Polymerization Reactor Control Using Autoregressive-Plus Volterra-Based MPC

Bryon R. Maner and Francis J. Doyle III
School of Chemical Engineering, Purdue University, West Lafayette, IN 47907

A nonlinear model-predictive control scheme based on the autoregressive-plus Volterra model is proposed. A "plant-friendly," yet persistently exciting, input sequence is designed for model identification. A second feature of this work is that the control action is determined by solving a less computationally burdensome nonlinear programming problem than the optimization problems associated with Newton-type controllers and polynomial ARMA model nonlinear MPC. A third contribution is that semiglobal closed-loop stability conditions are derived and are shown to be less conservative than those previously published. The control scheme is shown to outperform a feedback strategy using proportional integral control and a linear model-predictive control design for the control of two polymerization reactor case studies. The complex features of the latter case study [multivariable (3 × 3), recycle loop, high-order (24) dynamics] motivate the applicability of this approach for industrial problems.

Introduction

The manufacture of synthetic polymers represents an important part of the chemical industry, with a production of over 100 billion kg annually (Ray, 1986). The control of polymerization reactors is a difficult and complex problem for several reasons. First, these reactors often exhibit highly interactive nonlinear dynamic behavior. For example, the existence of steady-state multiplicities, parametric sensitivity, and limit cycles for the free-radical polymerization of some monomers in continuous stirred-tank reactors (CSTRs) has been shown theoretically and experimentally by several researchers (Hamer et al., 1981; Schmidt and Ray, 1981; Schmidt et al., 1984). In many chemical plants, a single polymerization reactor may be required to produce polymers of different grades (various molecular weights, compositions, etc.). Hence, one control objective is minimization of grade transition time, resulting in reduced off-specification product. However, the interactive nonlinear dynamic behavior of these reactors becomes more apparent during grade transitions than at steady-state operation. Hence, a multivariable, nonlinear model-based controller may yield improved performance over a linear model-based controller for grade transition control. A second challenge is that a first-principles model of a polymerization reactor may contain a large number of kinetic parameters. For example, one of the models used in this case study contains 38 kinetic parameters. Obtaining these parameters from the literature or from lab-scale and pilot-plant work can be a very time-consuming endeavor. In addition, there may be processes for which an understanding of the kinetic mechanism is not available. It may then be advantageous to use a model structure whose parameters may be identified from input–output data.

There have been several approaches to the explicit use of a nonlinear model to compute the control action for process control applications. Economou et al. (1986) considered using the inverse of a nonlinear model in the internal model control (IMC) framework for control of a nonlinear process. They first considered the analytical construction of the inverse using the results of Hirschorn (1979), but found that this approach failed to produce a satisfactory approximation to the exact inverse. Successive substitution and Newton's method were then considered for the numerical computation of the inverse. Newton's method was used for the control of a one-input, one-output, three-state system. The inverse controller used would have the model predictive control (MPC) tuning parameters for input move horizon and prediction horizon of one. A filter in the feedback path gave satisfactory results in terms of performance and robustness.

Li and Biegler (1988) extended the Newton-type controller to handle state and control variable constraints and applied the controller to a two-input, two-output, two-state system

Correspondence concerning this article should be addressed to F. J. Doyle III.

and a two-input, two-output, three-state system. Li and Biegler (1989) extended the controller to multistep prediction to handle systems with time delays. They demonstrated the performance of the resulting controller for prediction horizons of 2 and 3 for two three-state problems. Parameter estimation was combined with the controller to deal with the case of plant–model mismatch (Li and Biegler, 1990). Its performance was illustrated for the control of a two-state pH control problem. Eaton and Rawlings (1990) also used a first-principles model and solved a general nonlinear program to compute the control action.

One disadvantage of the previous approaches is that the computational load scales directly with the number of state equations in the model. One of the case studies considered in this article consists of 24 state equations and would pose a considerable computational challenge for these controllers for on-line implementation. A second drawback, as mentioned previously, is that these approaches rely on the availability of a first-principles model.

A more computationally tractable approach to nonlinear control that has appeared in the literature consists of linearizing a nonlinear model at each time step and computing the control action based on the resulting linear model. García (1984) first proposed this approach and applied it to the control of a semibatch reactor. Gattu and Zafiriou (1992) applied the algorithm to the control of an unstable CSTR. Ricker and Lee (1995) demonstrated the performance of this approach in the control of a plantwide control problem. Although this algorithm is attractive from a computational viewpoint, the noted implementations utilized mechanistic models.

A wide variety of nonlinear input–output models have been proposed for use in control of nonlinear processes. A significant advantage of these models is that they may be identified from input–output data. Yeo and Williams (1987) proposed using bilinear models in an MPC scheme and performed simulations for control of bilinear plants. Bartee and Georgakis (1992) identified a bilinear model of a CSTR using input–output data and implemented the model in a reference-system control structure. Eskinat et al. (1991) identified Hammerstein models to describe a binary distillation column and heat exchanger. Fruzzetti et al. (1995) used a Hammerstein model in a nonlinear MPC scheme for control of reactor pH and a binary distillation column. Zhu and Seborg (1994) proposed a modified Hammerstein model and illustrated its use for the control of reactor pH. Chien and Ogunnaike (1992) proposed an empirical low-order nonlinear model for describing a high-purity distillation column. The model time constant was a function of the output, and the gain was a function of the output and input. Hernández and Arkun (1993) proposed a nonlinear MPC scheme based on polynomial autoregressive moving-average (ARMA) models. The performance of the control scheme was investigated with state estimation and several disturbance models for a variety of case studies. Sriniwas et al. (1995) applied this control scheme to the nonlinear MPC of a high-purity distillation-column model developed by Weischedel and McAvoy (1980). The first-principles model contained 56 states, and the control problem involved two inputs and two outputs. Pearson (1994) used a Wiener model to describe the nonlinear behavior of a CSTR. Ogunnaike et al. (1994) proposed a low-order empirical model for describing the input–output behavior of single-input–single-output (SISO) nonlinear systems. The structure was obtained by taking a Taylor series expansion of a control-affine state model and retaining up to second-order terms. The modeling performance was demonstrated for two SISO reactors. Second-order Volterra models were used in the control of chemical reactors using the IMC framework (Doyle III et al., 1995). Second-order Volterra models were also incorporated in an MPC formulation for control of two chemical reactors (Maner et al., 1994). In the present work, we pursue a parsimonious formulation of the Volterra model, which is known to have general nonlinear approximation ability (Boyd and Chua, 1985). This model structure, denoted as an autoregressive plus Volterra model, was implemented in an MPC scheme for the simulated control of a distillation column (Kirnbauer and Jörgi, 1992).

Nonlinear Model Structure

This article considers a nonlinear MPC scheme based on the autoregressive plus Volterra model shown in Eq. 1:

$$y_i(k) = y_{i0} + \sum_{l=1}^{q_y}\sum_{j=1}^{n_y} \theta_{l,j}^{1,i} y_l(k-j) + \sum_{l=1}^{q_u}\sum_{j=1}^{n_u} \theta_{l,j}^{2,i}(k-j) + \sum_{l=1}^{q_u}\sum_{j=1}^{n_u}\sum_{m=l}^{q_u}\sum_{n=1}^{j} \theta_{l,j,m,n}^{3,i} u_l(k-j)u_m(k-n) + \sum_{l=1}^{q_u}\sum_{j=1}^{n_u}\sum_{m=l}^{q_u}\sum_{n=1}^{j}\sum_{o=m}^{q_u}\sum_{r=1}^{n} \times \theta_{l,j,m,n,o,r}^{4,i} u_l(k-j)u_m(k-n)u_o(k-r). \quad (1)$$

Equation 1 is a description for output i where q_y and q_u correspond to the number of outputs and inputs, respectively, and n_y and n_u pertain to the number of lags on the outputs and inputs, respectively. This model structure was first proposed by Kirnbauer and Jörgi (1992). The presence of the autoregressive terms results in a model that is much more parsimonious than the Volterra model. The autoregressive plus Volterra model structure is a subset of the polynomial ARMA model structure considered by Billings and Leontaritis (1982). The model is linear in the parameters, and its regressors and parameters may be identified from input–output data. Although these models cannot capture some of the nonlinear behavior described by polynomial ARMA models, such as output multiplicities, they offer several distinct advantages over polynomial ARMA models for use in nonlinear MPC, as discussed in the subsequent sections.

Identification

The input sequence used in past work (Hernández, 1992; Hernández and Arkun, 1993; Sriniwas et al., 1995) to identify polynomial ARMA models used a probability of switching factor, P_s, that corresponded to the probability that the input changed at the end of any sampling interval. If the input changed, its new value was drawn from a uniform distribution. There may be several disadvantages to this input se-

quence. For system identification, it is desirable to use an input sequence that excites the system evenly over the frequency range of interest. This would be evidenced by a power spectrum with a large magnitude over the frequency range of interest. The motivation for using an input sequence whose spectrum is flat is that the standard least-squares method determines the model parameters by minimizing the L_2-norm of the error weighted by the input spectrum (Ljung, 1987). In Figure 1, the power spectra of input sequences for various P_s values are shown. The spectra were generated numerically using the function spa.m in Matlab. The power at $\omega = 2.01$ rad/min for the $P_s = 0.1$ sequence denoted by the solid line is 0.0169. At this same frequency, the power for the $P_s = 0.3$ sequence given by the dashed line is 0.1126. The power of the sequence for $P_s = 0.8$ depicted by the dotted line at $\omega =$ 2.01 rad/min is 0.2918. Since the model parameters are determined by minimizing the L_2-norm of the error weighted by the input spectrum, models obtained using the higher probability of switching would result in models that are more accurate at higher frequencies. The power spectrum for $P_s =$ 0.8 is fairly constant over all frequencies. This is not surprising, since this sequence switches levels nearly as quickly as a sequence with $P_s = 1.0$, which is a sequence of independent, zero-mean random variables that would be the best choice for system identification. This sequence, however, is usually too demanding for practical implementation. The spectra for the $P_s = 0.1$ and $P_s = 0.3$ sequences contain the same trend in that they begin to roll off at low frequencies and gradually decay at higher frequencies. From an identification standpoint, it would be more desirable to use an input sequence whose power spectrum remains constant for a larger frequency range so that the resulting model would be more accurate over a larger frequency range as well. A second drawback of this input sequence is that the input would take on a very large number of values that may not be feasible in practice, because valves and other actuators have a minimum resolution with respect to positioning (Moore, 1992). Consider the case where a valve position of 53.45% is drawn from the uniform distribution and the corresponding steam valve has a resolution of $\pm 1\%$. It would not be possible to move the valve to the specified position. For this reason, it may be desirable to use an input sequence with a specified number of levels rather than an infinite number of levels, as would be used if values were drawn from a uniform distribution. Since many chemical processes have large time constants, it may also be desirable to incorporate a clock period, T_{cl}, that is an integer number of sample periods for which the input is held constant before possibly switching again, as is done in practice with pseudorandom binary sequence (PRBS) signals for linear system identification (Davies, 1970; McFarlane and Rivera, 1992).

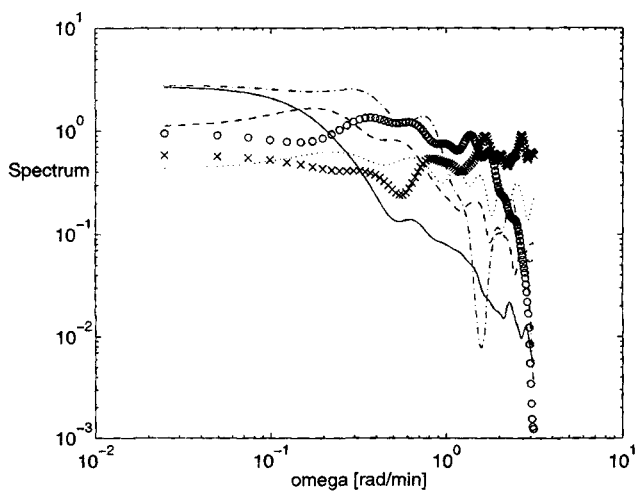

Figure 1. Power spectra of input sequences.

$P_s = 0.1$: solid line; $P_s = 0.3$: dashed line; $P_s = 0.8$: dotted line: $T_{cl} = 1$: x's line; $T_{cl} = 2$: o's line; $T_{cl} = 4$: dash-dot line.

