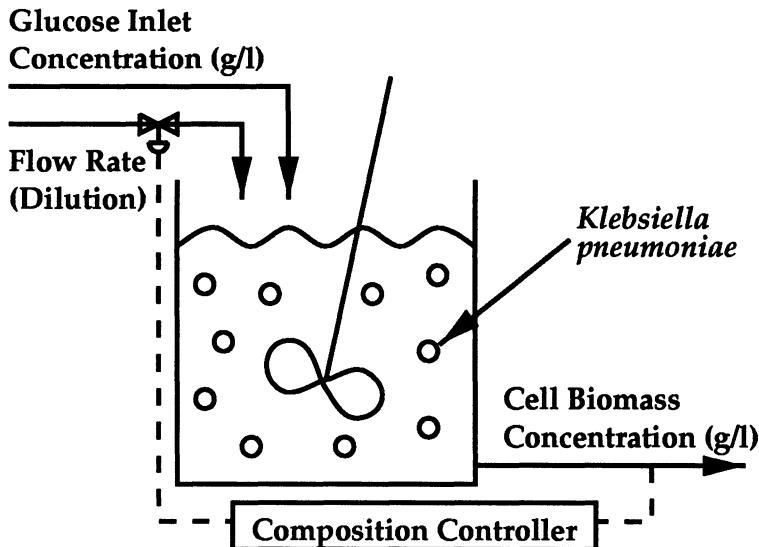


Fig. 9.45. Process schematic for the bioreactor

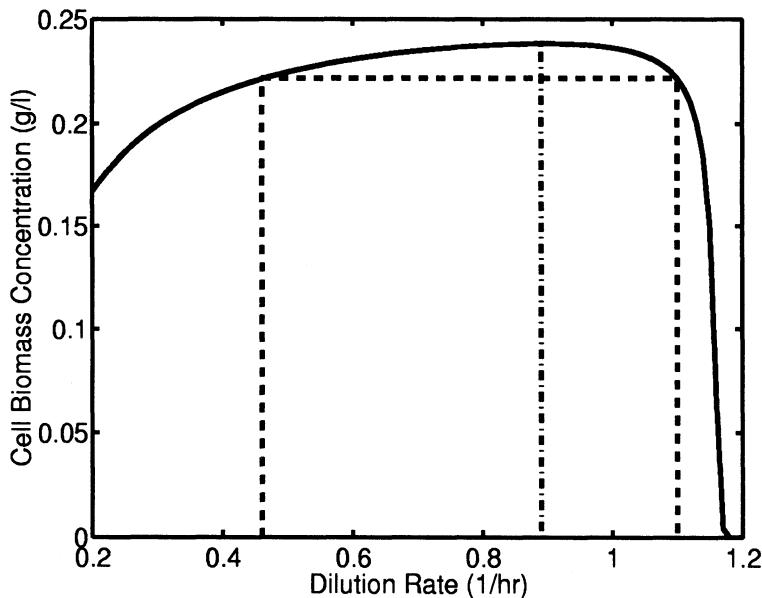


Fig. 9.46. Steady-state locus for the bioreactor (solid curve). Also shown are an input multiplicity (dashed lines) and the location of the process optimum (vertical dash-dot line)