Nowak and Van Veen (1994) proved that an $N+1$-level pseudorandom multilevel sequence is persistently exciting for a Volterra filter with nonlinearities up to polynomial degree N. The autoregressive plus Volterra model in Eq. 1 has an equivalent representation as a third-order Volterra model (Rugh, 1981). Hence, the autoregressive plus Volterra model can be identified with a four-level input sequence. In this work, a four-level random sequence was used rather than a four-level pseudorandom sequence due to the anomalies that may appear in the spectra for nonlinear system identification when pseudorandom sequences are used (Marmarelis and Marmarelis, 1978). In Figure 1, the power spectra of four-level input sequences for three T_{cl} values are also shown. These spectra were also generated numerically using the function spa.m in Matlab. The power at $\omega = 2.01$ rad/min for $T_{cl} = 4$ denoted by the dash-dot line is 0.1019. However, the spectrum has already rolled off for this T_{cl} value. The power for the $T_{cl} = 2$ sequence shown as the line represented by o's is 0.2595 at the same frequency. The spectrum of the $T_{cl} = 1$ sequence depicted by the line of x's is 0.5939 at $\omega = 2.01$ rad/min. Hence, models of the form in Eq. 1 obtained using these input sequences would be more accurate at higher frequencies than those obtained using sequences with $P_s = 0.1$ or $P_s = 0.3$. The power spectrum for $T_{cl} = 1.0$ is approximately equal to one over all frequencies as expected, since this sequence corresponds to a completely random four-level input sequence. While the $T_{cl} = 1.0$ input sequence would probably be too demanding for many practical applications, it is less demanding than the sequence obtained using $P_s = 1.0$ and selecting input levels from a uniform distribution. The reason for this is that the four-level input sequence using $T_{cl} = 1.0$ has a one-in-four chance that the input will remain at the same level. The other sequence, however, can use an infinite number of levels and will result in an input sequence that will nearly always change levels at each sampling time. A desirable feature of the spectra for $T_{cl} = 2$ and $T_{cl} = 4$ is that both spectra are fairly flat over a large frequency range and then roll off very sharply. This feature enables the user to have more control over the frequency range of interest by changing values of T_{cl} rather than using different values for P_s. The spectra for the $T_{cl} = 2$ and $T_{cl} = 4$ input sequences weight the error relatively evenly over a larger frequency range than if sequences using a probability of switching were used. Hence, the models obtained using the four-level input sequence will be accurate over a larger frequency range.

Synthesis

The nonminimal state-space realization used by Hernández and Arkun (1993) was adopted in this work. For the SISO case, the states are defined as

$$x_1^a(k) = y(k) \qquad x_1^b(k) = u(k-1)$$
$$\vdots \qquad\qquad \vdots$$
$$x_{n_y+1}^a(k) = y(k-n_y) \qquad x_{n_u}^b(k) = u(k-n_u)$$

or

$$\boldsymbol{x}^a(k) = \left[x_1^a(k), \ldots, x_{n_y+1}^a(k)\right]^T$$
$$\boldsymbol{x}^b(k) = \left[x_1^b(k), \ldots, x_{n_u}^b(k)\right]^T,$$

where $\boldsymbol{x}^a(k)$ and $\boldsymbol{x}^b(k)$ correspond to the output and input parts of the state, respectively. The complete state is given by

$$\boldsymbol{x}(k) = \begin{bmatrix} \boldsymbol{x}^a(k) \\ \boldsymbol{x}^b(k) \end{bmatrix}.$$

The realization becomes:

$$\begin{bmatrix} \boldsymbol{x}^a(k+1) \\ \boldsymbol{x}^b(k+1) \end{bmatrix} = \begin{bmatrix} 0 & \cdots & 0 & 0 & & & & & \\ 1 & & & 0 & & & & & \\ & \ddots & & \vdots & & & & & \\ & & 1 & 0 & & & & & \\ & & & 0 & 0 & \cdots & 0 & 0 & \\ & & & & 1 & \ddots & & 0 & \\ & & & & & & & \vdots & \\ & & & & & & 1 & 0 & \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^a(k) \\ \boldsymbol{x}^b(k) \end{bmatrix} + \begin{bmatrix} f(\boldsymbol{x}(k), u(k)) \\ 0 \\ \vdots \\ 0 \\ u(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$y(k) = [1\ 0\ \cdots\ 0 \mid 0\ \cdots\ 0] \begin{bmatrix} \boldsymbol{x}^a(k) \\ \boldsymbol{x}^b(k) \end{bmatrix} \tag{2}$$

or

$$\boldsymbol{x}(k+1) = F(\boldsymbol{x}(k), u(k))$$
$$y(k) = h(\boldsymbol{x}(k)),$$

where $F(\boldsymbol{x}(k), u(k))$ is defined by Eq. 1 at time $k+1$. State estimation can be incorporated into the MPC scheme using Eq. 3:

$$[\hat{\boldsymbol{x}}(k+1 \mid k)] = [F(\hat{\boldsymbol{x}}(k \mid k), u(k))]$$
$$[\hat{\boldsymbol{x}}(k \mid k)] = [\hat{\boldsymbol{x}}(k \mid k-1) + L\hat{d}(k \mid k)]$$
$$\hat{y}(k \mid k) = h(\hat{\boldsymbol{x}}(k \mid k)), \tag{3}$$

where the notation $(k+1 \mid k)$ denotes the estimate at future sampling period $k+1$ based on information available at period k; L is the estimator gain; and $\hat{d}(k \mid k)$ is the difference between the plant measurement and model prediction, or

$$\hat{d}(k \mid k) = \tilde{y}(k) - \hat{y}(k \mid k-1). \tag{4}$$

The control action was computed by solving the nonlinear program in Eq. 5:

$$\min_{\boldsymbol{u}} \sum_{i=1}^{p} \gamma_i (r^*(k+i) - \hat{y}(k+i))^2 + \sum_{j=1}^{m} \lambda_j (\Delta u(k+j-1))^2$$

subject to

$$y(k) = y_0 + \sum_{j=1}^{n_y} \theta_j^1 y(k-j) + \sum_{j=1}^{n_u} \theta_j^2 u(k-j)$$
$$+ \sum_{j=1}^{n_u} \sum_{n=1}^{j} \theta_{j,n}^3 u(k-j)u(k-n)$$
$$+ \sum_{j=1}^{n_u} \sum_{n=1}^{j} \sum_{r=1}^{n} \theta_{j,n,r}^4 u(k-j)u(k-n)u(k-r)$$
$$\boldsymbol{u} = [u(k)u(k+1)\cdots u(k+m-1)]^T \tag{5}$$
$$\boldsymbol{u}_{\text{low}} \le \boldsymbol{u} \le \boldsymbol{u}_{\text{high}}$$
$$\Delta \boldsymbol{u}_{\text{low}} \le \Delta \boldsymbol{u} \le \Delta \boldsymbol{u}_{\text{high}},$$

where p and m are the standard tuning parameters used in MPC corresponding to the prediction and input move horizons, respectively. The reference trajectory, $r^*(k+i)$ in Eq. 5, is computed by subtracting the filtered value of the disturbance from the filtered value of the setpoint. Hernández and Arkun (1993) and Sriniwas et al. (1995) filtered the setpoint, but used no filter in the feedback path. Placing a filter in the feedback path yields improved tuning (García and Morari, 1986; Prett and García, 1988; Ricker, 1990) and was very advantageous for the multivariable case study in this article. This approach effectively treats the disturbance as a step passed through a first-order system. Wellons and Edgar (1987) originally proposed this disturbance model, and Ricker (1991) demonstrated that it can yield improved performance over using step or ramp disturbance models.

The two filters in Figure 2 were implemented in the same manner as the reference model in the work by Ricker (1990). For $F_1(z)$, a model was obtained for

$$x_{r1}(k+1) = \Phi_{r1} x_{r1}(k) + \Gamma_{r1} \hat{d}(k \mid k)$$
$$y_{r1}(k) = x_{r1}(k). \tag{6}$$

The model for $F_2(z)$ was given by

$$x_{r2}(k+1) = \Phi_{r2} x_{r2}(k) + \Gamma_{r2} r(k)$$
$$y_{r2}(k) = x_{r2}(k) \tag{7}$$

and

$$r^*(k+i) = y_{r2}(k+i) - y_{r1}(k+i), \qquad i = 1, \ldots, p. \tag{8}$$

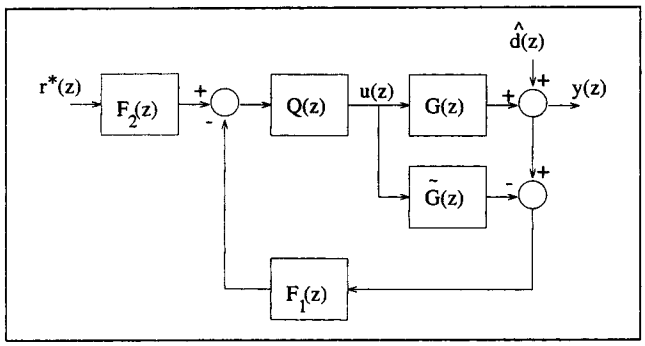

Figure 2. Internal model control block diagram with two-filter structure.

Hence, the control problem at the beginning of sampling period k is as follows: compute the reference trajectory $r^*(k+i)$ over the prediction horizon, p. Calculate the manipulated variable by solving the optimization problem in Eq. 5. Update the state by Eq. 3. Obtain the plant measurement, $\tilde{y}(k+1)$. Compute $\hat{d}(k+1|k+1)$. Set $k=k+1$ and repeat the sequence.

A second advantage of incorporating an autoregressive plus Volterra model into a nonlinear MPC scheme is that it leads to a sixth-order programming problem regardless of the value of the prediction horizon, p. While this nonlinear program is more challenging to solve than the second-order (quadratic) program encountered in linear MPC, it is easier to solve than the nonlinear programs encountered in some other nonlinear MPC schemes. A widely used algorithm for unconstrained optimization is the Broyden–Fletcher–Goldfarb–Shanno method (Dennis and Schnabel, 1983). MINOS 5.1 (Murtagh and Saunders, 1987) employs a sequential quadratic programming approach to solving nonlinear constrained problems. Both of these solvers approximate the objective function using a quadratic form. Hence, if two objective functions are considered in which one objective function can be more accurately approximated by a quadratic form, the corresponding optimization problem could be expected to be solved more efficiently and reliably. In addition, one group of researchers (Chow et al., 1994) has developed an algorithm for unconstrained optimization using tensor methods that employs a fourth-order model of the objective function. In their case studies, the tensor method required significantly fewer iterations and function evaluations to solve most unconstrained optimization problems than standard methods based on quadratic models.

Using a first-principles model directly in a nonlinear MPC scheme (Economou et al., 1986b; Li and Biegler, 1988, 1989, 1990; Eaton and Rawlings, 1990) yields a general nonlinear program that may contain exponential nonlinearities due to Arrhenius temperature dependence or division of one state by another as would be the case for computation of the molecular weight in a polymerization reactor. Since the quadratic approximation used by current nonlinear optimization algorithms may be a poor approximation for the objective function when a first-principles model is employed, the efficiency of the nonlinear solver is reduced. In addition, using a first-principles model has the disadvantage that the computational load scales with the number of state equations, which may prevent its use for control of a distillation column, for example.

A general polynomial ARMA model (Hernández, 1992; Hernández and Arkun, 1993; Sriniwas et al., 1995) has the advantage that the model nonlinearities are polynomial nonlinearities, as opposed to exponential nonlinearities, which do not appear in the form of a quadratic approximation. However, for the general case of nonlinearities in the y regressors, such as $y^2(k-1)y(k-2)$ and $y(k-1)u(k-2)$, the objective function becomes a higher order function of the decision variables [$u_1(k)$, $u_2(k)$, ..., for example] as p is increased. For a modest value of $p=5$, the resulting objective function could be 20th-order in the decision variables, for example. A higher order objective function could be expected to be more poorly approximated by a quadratic approximation, leading to a less efficient and possibly unreliable solution using current nonlinear optimization techniques.

Since the autoregressive plus Volterra model does not contain nonlinearities in the y regressors, its objective function never exceeds sixth-order, regardless of the value of p. Hence the nonlinear program associated with autoregressive plus Volterra models is less computationally burdensome than those encountered when first-principles models and polynomial ARMA models are used.

Stability analysis

A third advantage of the proposed model structure is that global stability conditions are available for the model. The autoregressive plus Volterra model for y_i in Eq. 1 is bounded-input–bounded-output (BIBO) stable if the model contains no y_j terms and the roots of Eq. 9 lie inside the unit circle (Narendra and Parthasarathy, 1990):

$$z^{n_y} - \theta^{1,i}_{l,0} z^{n_y-1} - \theta^{1,i}_{l,1} z^{n_y-2} - \cdots - \theta^{1,i}_{l,n_y-1} = 0. \qquad (9)$$

It follows from Eq. 9 that an impulse-response model and a Volterra model are BIBO stable. This is expected, since these models may only be used to describe stable systems, because an unstable system does not have a bounded impulse response. There are no global stability conditions for a general polynomial ARMA model. Global stability conditions do exist for bilinear models (Priestley, 1988; Subba-Rao, 1979). However, the stability conditions for bilinear models are conservative in that they contain models in addition to the one being analyzed. Equation 9, however, is not a conservative condition.

If the condition in Eq. 9 is satisfied, the autoregressive plus Volterra model is BIBO stable and nominal stability analysis of the closed-loop system is simplified considerably. Transfer-function techniques such as IMC can be used in the analysis of MPC in the case of linear, unconstrained systems (Prett and García, 1988). Figure 2 shows the IMC structure where filters may be used for setpoint and disturbance filtering, $F_2(z)$ and $F_1(z)$, respectively. The plant is denoted by $G(z)$; $\tilde{G}(z)$ corresponds to the model of the process; and $Q(z)$ is the controller designed using $\tilde{G}(z)$. Consider the nominal $[G(z)=\tilde{G}(z)]$ case where $G(z)$ is an autoregressive plus Volterra model. For the nominal case, there is no feedback in Figure 2, and $Q(z)$ becomes a feedforward controller. Closed-loop stability is guaranteed if each of the three blocks

in series [$F_2(z)$, $Q(z)$, and $G(z)$] is stable. The setpoint filter is specified by the designer and contains stable transfer functions on its diagonal. Hence, $F_2(z)$ is stable. The BIBO stability condition in Eq. 9 does not guarantee that the plant will be stable for an unbounded input signal. However, the input to the plant is the output of a control valve, which is always physically bounded. Hence, closed-loop BIBO stability is guaranteed from the setpoint to the output, provided the roots of Eq. 9 lie inside the unit circle.

Economou and Morari (1986a) employed the small-gain theorem (Zames, 1966) and the IMC structure to obtain a closed-loop stability condition for control of a nonlinear system using the inverse of the nonlinear process as the controller. The stability condition was also applied to the case in which a robustness filter was placed in the feedback path to detune the controller for plant–model mismatch.

The tools of robust control may be used to analyze nonlinear models and controllers over some region by ensuring that the nonlinear system behavior is contained inside the uncertainty description (Doyle, 1982). Doyle III et al. (1989) bounded nonlinearities with conic sectors. Hernández (1992) applied the conic sector approach in Doyle III et al. (1989) to analyze polynomial ARMA models and their inverses. The nonlinear model was approximated as a linear plus uncertain system. One of the contributions of Hernández (1992) was that the stability of the inverse of a linear plus uncertain system having an $M-\Delta$ structure could be determined without computing its inverse. Avoiding the construction of the analytical expression of the inverse is advantageous because it is impractical in general. If M_{22} and $(I-M_{11}\Delta)$ are invertible, the system may be transformed into an $M^I-\Delta$ configuration, representing the inverse of the system using the following relationships (Hernández, 1992):

$$\begin{aligned}
\Delta^I &= \Delta \\
M_{11}^I &= M_{11} - M_{12}M_{22}^{-1}M_{21} \\
M_{12}^I &= M_{12}M_{22}^{-1} \\
M_{21}^I &= -M_{22}^{-1}M_{21} \\
M_{22}^I &= M_{22}^{-1}.
\end{aligned} \tag{10}$$

Equation 11 bounds a nonlinear function $f(\boldsymbol{x}(k), u(k))$ using a single cone with center $[A\ b]$ and radius $[R_x\ r_b]$.

$$f(x,u) \in [A\ b]\begin{bmatrix} x \\ u \end{bmatrix} + \Delta(x,u)[R_x\ r_b]\begin{bmatrix} x \\ u \end{bmatrix}, \tag{11}$$

where $\Delta(x,u) \in [-1,1]$ represents a scalar uncertainty. The first and second terms on the righthand side of Eq. 11 are linear and uncertain parts of the approximation, respectively.

If the system is bounded in a region defined by conic sectors and it is assumed that the system does not leave this region, its inverse will be stable if $(I-M_{11}\Delta)$ and M_{22} are invertible, and

$$\min_D \bar{\sigma}(DM_{11}^I D^{-1}) < 1. \tag{12}$$

The optimization problem in Eq. 12 is a convex optimization problem that is solved for a constant scaling matrix, D, to provide a less conservative upper bound for μ, the structured singular value (Doyle, 1982). The condition in Eq. 12 guarantees that the controller is exponentially stable. Hence, it is a more stringent stability guarantee than BIBO stability.

Hernández (1992) used a conservative uncertainty formulation in which uncertainties were calculated as

$$\begin{aligned}
\frac{\partial x_1^a(k+1)}{\partial x_i^a(k)} &= \frac{\partial f(x,u)}{\partial x_i^a} = \alpha_i^o + \delta_i \alpha_i^R \\
\frac{\partial x_1^a(k+1)}{\partial x_j^b(k)} &= \frac{\partial f(x,u)}{\partial x_j^b} = \beta_j^o + \delta_{n_y+1+j}\beta_j^R
\end{aligned} \tag{13}$$

where $\delta_i \in [-1,1]$, for $i=1, \ldots, N$ and

$$\frac{\partial f(x,u)}{\partial u} = \gamma_u^o + \delta_u \gamma_u^R, \tag{14}$$

where $\delta_u \in [-1,1]$. The states (x_1^a and x_1^b), input (u), and function (f) are defined in the state-space realization in Eq. 2. The parameters α_i^o, β_j^o, and γ_u^o pertain to values in the matrices of a linear state-space model that would be used to describe a nonlinear polynomial ARMA model if there was no uncertainty ($\delta=0$). Hence, the uncertainty formulation reduces to a linear approximation of the nonlinear polynomial ARMA model. The parameters, δ_i, δ_{n_y+1+j}, and δ_u correspond to the uncertainties that are used to bound the nonlinearity of the polynomial ARMA model over a specified region.

While the stability of the inverse controller yields insight into the stability of the closed-loop system, the inverse controller is rarely used in practice because it usually requires very aggressive manipulated variable action. Aggressive manipulated variable profiles are undesirable because they cause excessive wear on control valves and may be an inefficient use of utilities such as steam and cooling water. In addition, the inverse controller can perform poorly when plant–model mismatch exists. Also, the inverse controller will be unstable for a model that contains a zero that lies outside the unit circle. These problems can be avoided if a relaxed inverse controller is used where $p>m$ in the MPC notation. However, there are no general stability results for the controller, $Q(z)$, in Figure 2 if $Q(z)$ is the general nonlinear model-predictive controller of Eq. 5. Hernández (1992) proposed a method for studying the stability of the p-inverse controller. This controller is a particular case of the model-predictive control algorithm, where $m=1$, $\lambda=0$, $\gamma_i=0$ for $i=1, \ldots, p-1$, and $\gamma_p=1$. The analysis of the p-inverse controller consists of describing the nonlinear model as a linear plus time-varying uncertain model relating $y(k+p)$ to $u(k)$, where $p>1$. The stability of the p-inverse controller is guaranteed, provided that the inverse of the model relating $y(k+p)$ to $u(k)$ is stable. The number of uncertainty parameters increases as p increases. Hence if the uncertainty description is conservative, the increase in the number of uncertainty parameters results in additional conservatism, limiting its usefulness.

BIBO closed-loop stability is guaranteed from the setpoint to the output for the nominal case, provided the roots of Eq.

9 lie inside the unit circle and the condition in Eq. 12 is satisfied for the p-inverse controller. In addition if a μ analysis is also performed on the model and $\mu < 1$, each of the three operators in series ($F_2(z)$, $Q(z)$, and $G(z)$) is exponentially stable, and exponential closed-loop stability is guaranteed for the nominal case from the setpoint to the output. In this work, a significantly less conservative uncertainty description is used to guarantee stability over a small operating region for unconstrained p-inverse controllers based on an autoregressive plus Volterra model identified from input–output data.

SISO Control of an Isothermal Polymerization Reactor

As an introduction to a multivariable case study, the identification, control, and stability analysis of a single-input–single-output (SISO) process is studied. Consider the isothermal free-radical polymerization of methyl methacrylate using azobisisobutyronitrile (AIBN) as initiator and toluene as solvent (Baillagou and Soong, 1985a,b; Congalidis et al., 1989; Daoutidis et al., 1990; Doyle III et al., 1995). Soroush and Kravaris (1992) investigated the control of the reactor temperature for this polymerization in an experimental batch reactor using global linearizing control with coordination rules for the two actual manipulated variables. In the present study, the number-average molecular weight is controlled by manipulating the inlet initiator flow rate. The process is shown in Figure 3. The isothermal model was obtained by setting the reactor temperature at its steady-state value of 335 K. Under these assumptions, the six-state model in Daoutidis et al. (1990) reduces to the following four-state model:

$$\frac{dC_m}{dt} = -(k_p + k_{fm})C_m P_0 + \frac{F(C_{m_{in}} - C_m)}{V} \tag{15}$$

$$\frac{dC_I}{dt} = -k_I C_I + \frac{F_I C_{I_{in}} - FC_I}{V} \tag{16}$$

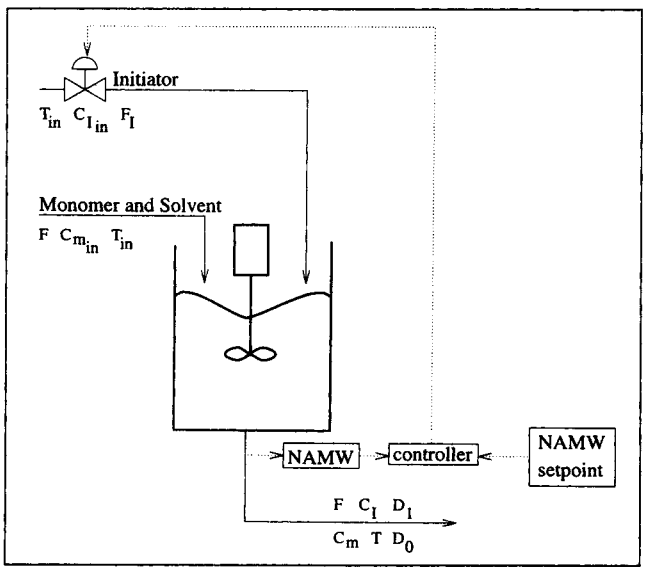

Figure 3. Control configuration for SISO case study.

Table 1. Parameters for the SISO Case Study

$k_{T_c} = 1.3281 \times 10^{10}$	$m^3/(kmol \cdot h)$
$k_{T_d} = 1.0930 \times 10^{11}$	$m^3/(kmol \cdot h)$
$k_I = 1.0225 \times 10^{-1}$	1/h
$k_p = 2.4952 \times 10^6$	$m^3/(kmol \cdot h)$
$k_{f_m} = 2.4522 \times 10^3$	$m^3/(kmol \cdot h)$
$f^* = 0.58$	
$F = 1.00$	m^3/h
$V = 0.1$	m^3
$C_{I_{in}} = 8.0$	$kmol/m^3$
$M_m = 100.12$	kg/kmol
$C_{m_{in}} = 6.0$	$kmol/m^3$

$$\frac{dD_0}{dt} = (0.5k_{T_c} + k_{T_d})P_0^2 + k_{f_m}C_m P_0 - \frac{FD_0}{V} \tag{17}$$

$$\frac{dD_1}{dt} = M_m(k_p + k_{f_m})C_m P_0 - \frac{FD_1}{V} \tag{18}$$

$$y = \frac{D_1}{D_0}, \tag{19}$$

where

$$P_0 = \left[\frac{2f^* k_I C_I}{k_{T_d} + k_{T_c}}\right]^{0.5}.$$

The model parameters and steady-state operating conditions are listed in Tables 1 and 2, respectively.

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 $u = [0.0046, 0.028966]$ m^3/h. The switching probability, P_s, was selected by trial and error. The inputs and outputs were placed in scaled, deviation form, that is, $\tilde{u} = (u - u_0)/u_0$ and $\tilde{y} = (y - y_0)/y_0$. The regressors and model parameters were obtained using a stepwise model building algorithm (Kortmann et al., 1988). This approach employed statistical tests such as Akaike information criteria (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 by using the same switching probability with values drawn from $u = [0.0151047, 0.0184613]$ m^3/h, which corresponded to $\pm 10\%$ of the nominal value of the input. The smaller range was used for identifying the linear model, because using the larger range yielded a poor lin-

Table 2. Steady-state Operating Conditions for the SISO Case Study

$C_m =$	5.506774	$kmol/m^3$
$C_I =$	0.132906	$kmol/m^3$
$D_0 =$	0.0019752	$kmol/m^3$
$D_1 =$	49.38182	kg/m^3
$u_0 =$	0.016783	m^3/h
$y_0 =$	25,000.5	kg/kmol

Table 3. Models for the SISO Case Study

Linear Model		Nonlinear Model	
Coefficients	Regressors	Coefficients	Regressors
0.0000	1	$\theta_0 =$ 0.0000	1
0.9504	$y(k-1)$	$\theta_1 =$ 1.3072	$y(k-1)$
−0.0120	$u(k-1)$	$\theta_2 =$ −0.0141	$u(k-1)$
−0.0105	$u(k-3)$	$\theta_3 =$ 0.0186	$u(k-5)u(k-1)$
−0.0204	$u(k-2)$	$\theta_4 =$ 0.0359	$y(k-5)$
−0.0694	$y(k-5)$	$\theta_5 =$ −0.0040	$u^2(k-5)$
−0.0039	$u(k-4)$	$\theta_6 =$ −0.0203	$u(k-2)$
		$\theta_7 =$ 0.0200	$u(k-4)u(k-2)$
		$\theta_8 =$ 0.0059	$u(k-5)$
		$\theta_9 =$ −0.0169	$u(k-5)u(k-4)u(k-1)$
		$\theta_{10} =$ 0.0051	$u(k-3)$
		$\theta_{11} =$ −0.0164	$u^2(k-4)$
		$\theta_{12} =$ −0.0183	$u(k-4)u^2(k-2)$
		$\theta_{13} =$ 0.0090	$u(k-3)u(k-2)$
		$\theta_{14} =$ −0.0036	$u(k-5)u(k-3)$
		$\theta_{15} =$ 0.0189	$u^3(k-4)$
		$\theta_{16} =$ −0.3860	$y(k-3)$
		$\theta_{17} =$ 0.0060	$u(k-4)$
		$\theta_{18} =$ −0.0097	$u(k-5)u(k-2)$
		$\theta_{19} =$ 0.0062	$u^2(k-2)u(k-1)$

ear model due to the change in the magnitude of the gain in this region. The inputs and outputs were also put in scaled, deviation form. The regressors and model parameters were obtained using the stepwise model-building algorithm with $n = 1$, $n_u = 5$, and $n_y = 5$. The models obtained are shown in Table 3.