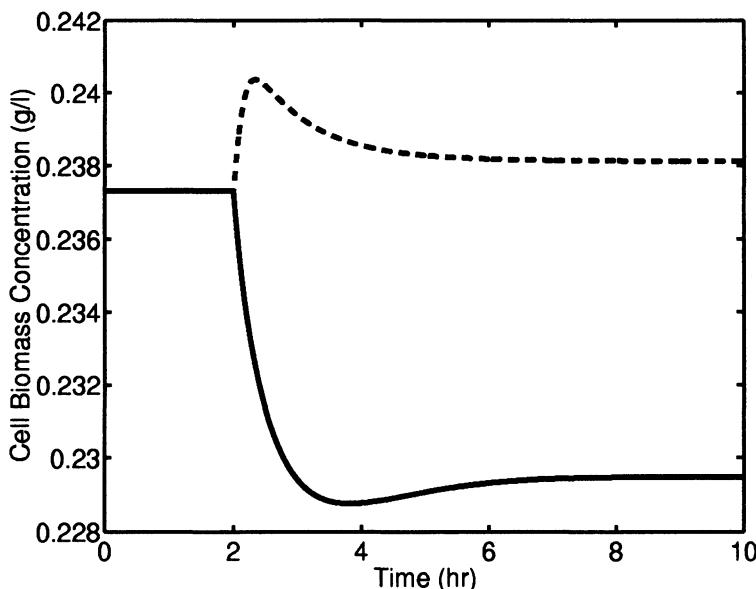


Fig. 9.47. Asymmetric bioreactor response to symmetric steps of $+0.1 \text{ h}^{-1}$ (solid) and -0.1 h^{-1} (dashed) at time 2 h

to avoid the closed-loop performance degradation that occurs in the presence of nonminimum phase dynamics (Skogestad and Postlethwaite, 1996).

9.5.2 Volterra-Laguerre IMC and MPC design

A second-order Volterra-Laguerre model is identified from the “unknown” bioprocess model, using the procedure described in Parker et al. (2001). From the identified model, a partitioned nonlinear inverse (PNLI) controller in the NIMC framework (Chapter 6) and an MPC algorithm (Chapter 7) are synthesized via the methods outlined in the noted chapters. These control algorithms were implemented in closed-loop, and analyzed for performance in continuous-flow bioreactor regulation studies.

Based on organism growth characteristics, the implementation in closed-loop includes input magnitude constraints to guarantee cell viability within the reactor. The minimum dilution rate is 0.2 h^{-1} since the cells starve due to lack of glucose at lower values. Additionally, if the residence time in the reactor is too small, wash-out occurs, as demonstrated by the zero cell mass exit concentration at dilution rates in excess of 1.16 h^{-1} . For the PNLI controller, these constraints are implemented by clipping the manipulated variable move calculated by the controller. The MPC problem is solved with full knowledge of the constraints.

For testing in closed-loop, performance in disturbance rejection and reference tracking are examined simultaneously. Disturbances shift the steady

state locus in a vertical manner, increasing or decreasing the maximum achievable cell biomass exit concentration. Upward shifts are easily handled, and are related to a decrease in the setpoint, although the dynamics are different. Downward shifts in the steady-state locus create “unreachable” setpoints, located above the maximum achievable steady-state cell biomass concentration, and can be simulated through increases in the reference value. In this way, disturbance rejection and setpoint tracking can be evaluated through an examination of only the latter. The controller objective is setpoint tracking for “reachable” setpoints, and tracking the model maximum otherwise.

Simulation results—IMC controller. Figure 9.48 shows the closed-loop behavior of reference profile tracking. The partitioned nonlinear inverse con-

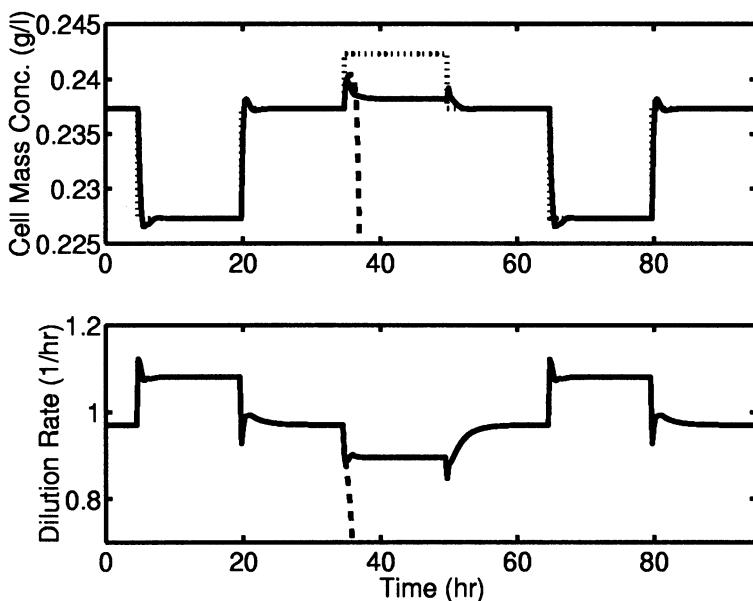


Fig. 9.48. PNLI controller response to a setpoint profile (dotted). Shown are PNLI controller (dashed), and PNLI controller with reference triggered switching algorithm (solid). Top: cell biomass concentration. Bottom: calculated dilution rate

troller easily handles the negative step change (akin to disturbances producing “reachable” setpoints). However, when faced with an “unreachable” setpoint, the controller becomes unstable. The reference value causes the linear inverse-based controller to calculate a manipulated input decrease, which in turn results in an increase in the reference value as the nonlinear feedback term is negative definite. This positive feedback loop drives the system to its lower input magnitude constraint, from which recovery is only possible by reinitializing the controller.

To handle this destabilizing behavior, a reference signal-based switching algorithm was employed, based on the nonlinear identified model maximum value. Alternative, and more complicated, methods for controlling systems requiring discontinuous feedback have been examined by Sontag and co-authors (Clarke et al., 1997; Sontag, 1999). If the reference signal is less than the Volterra-Laguerre model maximum, the nonlinear feedback term is subtracted from the incoming reference signal. For “unreachable” setpoints, stabilizing feedback is returned if the sign of the nonlinear feedback signal is changed. The interpretation of this switching behavior is that the nonlinear correction to the linear calculated input move should always be toward the optimum. One drawback of this approach is the inability to track the optimal value for “unreachable” setpoints, as the nonlinear feedback term induces some offset from the true process optima. The degree of controller performance sub-optimality scales directly with the relative difference in magnitude between the “unreachable” reference and the achievable maximum. Performance for this controller in the mismatch case is shown in Figure 9.48 as the solid line. Attempting to track the same reference profile as above, the switching algorithm maintains stability in the presence of “unreachable” setpoints, and behaves identically to the previous algorithm for reachable references.

Figure 9.48 also demonstrates the stability of the PNLI controller across the process optimum. An examination of the unreachable setpoint response, and the corresponding return to a reachable reference value (time values from 35 to 62 h) is shown as a phase diagram in Figure 9.49. From the initial point at $(u, y) = (0, 0)$ corresponding to $t = 35$ h, an unreachable setpoint of +0.5 (in scaled units) is implemented. Following the arrows, the controller undergoes a brief transient to the left-hand side of the optimum, given by the vertical dash-dot line. This is corrected quickly, and the process proceeds to a steady state at $(-0.934, 0.088)$. Recall, however, that the magnitude of the unreachable reference can dramatically affect closed-loop optimum tracking performance, as the system will not necessarily track the process optimum for different reference signals. At time $t = 50$ h a reference of $r = 0$ is selected, and the controller makes a series of aggressive manipulated variable moves. These are to the left of the process optimum, but the controller recovers such that the closed-loop system steady state matches the reference by $t = 62$ h. Note that the switching algorithm controller and the *nominal* controller are stable across the optimum. The first transient optimum crossing is during an unreachable setpoint change; hence, the switching algorithm is “on,” and, as expected, the controller is stable near the optimum output value. Once the reference is returned to $r = 0$, however, the switch is “off,” and the originally derived controller remains stable as it drives the closed-loop system to an offset-free steady state, even while implementing moves to the left side of the optimum.

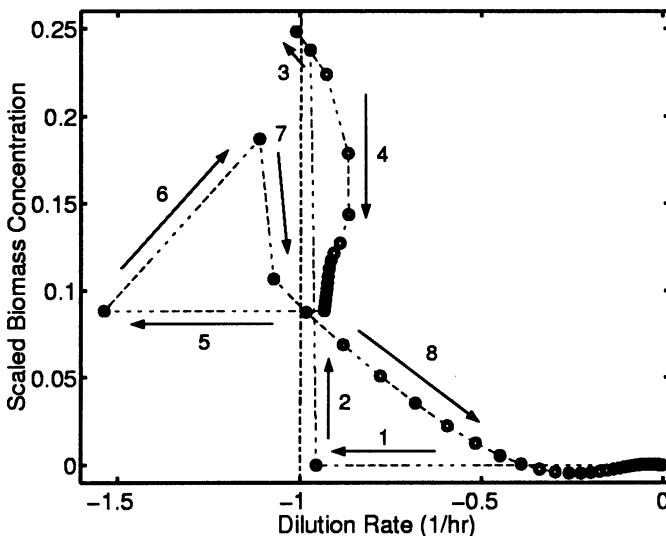


Fig. 9.49. Phase diagram of the PNLI controller given an unreachable setpoint and returning to a reachable reference. Initial point ($t = 35$ h): $(u, y) = (0, 0)$; unreachable steady state ($t = 49$ h): $(-0.934, 0.088)$; final point, ($t = 62$ h): $(0, 0)$