In Figure 4, the changes in the number-average molecular weight in response to initiator flow rates of $F_I = 0.016783 \pm 0.012183$ m^3/h are shown. These step inputs have predominantly low-frequency characteristics and demonstrate the long-term prediction accuracy of the identified models. The range of the y-axis in Figure 4 is $\pm 15{,}500$ from the nominal value of 25,000.5 kg/kmol to highlight the asymmetric behavior of this system. The true process output is represented by the solid lines, the linear model output is given by the dashed

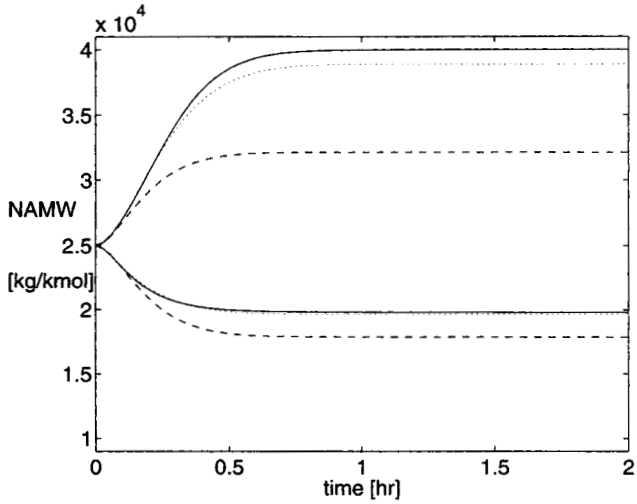

Figure 4. Open-loop simulations for step changes of ± 0.012183 m^3/h in F_I from its nominal value of 0.016783 m^3/h.

Nonlinear: solid; linear: dashed; autoregressive plus Volterra: dotted.

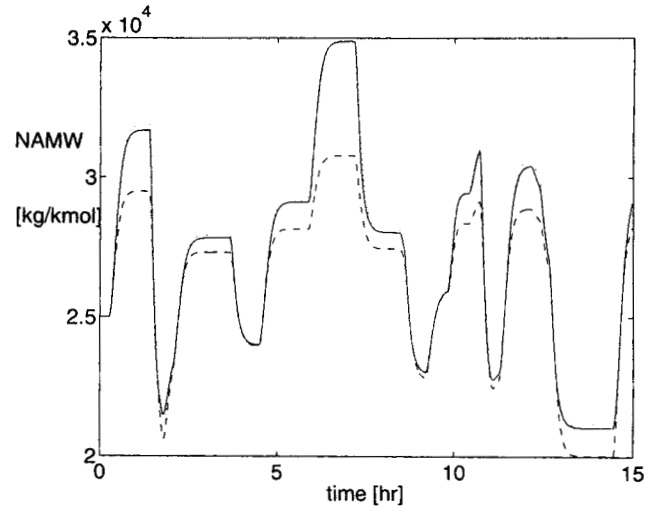

Figure 5. Validation sequence for random steps in initiator flow rate.

Nonlinear: solid; linear: dashed; autoregressive plus Volterra: dotted.

lines, and the dotted lines correspond to the output of the autoregressive plus Volterra model. 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 step changes are modeled well. The results shown in Figure 5 demonstrate the improved modeling performance of the autoregressive plus Volterra model for a validation input sequence with $P_s = 0.05$.

Setpoint tracking

For the closed-loop simulations, it was assumed that there was a performance specification of $\pm 1{,}000$ kg/kmol around the setpoint for which the product is considered acceptable. Hence for a setpoint change simulating a grade change, it is desirable to quickly enter and remain within the new product specification bounds. Nonlinear MPC, linear MPC, and a proportional-integral (PI) controller were tuned to yield comparable responses for a setpoint change from 25,000.5 kg/kmol to 38,000 kg/kmol. The PI controller tuning parameters were obtained using IMC tuning rules (Morari and Zafiriou, 1989) and were $K_c = -1.371$ and $\tau_I = 0.225$ h. The corresponding IMC filter time constant was 0.40 h. The tuning parameters for linear MPC were $m = 1$, $p = 20$, $\gamma = 1$, $\lambda = 0$, $\Phi_{r2} = 0.94$, $\Phi_{r1} = 0.94$, and L was an 11×1 matrix of zeros. The tuning parameters for nonlinear MPC were $m = 1$, $p = 10$, $\gamma = 1$, $\lambda = 0$, $\Phi_{r2} = 0.8$, $\Phi_{r1} = 0.7$, and L was an 11×1 matrix of zeros. There were 11 states in the nonminimal state-space realization in Eq. 2 for the input–output models used in this case study. The linear and nonlinear MPC schemes employed different parameters because the underlying linear models used by the two controllers were 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 were

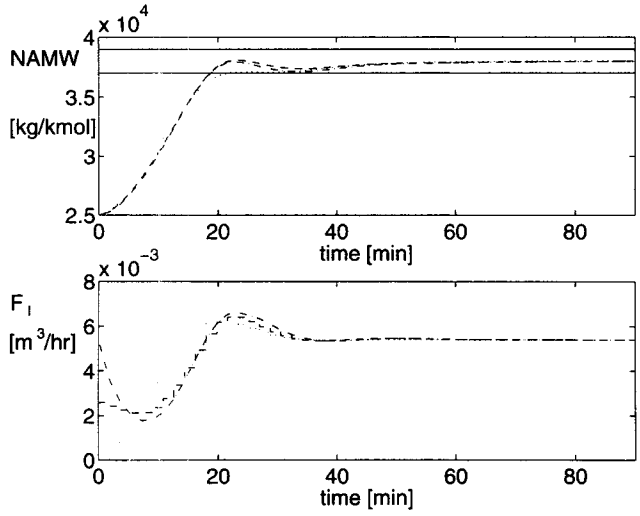

Figure 6. Closed-loop simulation for a filtered-step setpoint change from 25,000.5 kg/kmol to 38,000 kg/kmol in number-average molecular weight.

PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted.

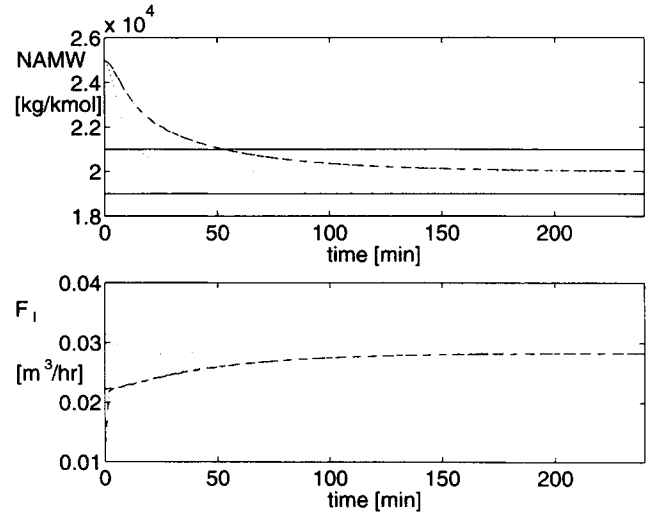

Figure 7. Closed-loop simulation for a filtered-step setpoint change from 25,000.5 kg/kmol to 20,000 kg/kmol in number-average molecular weight.

PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted.

tuned so that the manipulated variable did not reach its lower constraint. This was done so that improved performance could be solely attributed to incorporating a more accurate model into the control scheme and not be influenced by the ability of MPC to take into account constraints.

The closed-loop responses obtained by these three controllers are nearly indistinguishable for this setpoint change, as shown in Figure 6. The number-average molecular weight 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 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 kg/kmol are shown in Figure 7. Here the performance of linear MPC and a PI controller are again nearly the same. The failure of linear MPC to outperform PI control for both setpoint changes is a result of this problem being an unconstrained, SISO problem. The strengths of MPC that enable it to outperform PI control are its ability to account for constraints and interactions in multivariable problems. 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 70%. In addition, the nonlinear controller reaches the new setpoint after 1.32 h, while the other two controllers take 7.05 h, an improvement of 81.3%. The performance improvement is due to the nonlinear model being more accurate, enabling the nonlinear model-based controller to take more appropriate control action.

It may be argued that the linear MPC and PI control schemes could be tuned more aggressively so that the 20,000-kg/kmol setpoint would be reached more quickly. However, this approach would yield improved performance for the 20,000-kg/kmol setpoint change at the expense of poorer performance for the 38,000-kg/kmol setpoint change. Linear MPC and the PI controller are already tuned quite aggressively, as evidenced by the underdamped responses for the 38,000-kg/kmol setpoint change. Tuning these two controllers more aggressively would yield additional oscillations to those shown in Figure 6 and lengthen the time it takes for these two controllers to keep the number-average molecular weight within its performance specification bounds. Hence, a compromise must be made in tuning the linear model-based controllers for servo control of the polymerization reactor.

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 reenter 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 2 allows the designer to tune optimally and independently for both setpoint tracking and disturbance rejection. However in the case of a nonlinear plant, the $F_1(z)$ filter in Figure 2 acts to filter both unmeasured disturbances and plant–model mismatch, both of which are nonlinear. The nonlinearity may be so severe that $F_1(z)$ 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 capable of treating setpoints and disturbances differently. Controlling the nonlinear reactor with linear MPC using the two-degrees-of-freedom controller with $F_2(z) \neq F_1(z)$ required a compromise to be made. Disturbance rejection could be improved by tuning $F_1(z)$ very aggressively. However, the resulting filter parameter for $F_1(z)$ resulted in a significantly worse performance for the 38,000-kg/kmol setpoint change. The poorer performance was due to the fact that the

optimal $F_1(z)$ filter parameter for disturbance rejection was not the optimal $F_1(z)$ filter parameter for setpoint tracking, owing to the nonlinear behavior of the polymerization reactor. For this case study, the filter parameter for $F_1(z)$ could not be reduced below that of $F_2(z)$ without observing performance deterioration for the 38,000-kg/kmol setpoint change. Hence, linear MPC had to be detuned for robustness to such a degree that it was not able to treat setpoints and disturbances differently. This observation is in agreement with the statement by Prett and Gracía (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 $F_2(z) = F_1(z)$. When combined with state estimation, this structure 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-kg/kmol setpoint change.

As evidenced by the results in Figures 4 and 5, the autoregressive plus Volterra model is a very accurate model of the true nonlinear process in the operating region of the plant. Hence, the plant–model mismatch is quite small when the autoregressive plus Volterra model is used in a control scheme, and the value being filtered by $F_1(z)$ (Figure 2) is due primarily to unmeasured disturbances and not plant–model mismatch. While the nonlinearity of the process prevents complete decoupling of the tuning problem for setpoint 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 ($\Phi_{r2} = 0.8$, $\Phi_{r1} = 0.7$). $\pm 1{,}000$ kg/kmol, 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 $[G(z) = \tilde{G}(z)]$ case where the autoregressive plus Volterra model in Table 3 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. Equation 9 may be used to compute the five poles of the plant that are $z = 0.8254$, 0.8254, 0.3746, 0.3746, and 0.3755. All poles lie inside the unit circle, making $G(z)$ BIBO stable. In addition, a μ-analysis for the model resulted in $\mu < 1$ for the region over which the model is valid ($\tilde{u} \in [-0.7259, 0.7259]$). Hence, the model is exponentially stable for its region of validity.