Simulation results—MPC controller. To demonstrate the utility of the nonlinear MPC methods over the PNLI controller, the nonlinear MPC algorithms are challenged with the same setpoint profile that was presented in Figure 9.48. The MPC tuning parameters are $m = 1$, $p = 16$, $\Gamma_u = 0$, and $\Gamma_y = 1$. Tracking results are shown in Figure 9.50.

Performance for the NLP MPC controller degrades after being subjected to an unreachable setpoint. The controller is unable to remain on the minimum-phase side of the optimal point, and crosses to the low dilution rate side in response to the unreachable reference. This crossing is due to an initial aggressive manipulated variable change based on the current objective function gradient, followed by the calculation of a new gradient of opposite sign. The controller continues to track the reference, but is trapped on the low-dilution rate side of the optimum. The ability of the NLP MPC controller to track further step changes is hindered by its operation on the nonminimum phase side of the operating locus. The analytical NMPC controller, however, is able to track the reachable portions of the profile with little over- and under-shoot. Additionally, it moves the process output to within < 1% of the true optimum when faced with the unreachable setpoint shown.

Controller tuning for MPC systems is often an ad hoc procedure, but performance requirements can lead to heuristic bounds on parameters. Zheng (1998) explains that integral control for NMPC systems is guaranteed only if a global solution is implemented and the prediction horizon is infinite.

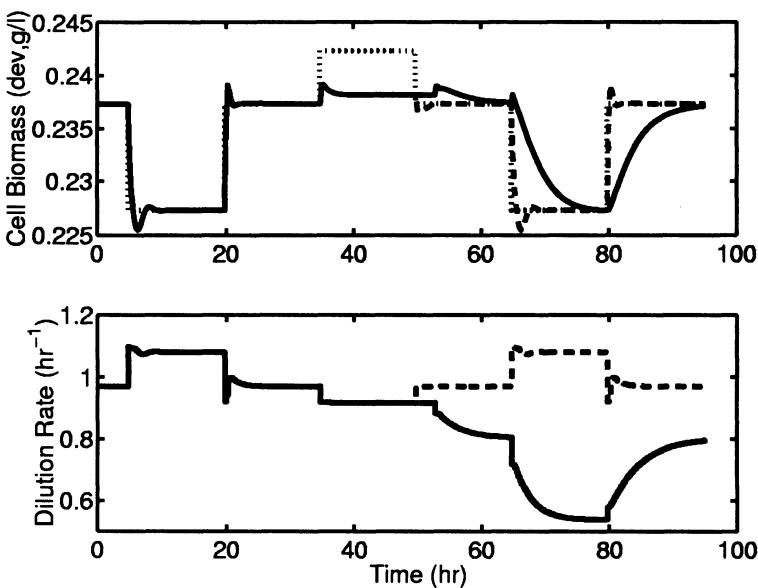


Fig. 9.50. Nonlinear MPC controller response to tracking a setpoint profile. Top: cell biomass concentration; reference (dotted), NLP MPC (solid), and analytical NMPC (dashed). Bottom: dilution rate; NLP MPC (solid) and analytical NMPC (dashed)

A fading memory system, such as a Volterra-Laguerre or Laguerre model, requires an infinite prediction horizon for guaranteed integral control. A minor adjustment for finite memory systems (like a Volterra model) is that *nominal case* integral tracking requires p equal the model memory. As an example, the nominal extremum tracking case with an unreachable reference of $r = +0.5$ (scaled) is examined. Controllers tuned with short prediction horizons ($p = 4$) are unable to track the optimal biomass concentration. Several objective function contours, shown in Figure 9.51, demonstrated the convexity of the unreachable setpoint problem, and also the shift in objective function minimum with increasing prediction horizon.