The zeros of the linear part of the autoregressive plus Volterra model are $z = 1.6015$, 0.7554, 0.5881, and 0.5881. Although the linear part of the model has a zero that lies outside the unit circle, it does not exhibit inverse response. In addition, the output of the autoregressive plus Volterra model in Figures 4 and 5 does not display nonminimum phase behavior. It can be proved for linear systems that a discrete system may have a zero outside the unit circle and not exhibit inverse response (Åström and Wittenmark, 1990). This result holds locally for nonlinear discrete systems. The zero outside the unit circle results in unstable zero dynamics and renders the inverse controller unstable. This result was verified with an unstable nominal closed-loop simulation for a setpoint change from from 25,000.5 kg/kmol to 28,000 kg/kmol using the inverse of the autoregressive plus Volterra model as the controller (not shown). The stability of the p-inverse controller was considered with $p > 1$. Employing the uncertainty description of Hernández (1992) led to an uncertain system with five uncertainities. This approach uses "lumped" uncertainty and is a conservative uncertainty description. The R_x matrix containing uncertainties for the states and r_b vector containing uncertainities for the input over this $\tilde{u}$ range were computed using Eqs. 13 and 14 and are given by

$$R_x = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 2.0575e-03 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 6.3000e-04 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.7760e-03 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.0372e-03 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (20)$$

$$r_b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 9.8000e-04 \end{bmatrix}. \quad (21)$$

In Figure 8, the responses of the three controllers for a change in monomer feed concentration from $C_{m_{in}} = 6$ kmol/m^3 to $C_{m_{in}} = 5$ kmol/m^3 are shown. A decrease in monomer concentration reduces the degree of polymerization, lowering the molecular weight (Hill, 1977). Linear MPC and a PI controller return the molecular weight to its specification range after 37.8 min, while nonlinear MPC achieves this goal after only 25.2 min, a 33.3% improvement. The times needed to return the molecular 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

Additional details of the calculation are given in Maner (1996). The results of the stability analysis of the p-inverse controller are listed in Table 4. The optimization problem in Eq. 12 was solved using mu.m in the Matlab μ-Analysis and Synthesis Toolbox (Balas et al., 1995). The upper bounds for

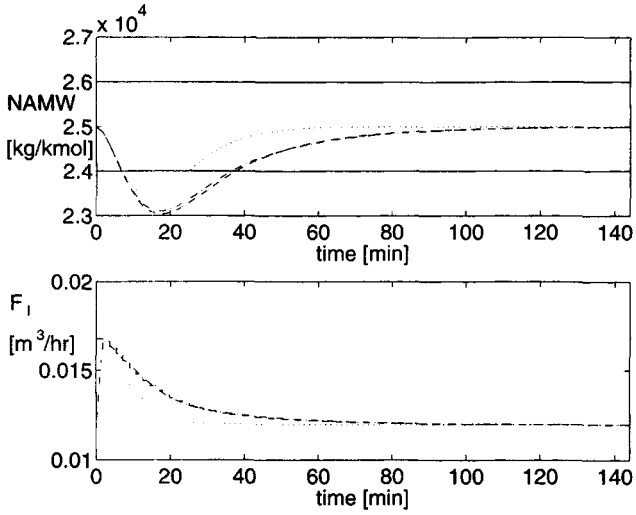

Figure 8. Closed-loop simulation for an unmeasured step disturbance in monomer feed composition from its nominal value of $C_{min}=6$ kmol/m^3 to $C_{min}=5$ kmol/m^3.

PI control: dash-dot; linear MPC: dashed; nonlinear MPC: dotted.

μ, which are less than one, indicate that stability of the p-inverse controller is guaranteed for the $\tilde{u}\in[-0.05,0.05]$ range for p values of 2, 3, 5, and 10. The upper bound for the structured singular value in Table 4 first decreases and then increases as p increases. The reason for this is that as p is increased to values of 2, 3, and 5, the p-inverse controller is being detuned. Although a larger p value (e.g., 10) usually results in a less aggressive controller, the upper bound increases due to the significant increase in the number of uncertainties and the corresponding increase in conservatism.

Table 5 lists the results of the analysis of the p-inverse using the conservative uncertainty description over the larger range of $\tilde{u}\in[-0.1,0.1]$. Stability of the p-inverse cannot be guaranteed for any p values due to the conservatism of the uncertainty description employed.

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

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

Table 4. Stability Analysis of the p-Inverse Model for $\tilde{u}\in[-0.05, 0.05]$ Using the Conservative Uncertainty Description

p	$\min_D \bar{\sigma}(DM_{11}^I D^{-1}$	No. of Uncertainties
1	1.8254	5
2	0.9616	10
3	0.9434	15
5	0.9368	25
10	0.9527	50

Table 5. Stability Analysis of the p-Inverse Model for $\tilde{u}\in[-0.1, 0.1]$ Using the Conservative Uncertainty Description

p	$\min_D \bar{\sigma}(DM_{11}^I D^{-1})$	No. of Uncertainties
1	2.0839	5
2	1.1265	10
3	1.1519	15
5	1.1613	25
10	1.1869	50

and noting that the following regressors lead to six independent uncertainties (δ's):

$$\tilde{u}^2(k-3)=x_8^2(k)$$
$$\tilde{u}(k-3)=x_8(k)$$
$$\tilde{u}(k-4)=x_9(k)$$
$$\tilde{u}^2(k-1)=x_6^2(k)$$
$$\tilde{u}(k-4)\tilde{u}(k-3)=x_9(k)x_8(k)$$
$$\tilde{u}(k-2)=x_7(k).$$

The autoregressive plus Volterra model in Table 3 may be recast as a linear plus uncertain system as

$$x(k+1)=[A\ b]\begin{bmatrix}x(k)\\ \tilde{u}(k)\end{bmatrix}=\delta_1[r_{B1}x_8]$$
$$+\delta_2[r_{B2}x_8+r_{B4}x_6]+\delta_3[r_{B3}\tilde{u}+r_{B5}x_9+r_{B6}x_6]$$
$$+\delta_4[r_{B7}\tilde{u}+r_{B8}x_8]+\delta_5[r_{B9}\tilde{u}]+\delta_6[r_{B10}x_6+r_{B11}x_9], \tag{22}$$

where $\delta_i\in[-1,1]$ and $r_{B1}, r_{B2}, \ldots, r_{B11}$ denote the bounds on selected terms. Table 6 contains the variables for the r_B values and numerical values for the $\tilde{u}\in[-0.14,0.14]$ range. As an example of how the r_B values are computed, consider $[r_{B7}\tilde{u}+r_{B8}x_8]$.

$$r_{B7}\tilde{u}+r_{B8}x_8=\theta_{19}\tilde{u}^2(k-1)\tilde{u}(k)+\theta_{12}\tilde{u}^2(k-1)\tilde{u}(k-3)$$
$$=(\theta_{19}x_6^2)\tilde{u}+(\theta_{12}x_6^2)x_8$$
$$=(0.0062)(0.14)^2\tilde{u}+(-0.0183)(0.14)^2x_8$$
$$=1.22\times10^{-4}\tilde{u}+(-3.59\times10^{-4})x_8$$

Table 6. Bounds Using the Structured 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}

where the numerical values for r_{B7} and r_{B8} appear in Table 6. The R_x matrix and r_b vector of uncertainties obtained for the $\tilde{u} \in [-0.14, 0.14]$ range using the structured uncertainty description are given by

$$R_x = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.8900e-04 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.0000e-03 & 0 & 1.6400e-03 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9.7000e-04 & 0 & 0 & 4.0000e-04 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.8300e-04 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9.0000e-04 & 0 & 0 & 3.6000e-04 \end{bmatrix} \quad (23)$$

$$r_b = \begin{bmatrix} 0 \\ 0 \\ 1.8600e-03 \\ 6.2000e-05 \\ 1.6900e-04 \\ 0 \end{bmatrix}. \quad (24)$$

Additional details of the calculation are given in (Maner, 1996). Although the number of uncertain parameters increased, the uncertainty was focused in a more structured manner rather than using a "lumped" uncertainty description. The sizes of the uncertain parameters decreased significantly, and the smaller uncertainty values more than offset the effect of adding one additional uncertainty parameter. The results of the stability analysis for the p-inverse controller are given in Table 7 for $\tilde{u} \in [-0.14, 0.14]$. Stability is guaranteed for $p = 2$, 3, 5, and 10. Hence, the region over which stability is guaranteed has been increased by over 40%. The same trend as before is observed in the value of the upper bound for the structured singular value with increasing p values.

The results in Figure 9 show a nominal closed-loop simulation for a setpoint change from 25,000.5 kg/kmol to 26,300 kg/kmol using the autoregressive plus Volterra model and a p-inverse controller with $p = 10$ illustrating that the closed-loop system is indeed stable for $\tilde{u} \in [-0.14, 0.14]$. The lower limit of $\tilde{u} = -0.14$ corresponds to $F_I = 0.01443$ m^3/h. The other tuning parameters were $\Phi_{r2} = 0.8$, $\Phi_{r1} = 0.7$, $\gamma = 1$, $\lambda = 0$, and $m = 1$. The tuning parameter for Φ_{r1} was irrelevant for this servo simulation because there is no feedback for the nominal case. In this example exponential closed-loop stability is guaranteed for the nominal case for $\tilde{u} \in [-0.14, 0.14]$, since each of the three operators in series $[F_2(z), Q(z)$, and $G(z)]$ is exponentially stable for this range of operation. If the μ test had failed for the controller but the roots of Eq. 9 were inside the unit circle, BIBO stability from setpoint to output would still be guaranteed, since a control valve would clip any large output from the controller to yield a bounded input to the BIBO stable model.

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 6, 7, and 8, show that the closed-loop system using a more aggressively tuned general nonlinear MPC scheme based on the autoregressive plus 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 autoregressive plus Volterra model in a nonlinear MPC scheme, this case study was a SISO problem. Most problems encountered in the chemical industry are multivariable. However, the proposed model structure can be implemented in multivariable control problems as well.

Table 7. Stability Analysis of the p-Inverse Model for $\tilde{u} \in [-0.14, 0.14]$ Using the Structured Uncertainty Description

p	$\min_D \bar{\sigma}(DM_{11}^I D^{-1})$	No. of Uncertainties
1	1.9220	6
2	0.9843	12
3	0.9861	18
5	0.9815	30
10	0.9837	60

MIMO Control of a Copolymerization Reactor

Process model

Figure 10 is a flow sheet 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 as-

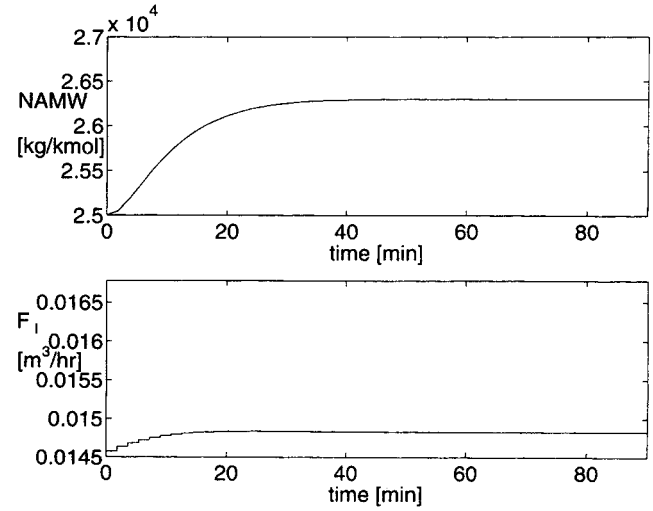

Figure 9. Nominal closed-loop simulation for a filtered setpoint step change from 25,000.5 to 26,300 kg/kmol using p-inverse control with $p = 10$.

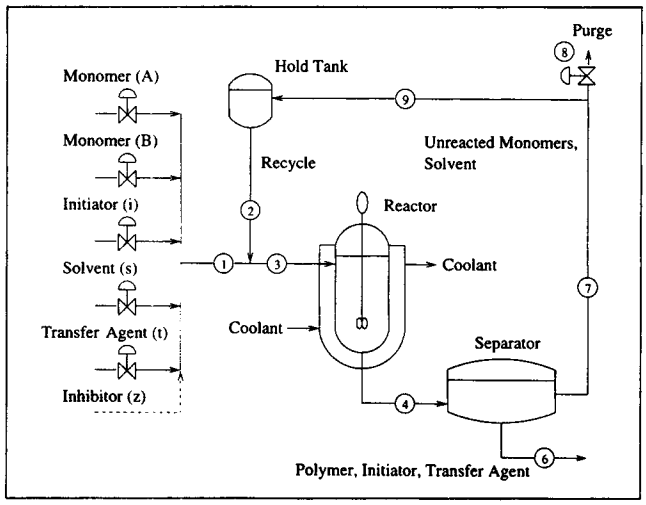

Figure 10. Flow diagram of copolymerization reactor with recycle.

sumed 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.

A few typographical errors appeared in the model in the original article (Congalidis et al., 1989). Specifically, the following activation energies should be corrected to $E_i = 1.255 \times 10^5$ kJ/kmol, $E_{pbb} = 1.80 \times 10^4$ kJ/kmol, and $E_{xbt} = 1.80 \times 10^4$ kJ/kmol. In addition, Eqs. 10, 26, and 27 in the original article should be corrected to be those shown in Eqs. 25, 26 and 27:

$$G_{pi} = (R_a M_a + R_b M_b) V_r, \tag{25}$$

$$\frac{d\psi_1^P}{dt} = \frac{\psi_{1f}^P - \psi_1^P}{\theta_r} + k_{caa}\psi_0^a \cdot \psi_1^{a\cdot} + k_{cab}(\psi_0^{a\cdot}\psi_1^{b\cdot} + \psi_0^b \cdot \psi_1^{a\cdot}) + k_{cbb}\psi_0^b \cdot \psi_1^{b\cdot} + L_1\psi_1^{a\cdot} + L_2\psi_1^{b\cdot}, \tag{26}$$

$$\frac{d\psi_2^P}{dt} = \frac{\psi_{2f}^P - \psi_2^P}{\theta_r} + k_{caa}\left\{(\psi_1^{a\cdot})^2 + \psi_0^a \cdot \psi_2^{a\cdot}\right\} + k_{cab}(2\psi_1^{a\cdot}\psi_1^{b\cdot} + \psi_2^b \cdot \psi_0^{a\cdot} + x_2^a \cdot \psi_0^{b\cdot}) + k_{cbb}\left\{(\psi_1^{b\cdot})^2 + \psi_0^b \cdot \psi_2^{b\cdot}\right\} + L_1\psi_2^{a\cdot} + L_2\psi_2^{b\cdot}. \tag{27}$$

The separator and hold tank are modeled as first-order lags with constant level and residence time equal to the reactor residence time. The purge is modeled as a constant valve setting for the ratio of stream 8 to stream 7. Monomer A is methyl methacrylate, 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 8. 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 reactor medium remains moderate.

Table 8. Steady-state Operating Conditions

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

Feedforward control of recycle

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

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

$$F_{a3} = F_{a1} + y_{a2}F_2. \tag{28}$$

The nomenclature is listed in the Notation section. Since it is desired to keep the flow of monomer A to the reactor (F_{A3}) constant, Eq. 28 is solved for the fresh feed of monomer A:

$$F_{a1} = F_{a3} - y_{a2}F_2. \tag{29}$$

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

$$F_{b1} = F_{b3} - y_{b2}F_2 \tag{30}$$

$$F_{s1} = F_{s3} - y_{s2}F_2. \tag{31}$$

The implementation of Eqs. 29, 30, and 31 in the overall control strategy can be seen in Figure 11. If any feedforward

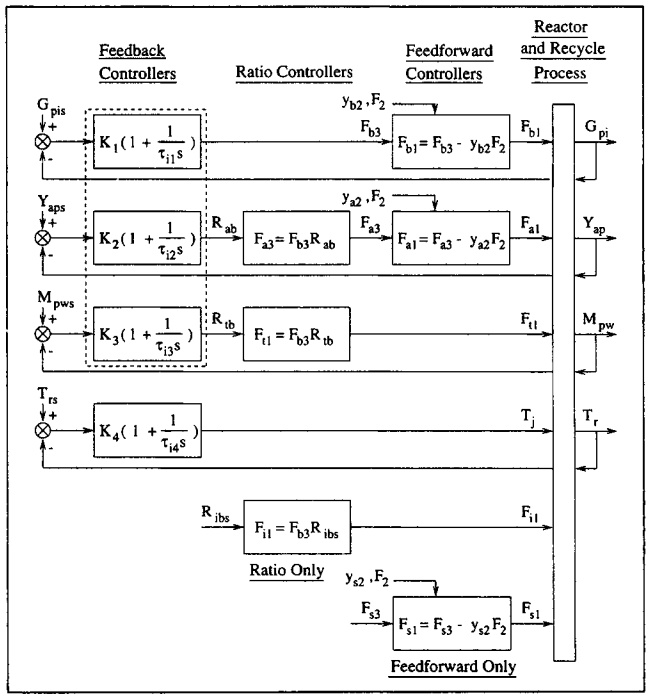

Figure 11. Control structure for multivariable case study.

control equation causes a fresh feed to go negative, the value of that fresh feed is set to zero.

Multiloop feedback control strategy for polymer properties and production rate

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 Eq. 32 and Table 9.

$$\begin{bmatrix} G_{pi}^{+} \\ Y_{ap}^{+} \\ M_{pw}^{+} \\ T_r^{+} \end{bmatrix} = [A \quad B \quad C \quad D] \begin{bmatrix} G_{bf}^{+} \\ (G_{af}/G_{bf})^{+} \\ (G_{tf}/G_{bf})^{+} \\ T_j^{+} \end{bmatrix} \qquad (32)$$

Table 10. 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 time constants in the transfer functions in Table 9 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 indicates an ill-conditioned, difficult control problem, while a smaller condition number of approximately 10 is representative of a well-conditioned, easier control problem (Skogestad and Postlethwaite, 1996).

The feedback control strategy implemented by Congalidis et al. (1989) consisted of four SISO PI controllers. The controller tuning parameters are listed in Table 10. Controller reset time, τ_I, is reported as a fraction of the reactor residence time. The temperature loop was 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 11.

Linear MPC strategy

The transfer-function matrix given by Eq. 32 and Table 9 was 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, it was not possible to obtain tuning parameters for the multivariable controller that resulted in significantly better performance than that of the multiloop PI control strategy for control of this process. The closed-loop responses using MPC were 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) noted that strongly ill-

Table 9. 4×4 Process Transfer-function Matrix

A	B	C	D
$\dfrac{0.98711(0.12011s+1)}{0.065948s^2+0.36662s+1}$	$\dfrac{0.20527}{0.4195s+1}$	0	$\dfrac{6.4595\,(0.89968s+1)}{0.065739s^2+0.29708s+1}$
0	$\dfrac{0.66018}{1.5098s+1}$	0	$\dfrac{-3.7235}{0.79590s+1}$
$\dfrac{-0.16099}{0.908165s+1}$	$\dfrac{0.49084}{1.5443s+1}$	$\dfrac{-0.19626}{2.711s+1}$	$\dfrac{-4.7145}{0.075213s^2+0.40798s+1}$
0	0	0	$\dfrac{1.0252(0.22710s+1)}{0.072732s^2+0.30978s+1}$

conditioned systems prevent the full exploitation of the interaction compensation abilities of inverse-based controllers such as MPC. They observed that the performance of MPC is not better than that of a set of well-tuned, multiloop proportional-integral-derivative (PID) controllers for such systems. Chien and Ogunnaike (1992) concluded 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 was closed with a PI controller with the same tuning parameters as used by Congalidis et al. (1989) to possibly obtain a system that was better conditioned. If a well-conditioned transfer function matrix is obtained, the interaction capabilities of MPC could be exploited for the remaining 3×3 system.

There may also be several practical reasons for closing the temperature loop with a PI controller and designing a multivariable controller for the remaining three outputs. For example, the temperature may be sampled more frequently than the other three outputs. Hence, the temperature was sampled continuously, and the remaining outputs were sampled every 0.25 h in this case study. A second reason for closing the temperature loop is that a 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. By closing the temperature loop in the hope of obtaining a better conditioned 3×3 problem, the complexity of the control problem is not reduced, it is merely a structured approach to the control design. The control problem still consists of four inputs and four outputs. Although the jacket temperature is manipulated separately by a PI controller, this loop affects the three outputs controlled by the MPC scheme. However, 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 Eq. 33 and Table 11:

$$\begin{bmatrix} G_{pi}^{+} \\ Y_{ap}^{+} \\ M_{pw}^{+} \end{bmatrix} = [E \quad F \quad G] \begin{bmatrix} G_{bf}^{+} \\ (G_{af}/G_{bf})^{+} \\ (G_{tf}/G_{bf})^{+} \end{bmatrix}. \qquad (33)$$

Table 11. 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	$\frac{0.47417}{8.6s+1}$	$\frac{-0.18288}{12.8s+1}$

The time constants in the transfer functions in Table 11 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 vs. using a multiloop PI control strategy. Comparison with Eq. 32 and Table 9 shows that the time constants are smaller with the temperature loop closed. Closing the temperature loop effectively sped up this system. This is reasonable, since the closed-loop time constant of a system is usually smaller than its open-loop time constant. The sampling time of $T_s = 0.25$ h is still in the 4–10 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 11 indicates that these PI controllers are replaced by a 3×3 MPC scheme.

Nonlinear model identification

Autoregressive plus Volterra models were identified for each of the three outputs. The entries in Table 11 that are zero pertain 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 11 were manipulated to identify the corresponding output. Hence in the identification of outputs y_1 and y_2, only u_1 and u_2 were manipulated. In the identification of y_3, only u_2 and u_3 were allowed to change values. The input profiles consisted of random, four-level sequences using $T_{cl} = 4$. The clock time, T_{cl}, was selected by trial and error. Hence the sample time was $T_s = 0.25$ h, but an input was not allowed to change values more than once per hour. The number of input–output data points was 500. Figure 12 is a plot of the raw input–output data used to iden-

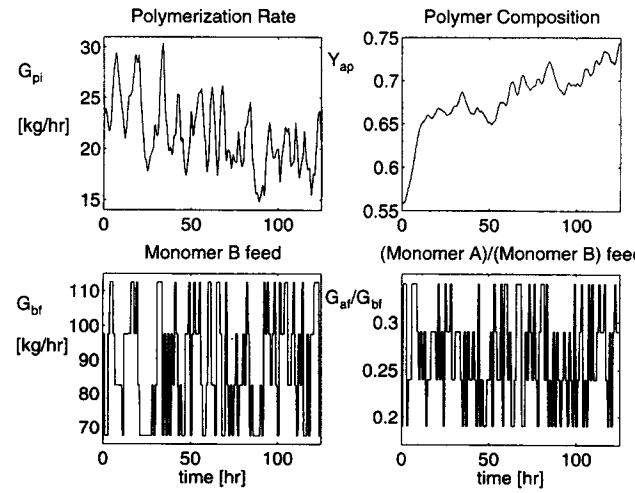

Figure 12. Input–output data used to identify y_1 (polymerization rate) and y_2 (polymer composition) (u_3 (transfer agent/monomer *B*) feed rate is held constant).

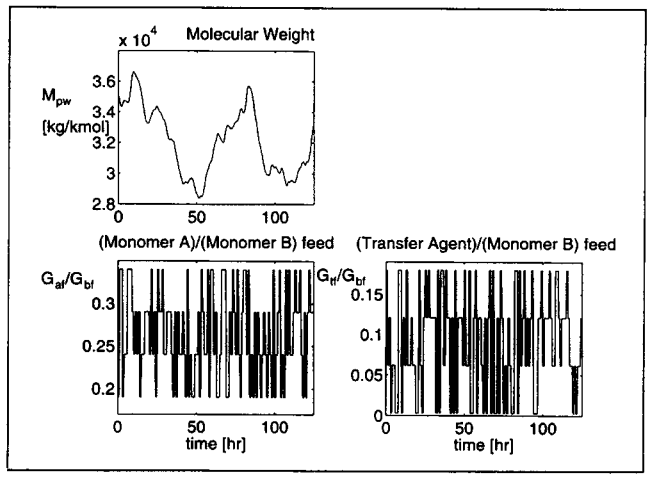

Figure 13. Input–output data used to identify y_3 (molecular weight) (u_1 (monomer *B*) feed rate is held constant).

tify models for y_1 and y_2. Figure 13 is a plot of the raw input–output data used to identify a model for y_3. The inputs and outputs were placed in scaled, deviation variables. The scale factors were chosen so that the maximum magnitude of a scaled input or output was 1.0 (Skogestad and Postlethwaite, 1996) that is,

$$\tilde{u}_i = \frac{u_i - u_{i0}}{\hat{u}_{i0}} \quad \text{and} \quad \tilde{y}_i = \frac{y_i - y_{i0}}{\hat{y}_{i0}}.$$

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 were also scaled using the same scale factors. The values of the four-level input sequences and scale factors are listed in Table 12. The stepwise model building algorithm was used with $n = 3$, and a maximum value of 4 for n_y and n_u. The linear models are listed in Table 13 and the nonlinear models are given in Table 14.