The dilution rate corresponding to the minimum objective function value shifts from 0.87 h^{-1} for $p = 4$ to the near-optimal 0.92 h^{-1} for $p = 16$. This represents a steady-state tracking error decrease from 5 mg l^{-1} to 0.3 mg l^{-1} (94% change). Improvement beyond $p = 16$ can be made in tracking the true output maximum (tracking error $\approx 0.1 \text{ mg l}^{-1}$ for $p = 40$, and is $\approx 0.06 \text{ mg l}^{-1}$ for $p = 99$), although the error decrease does not offset the increased computational demand (an incremental 5% decrease in error for a two-fold to six-fold increase in p). Therefore, the prediction horizon in nonlinear MPC is selected such that the sign of the predicted output is accurate *and* the approximate steady-state gain is captured. This is in agreement with the results of Zheng (1998). The reader should note, however, that mismatch

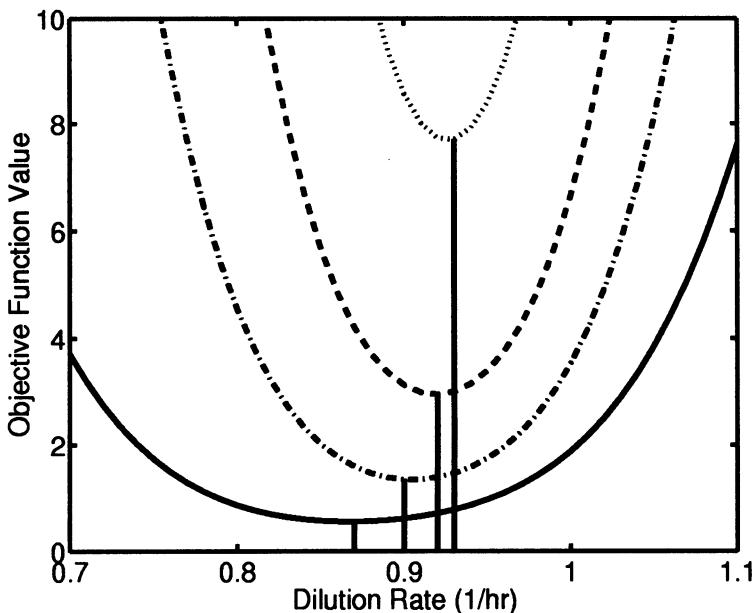


Fig. 9.51. Objective function contours for an $r = +0.5$ ($\bar{r} = +0.005 \text{ g l}^{-1}$) setpoint change. Solid: $p = 4$; dash-dot: $p = 10$; dashed: $p = 16$; and dotted: $p = 40$

between the plant and the empirical model typically precludes *exact* tracking of the plant maximum in the unreachable case, such that small deviations around the true maximum are acceptable. In this way, extremum control can be accomplished by using only the Volterra-Laguerre model without enforcing strict geometric criteria, as in Golden and Ydstie (1989).

In the case of unreachable setpoints, the controller should minimize the error between the reference and the process output by moving to the process optimum. Figure 9.52 shows the nonlinear MPC controller responses to an unreachable setpoint $r = +0.5$.

The controller of Dumont et al. (1994) utilizes the asymptotic analytical solution to the nonlinear MPC problem with only an end-point penalty in the objective function. Improvement in extremal tracking is seen when the full analytical solution, including dynamic error compensation, is implemented. To correct for the plant-model mismatch, a recursive-least-squares (RLS) update is incorporated into the analytical NMPC algorithm, and this controller output (0.897 h^{-1}) approached the true nonlinear process optimum dilution rate of 0.895 h^{-1} at steady-state.

Reachable setpoints showed different objective function behavior, and sample contours are displayed in Figure 9.53. Shown are the multiple extrema expected when solving fourth-order objective function problems. Clearly, with increasing prediction horizon, the barrier between the two minima increased. Gradient descent-based NLP solvers can converge to a local minimum, and

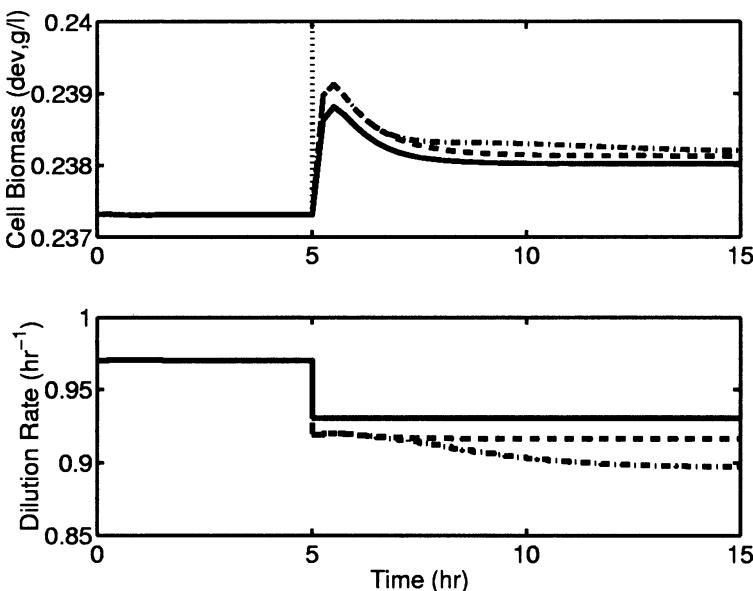


Fig. 9.52. Nonlinear MPC controller response comparison for an $r = +0.5$ (unreachable) reference. Top: cell biomass concentration; reference (dotted), analytical NMPC (dashed), analytical NMPC with RLS (dash-dot), and literature analytical controller using the asymptotic solution (solid). Bottom: dilution rate; analytical NMPC (dashed), analytical NMPC with RLS (dash-dot), and literature controller (solid)

the analytical solution controller guarantees convergence to the global minimum objective function value. This performance degradation is complicated further by process constraints.

In every case, the analytical controller performed *at least* as well as the NLP MPC controller for the nominal problem. In implementing this algorithm on the real process, magnitude constraints on $u(k)$ would be enforced, and for demonstrative purposes the constraint set ($0.2 \text{ h}^{-1} \leq u(k) \leq 1.1 \text{ h}^{-1}$) is selected. A setpoint change of -0.05 g l^{-1} is tested in the constrained nominal case. Results for NLP MPC and the analytically-derived controller are shown in Figure 9.54.

Similar results can be demonstrated for the mismatch case, although rigorous guarantees of tracking and constraint handling have not yet been derived. A comparison of the NLP MPC and analytical controllers to a setpoint change of -0.05 g l^{-1} for the real process is shown in Figure 9.55.

In both cases the NLP MPC controller becomes entrapped in the local minimum at the upper constraint, whereas the analytical NMPC algorithm determines the *constrained local* minimum objective function value and successfully tracks the reference. The objective function contours for this problem have qualitative characteristics similar to those in Figure 9.53, except

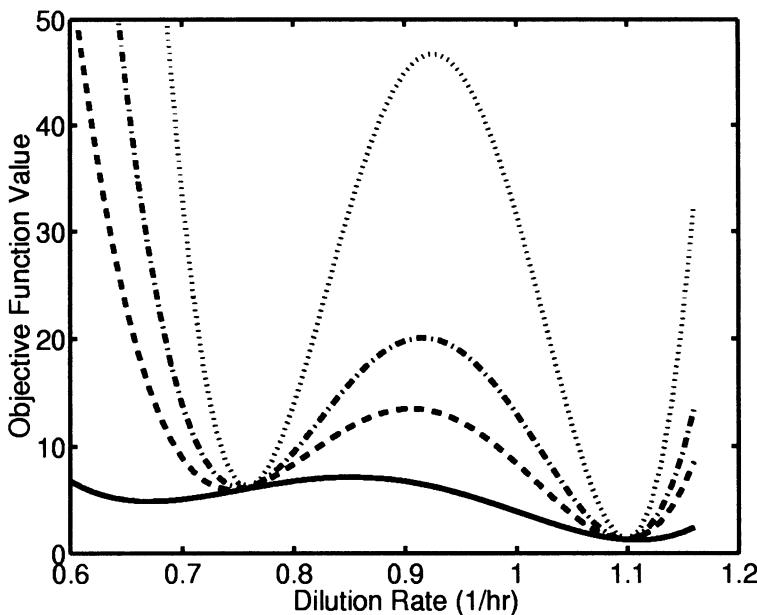


Fig. 9.53. Objective function contours for -0.01 g l^{-1} setpoint change, with varying prediction horizon. Solid: $p = 4$; dashed: $p = 10$; dash-dot: $p = 16$; and dotted: $p = 40$