Table 12. Initial Conditions

States	
$C_{a0} = 2.4500 \times 10^{-1}$	kmol/m^3
$C_{b0} = 5.6010$	kmol/m^3
$C_{i0} = 1.8163 \times 10^{-3}$	kmol/m^3
$C_{s0} = 2.7580$	kmol/m^3
$C_{t0} = 3.6524 \times 10^{-1}$	kmol/m^3
$C_{z0} = 0.0$	kmol/m^3
$T_{r0} = 353.0161$	K
$\lambda_{a0} = 8.3092 \times 10^{-1}$	kmol/m^3
$\lambda_{b0} = 6.5524 \times 10^{-1}$	kmol/m^3
$\psi^p_{00} = 6.6571 \times 10^{-3}$	kmol/m^3
$\psi^p_{10} = 1.4133 \times 10^{2}$	kg/m^3
$\psi^p_{20} = 4.9458 \times 10^{6}$	kg^2/kmol·m^3
$C_{as0} = 2.4500 \times 10^{-1}$	kmol/m^3
$C_{bs0} = 5.6010$	kmol/m^3
$C_{is0} = 1.8163 \times 10^{-3}$	kmol/m^3
$C_{ss0} = 2.7580$	kmol/m^3
$C_{ts0} = 3.6524 \times 10^{-1}$	kmol/m^3
$C_{zs0} = 0.0$	kmol/m^3
$C_{ah0} = 2.9821 \times 10^{-1}$	kmol/m^3
$C_{bh0} = 6.8176$	kmol/m^3
$C_{ih0} = 0.0$	kmol/m^3
$C_{sh0} = 3.3571$	kmol/m^3
$C_{th0} = 0.0$	kmol/m^3
$C_{zh0} = 0.0$	kmol/m^3
$\lambda_{af} = 0.0$	kmol/m^3
$\lambda_{bf} = 0.0$	kmol/m^3
$\psi^p_{0f} = 0.0$	kmol/m^3
$\psi^p_{f} = 0.0$	kg/m^3
$\psi^p_{2f} = 0.0$	kg^2/kmol·m^3
$R_{ibs} = F_{i1s}/F_{b3s} = 1.0485 \times 10^{-3}$	
$M_a = 100.12$	kg/kmol
$M_b = 86.09$	kg/kmol
$M_i = 164.21$	kg/kmol
$M_s = 78.11$	kg/kmol
$M_t = 44.05$	kg/kmol
$M_z = 168.11$	kg/kmol
Scale factors	
$\hat{y}_1 = 7.2522$	
$\hat{y}_2 = 0.1449$	
$\hat{y}_3 = 4.8891 \times 10^{3}$	
$\hat{u}_1 = 22.5$	
$\hat{u}_2 = 0.14$	
$\hat{u}_3 = 0.15$	
$\tilde{u}_1 = [-1.0, -0.3333, 0.3333, 1.0]$	
$\tilde{u}_2 = [0.0714, 0.2857, 0.6428, 1.0]$	
$\tilde{u}_3 = [-0.18, 0.2133, 0.6066, 1.0]$	

Setpoint tracking

As a first comparison of the performance of three control strategies, a +10% setpoint change in polymer composition was considered. The tuning parameters for linear and nonlinear MPC were $m = 1$, $p = 6$, $\gamma = \text{diag}[1\ \ 1\ \ 1]$, $\lambda = \text{diag}[0\ \ 0\ \ 0]$, $\Phi_{r2} = \Phi_{r1} = \text{diag}[0.85\ \ 0.95\ \ 0.97]$. For linear MPC, L was a 17×3 zero matrix of estimator gains. For nonlinear MPC, L was a 21×3 zero matrix of estimator gains. The difference in dimensions arises from the different dimensions for the nonminimal state-space realizations given by Eq. 2 for the two models used in this case study. The nonlinear MPC

Table 13. Linear Models

$y_1(k)$		$y_2(k)$		$y_3(k)$	
Coefficients	Regressors	Coefficients	Regressors	Coefficients	Regressors
2.5746e+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	$u_1(k-2)$	−6.6750e−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	$u_2(k-1)$				
−1.0486e−01	$u_2(k-2)$				
4.6441e−02	$u_2(k-3)$				

Table 14. 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
$1.35354e+00$	$y_1(k-1)$	$9.91611e-01$	$y_2(k-1)$	$1.25762e+00$	$y_3(k-1)$
$6.95238e-02$	$u_1(k-1)$	$1.14739e-02$	$u_2(k-3)$	$-1.08825e-03$	$u_3(k-4)$
$6.24946e-02$	$u_2(k-1)$	$2.76432e-03$	$u_1(k-3)$	$-6.02635e-03$	$u_3(k-2)$
$-2.81177e-02$	$y_2(k-1)$	$4.61508e-03$	$u_2^2(k-3)u_1(k-2)$	$1.05496e-02$	$u_2(k-2)$
$-3.81750e-01$	$y_1(k-3)$	$2.31885e-03$	$u_2(k-1)$	$-2.62788e-01$	$y_3(k-4)$
$1.83873e-02$	$u_2^2(k-1)u_1(k-1)$	$2.49130e-03$	$y_1(k-3)$	$-8.30094e-02$	$y_2(k-1)$
$-1.94439e-02$	$u_1(k-2)$	$1.48864e-03$	$u_1^3(k-1)$	$-3.53710e-03$	$u_3(k-3)$
$-2.73727e-02$	$u_1^2(k-2)u_2(k-3)$	$3.56563e-03$	$u_2^3(k-2)$	$6.44994e-03$	$u_2(k-1)$
$9.83108e-03$	$u_1^2(k-1)u_1(k-1)$			$-3.52817e-03$	$u_3(k-1)$
$-1.00719e-02$	$u_2(k-2)$			$5.97053e-03$	$u_2(k-3)$
$-3.83721e-02$	$u_1(k-3)$			$-1.31409e-03$	$u_2^2(k-4)u_3(k-2)$
$-2.45688e-02$	$u_2(k-3)$			$3.15526e-03$	$u_2(k-4)u_2^2(k-1)$
$1.94764e-02$	$u_1(k-2)u_1(k-1)u_2(k-3)$			$1.44743e-03$	$u_3(k-4)u_3(k-2)u_2(k-4)$
$-2.00627e-02$	$u_2^2(k-2)\,u_1(k-2)$			$-2.03269e-03$	$u_3^2(k-4)u_2(k-1)$
$-9.57738e-03$	$u_2^3(k-3)$			$-2.46657e-03$	$u_2(k-4)u_2(k-1)u_3(k-1)$
$1.05313e-02$	$u_1(k-2)u_2(k-1)$			$7.52168e-02$	$y_2(k-4)$
				$1.42674e-03$	$u_3(k-4)u_3(k-1)u_2(k-1)$
				$8.49375e-04$	$u_2^2(k-4)u_3(k-4)$

scheme consisted 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 14 is a plot of the closed-loop performance comparison of the three control strategies. The horizontal 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 denoted by the dashed lines, and nonlinear MPC is depicted by the solid 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, while 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 control.

In Figure 15, the closed-loop performance comparison of the three control strategies for a $+20\%$ setpoint change in polymer composition is shown. The PI, linear MPC, and nonlinear MPC schemes produce responses given by the dash-dot, dashed, and solid lines, respectively. The PI control strategy requires 11.25 h to reach the bounds for the copolymer composition. Linear MPC brings the polymer composition to its specification bounds after 9.25 h, while nonlinear MPC needs

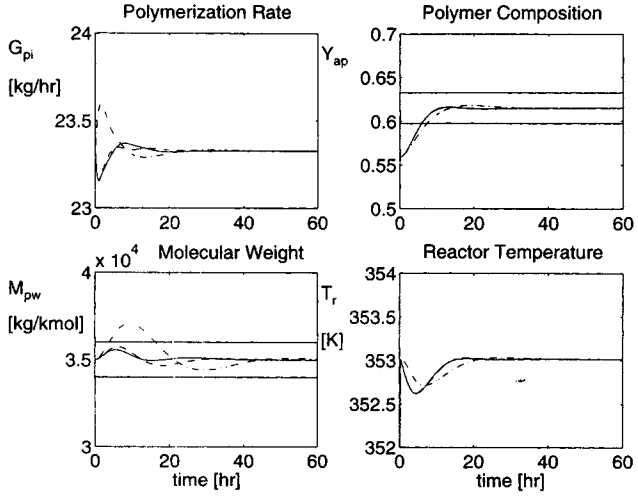

Figure 14. Closed-loop simulation for a filtered-step setpoint change of +10% in copolymer composition.

Four PI controllers: dash-dot lines; one PI controller and 3×3 linear MPC: dashed lines; one PI controller and 3×3 MPC with nonlinear model for M_{pw}: solid lines.

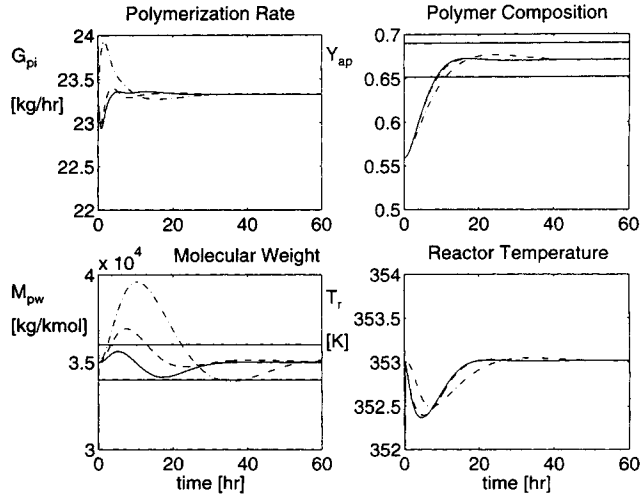

Figure 15. Closed-loop simulation for a filtered-step setpoint change of +20% in copolymer composition.

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}: solid lines:

only 8.75 h. The difference in performance is most clearly shown in the plot for molecular weight in Figure 15. The PI scheme permits the molecular weight to reach 39,600 kg/kmol, which is far outside the product quality bounds. The molecular weight reenters the specification region at 22.5 h, but 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 autoregressive plus 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 14 and 15 the multiloop PI control scheme results in tighter control of the temperature than the temperature control in the strategies using 3×3 MPC plus one PI controller. This result is expected, since a tradeoff was made in designing the MPC schemes. 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.

Disturbance rejection

In addition to smooth setpoint 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 that 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. Equation 11 in the Congalidis et al. (1989) article was incorrectly coded and led to an incorrect open-loop simulation in response to an inhibitor disturbance in figure 10 of that article. For an inhibitor disturbance of 4 parts per 1,000 (mole basis) in the fresh feed, the true open-loop behavior of this process is shown in Figure 16. Results shown in Figure 17 indicate the futility of only 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 (34,994.7 kg/kmol) more with the temperature loop closed (31,134 kg/kmol) than with the temperature loop open (32,153 kg/kmol).

Figure 18 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

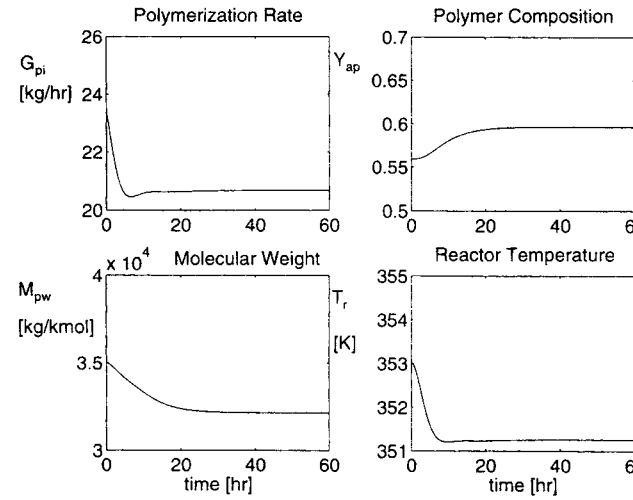

Figure 16. Open-loop simulation for an inhibitor disturbance of 4 parts per 1,000 (mole basis).

The steady-state value of the molecular weight is 32,153 kg/kmol.

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. Hence, only nonlinear MPC is able to continue producing a product that meets specifications in responses to this disturbance. Linear MPC could be tuned more aggressively enabling it to also make specification product in the presence of the inhibitor disturbance. However, the aggressive tuning yields a very oscillatory response for y_1 for the +20% setpoint change in polymer composition. Hence just as in the SISO case study, the filter in the feedback path for linear

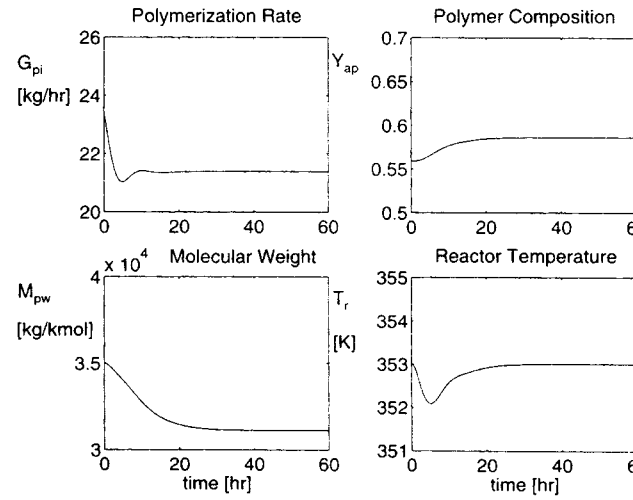

Figure 17. Open-loop simulation for an inhibitor disturbance of 4 parts per 1,000 (mole basis) with temperature feedback control.

The steady-state value of the molecular weight is 31,134 kg/kmol.