the low dilution rate well is the global minimum. Performance degradation is seen due to the nonminimum phase characteristics of the process for dilution rates $< 0.895 \text{ h}^{-1}$. To improve the closed-loop response in the mismatch case, an RLS algorithm with $\eta = 0.99$ and $P_{RLS}(0) = 2.5 \times 10^{-4} * I_{9 \times 9}$ is utilized in conjunction with the analytical NMPC controller. Significant performance improvements are observed, such as reduced settling time, reduced rise time, and decreased undershoot.

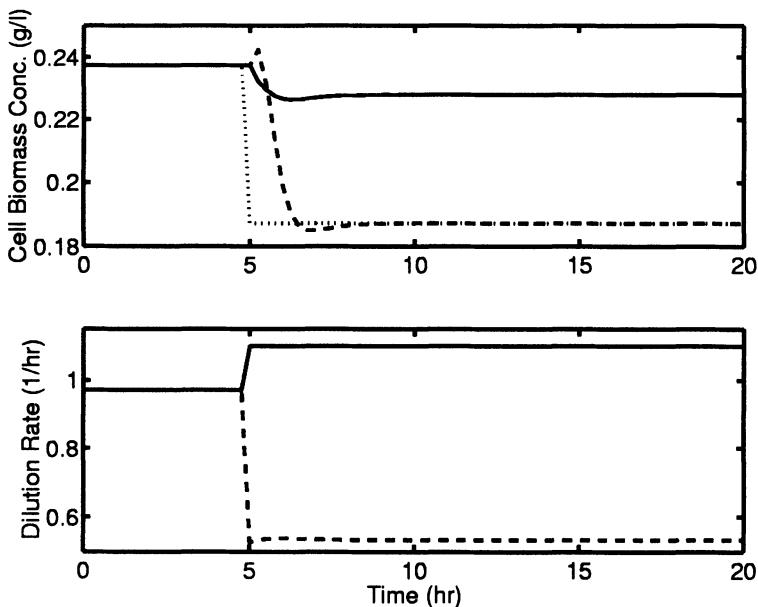


Fig. 9.54. Setpoint tracking for a -0.05 g l^{-1} setpoint change (constrained nominal case); reference (dotted), NLP MPC (solid), and analytical solution controller (dashed)

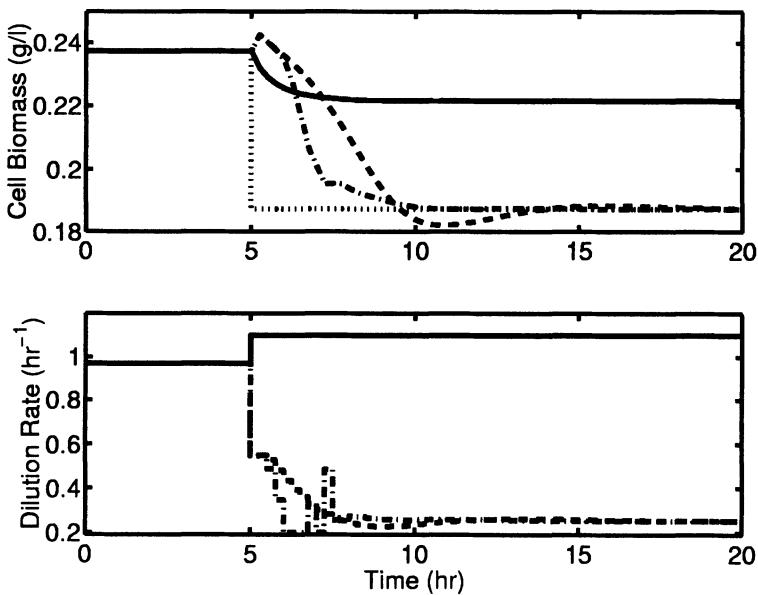


Fig. 9.55. Setpoint tracking for a -0.05 g l^{-1} setpoint change; reference (dotted), NLP MPC (solid), analytical MPC (dashed), and analytical MPC with RLS (dash-dot)