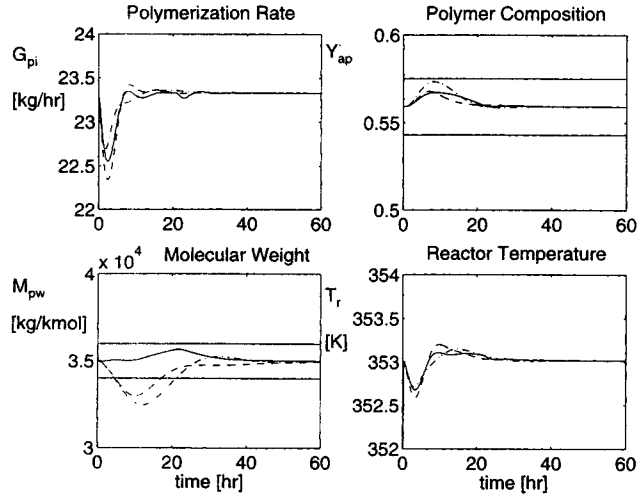

Figure 18. Closed-loop simulation for an unmeasured step disturbance in inhibitor from its nominal value of 0 to 4 parts per 1,000 (mole basis).

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}: solid lines.

MPC must be detuned to account for plant–model mismatch. Using an autoregressive plus Volterra model for y_3, however, results in less plant–model mismatch. The controller using the nonlinear model does not have to be detuned to the degree that the linear model-based controller does, thus yielding improved closed-loop performance.

Figure 19 is a plot of the manipulated variable profiles for all three control schemes for the inhibitor disturbance simulation; this gives insight into the performance of all three control strategies. From the results given in Table 11, it is evident from the time constants that y_1 (polymerization rate) responds fastest to changes in the inputs. Figures 16, 17, and

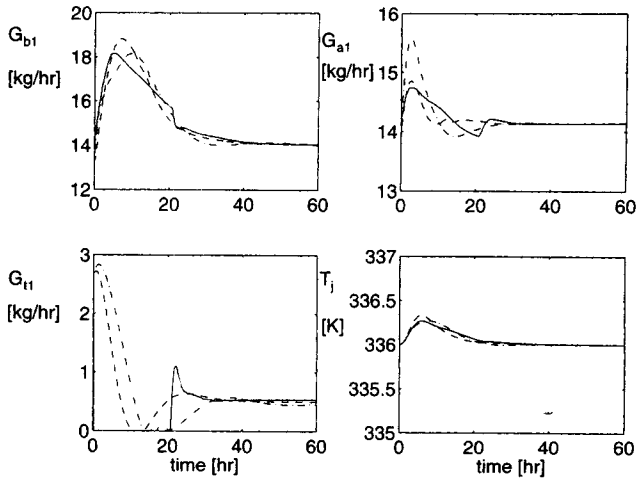

Figure 19. Manipulated variable profiles for an unmeasured step disturbance in inhibitor from its nominal value of 0 to 4 parts per 1,000 (mole basis).

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}: solid lines.

18 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 (polymer composition), 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 19 saturate at some point throughout this simulation for all three control schemes. The PI controller was coded with antiwindup 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 corrective 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 autoregressive plus 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.

The performance improvements noted in the previous two sections raise the possibility of additional performance improvement if a 3×3 nonlinear MPC scheme were used. A closed-loop simulation for a +20% setpoint change in polymer composition using a 3×3 nonlinear MPC scheme based on the autoregressive plus Volterra models in Table 14 resulted in a worse performance than that of any of the control strategies used to generate Figure 15. Open-loop validation tests using input sequences similar to those used during identification showed that the nonlinear models for $\tilde{y}_1$ and $\tilde{y}_2$ were slightly better than the corresponding linear models, while the nonlinear model for $\tilde{y}_3$ was significantly better than the corresponding linear model. Evaluation of the models using step inputs showed that the nonlinear models were poor at describing low-frequency (steady-state) behavior. Hence, the identified nonlinear models are more accurate at higher frequencies than at steady state. A similar situation was encountered by Sriniwas et al. (1995) in identifying polynomial ARMA models using input sequences that used values from a uniform distribution. The resulting models were incorporated in a nonlinear MPC scheme for control of a high-purity distillation column. As the prediction horizon p increased, model prediction became poorer and control performance deteriorated. As p is increased, MPC is detuned and ap-

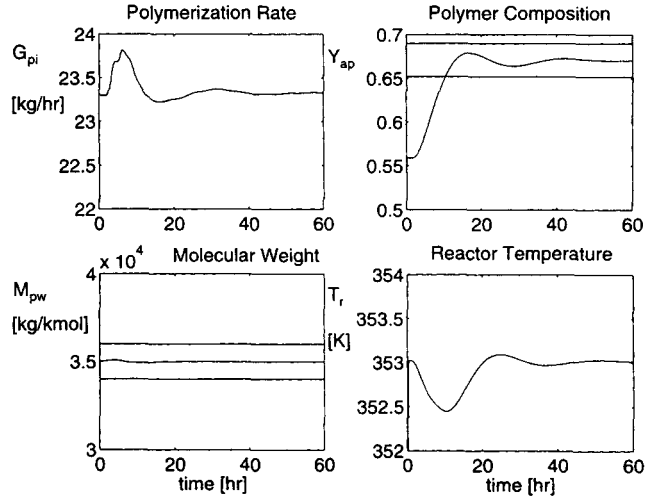

Figure 20. Closed-loop simulation for a filtered-step setpoint change of +20% in copolymer composition using one PI controller and 3×3 nonlinear MPC with state estimation.

proaches a steady-state controller. Hence if the model is poor at describing low-frequency behavior, large values of p can cause performance degradation. Chien and Ogunnaike (1992) noted that models used in MPC should be accurate in the frequency range where they are to be operated in the closed-loop.

The closed-loop performance was considerably improved through the use of state estimation. Figure 20 is a closed-loop simulation for the same setpoint change using a PI controller and a 3×3 nonlinear MPC scheme with state estimation. The tuning parameters were $m = 1$, $p = 6$, $\gamma = \text{diag}[1 \quad 1 \quad 1]$, $\lambda = \text{diag}[2.0 \quad 0.1 \quad 0.1]$, $\Phi_{r2} = \Phi_{r1} = \text{diag}[0.89 \quad 0.96 \quad 0.98]$, and L was a 23×3 matrix of estimator gains given by

$$L = \begin{bmatrix} 0.4 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{bmatrix}.$$

The nonminimal state-space realization has 23 states for the 3×3 nonlinear input–output model. Comparison of the results shown in Figures 15 and 20 indicate that the 3×3 nonlinear MPC scheme maintains the molecular weight very close to its setpoint for this setpoint change in polymer composition. However, the polymerization rate and polymer composition have more oscillatory behavior than that shown in Figure 15 where only one autoregressive plus Volterra model was used in the 3×3 MPC scheme. The inhibitor disturbance forced the closed-loop system to be unstable when 3×3 nonlinear MPC was used.

The unstable disturbance rejection performance was attributed to using a poor model with a large estimator gain. Ricker (1990) noted that using a poor model with an estimator gain that is too large can lead to poor performance or an unstable closed-loop system. Sriniwas et al. (1995) observed poorer disturbance rejection performance in the control of the high-purity distillation column using nonlinear MPC with state estimation than that encountered using two multiloop PI controllers (Chien and Ogunnaike, 1992). Performance was improved using on-line parameter adaptation.

Although the performance was poor for the 3×3 nonlinear MPC scheme due to the difficulty of identifying accurate nonlinear models for y_1 (polymerization rate) and y_2 (polymer composition), there may not be a significant performance improvement if accurate nonlinear models were obtained for y_1 and y_2. For the simulations considered in this work, the polymer composition may be modeled quite accurately with a linear model. Hence, there would not be a significant improvement in modeling accuracy if a nonlinear model were employed for y_2. The polymerization rate does behave nonlinearly. However, in practice the flow of monomer B to the reactor may not be allowed to change very rapidly. The reason for this is that an increase in flow of monomer B to the reactor increases the polymerization rate. Since the polymerization is exothermic, the temperature of the reactor increases until the PI controller brings the reactor temperature back to its setpoint. However, the increase in reactor temperature causes the polymer properties to change. To avoid this situation, it may be required that the rate of change in the flow of monomer B to the reactor is constrained. This corresponds to detuning the controller and limits the achievable closed-loop performance. Even if an accurate nonlinear model for y_1 were identified, the resulting 3×3 control scheme may need to be detuned such that its performance is no better than that obtained using a single autoregressive plus Volterra model for y_3. A stability analysis was not performed for this case study, since the larger number of parameters associated with a multivariable problem would lead to increased conservatism.

Conclusions

The control algorithm described in this article offers a number of useful advantages over alternative data-driven nonlinear control algorithms. In particular, the Volterra plus autoregressive model structure admits straightforward identification using the "plant-friendly" four-level input sequence described in this work. This input sequence is less demanding than a continously distributed uniform distribution that is typically reported for the identification of polynomial ARMA models. A second advantage of the proposed use of this model structure in nonlinear MPC is that it leads to a sixth-order nonlinear program to be solved at each sampling interval. A sixth-order optimization problem is more tractable than the nonlinear programs encountered in several other published nonlinear model predictive control schemes (Economou et al., 1986; Li and Biegler, 1988; Eaton and Rawlings, 1990; Hernández and Arkun, 1993). Closed-loop stability analysis was facilitated by the global BIBO stability results existent for the Volterra plus autoregressive model structure; however, the derived closed-loop stability results yielded roughly 40% improvement over previously published results due to the minimally conservative formulation adopted in this work. In terms of closed-loop performance, improvement over PI and linear model-predictive control was demonstrated for a single-variable case study for both setpoint tracking and dis-

turbance rejection. A significantly more complex multivariable case study was also employed to demonstrate the practical applicability of the approach, and again both PI and linear model-predictive control were outperformed for both setpoint tracking and disturbance rejection.

Acknowledgment

The authors acknowledge funding from the National Science Foundation NYI program (Grant CTS-9257059) and from DuPont. The authors also thank Dr. J. P. Congalidis and Dr. J. R. Richards of DuPont for their helpful correspondence regarding the copolymerization reactor case study, as well as collaboration with Drs. B. A. Ogunnaike and R. K. Pearson of DuPont.

Notation

A = discrete-time A matrix
b = discrete-time B matrix
C_I = molar concentration of the initiator, kmol/m^3
$C_{I_{in}}$ = molar concentration of the initiator in the initiator inlet stream, kmol/m^3
C_m = molar concentration of the monomer, kmol/m^3
$\hat{d}$ = disturbance estimate
D_0 = molar concentration of the dead polymer chains, kmol/m^3
D_1 = mass concentration of the dead polymer chains, kg/m^3
f^* = initiator efficiency
F = polynomial functions; molar flow rate, kmol/h; volumetric flow rate, m^3/h
G = mass flow rate, kg/h
h = discrete polynomial function
k = kinetic rate contant; time index
k_{fm} = kinetic rate constant for chain transfer to monomer, m^3/(kmol h)
k_I = kinetic rate constant for initiation, m^3/(kmol·h)
k_p = kinetic rate constant for propagation, m^3/(kmol·h)
k_{T_c} = kinetic rate constant for termination by coupling, m^3/(kmol·h)
k_{T_d} = kinetic rate constant for termination by disproportionation, m^3/(kmol·h)
K_c = controller gain
L = intermediate variable in MWD calculations
m = summation index
M = molecular weight, kg/kmol
M_m = molecular weight, kg/kmol
n = summation index
N = degree of polynomial
P = dead polymer
P_0 = molar concentration of the live polymer chains, kmol/m^3
R = reaction rate, kmol/m^3·h
t = time, h
T = temperature, K; chain transfer agent
u_0 = nominal value of the input
$\boldsymbol{u}$ = vector of future inputs
Δu = change in the input
$\boldsymbol{\Delta u}$ = vector of future input changes
V = reactor volume, m^3
$\boldsymbol{x}$ = state vector
$\hat{\boldsymbol{x}}$ = corrected state vector
x_{r1} = disturbance-model state
x_{r2} = reference-model state
y = output; mole fraction
Y = mole fraction
y_0 = bias term; nominal value of the output
$\bar{y}$ = plant measurement
$\hat{y}$ = model prediction
y_{r1} = disturbance-model output
y_{r2} = reference-model output
γ = output weight
Γ_{r1} = disturbance-model B matrix
Γ_{r2} = reference-model B matrix
θ = regressor coefficient; residence time, s
λ = input change weight
$\bar{\sigma}$ = maximum singular value
τ_I = integral time, h
Φ_{r1} = disturbance-model A matrix
Φ_{r2} = reference-model A matrix
ψ = moment of molecular-weight distribution

Subscripts and superscripts

c = termination by coupling (combination)
f = feed to the reactor
i = initiator
j = cooling jacket; stream counter; summation index
l = summation index
o = summation index
p = output prediction horizon; number of outputs; propagation; dead polymer
r = reactor; summation index
t = chain transfer agent
x = chain transfer
$(\bullet)$ = free radical
$(^+)$ = fractional deviation variable, for example, $Y_{ap}^+ = (Y_{ap} - Y_{aps})/Y_{aps}$

Literature Cited

Åström, K. J., and B. Wittenmark, *Computer-controlled Systems Theory and Design*, 2nd ed., Prentice Hall, Englewood Cliffs, NJ (1990).
Akaike, H., "Information Theory and an Extension of the Maximum Likelihood Principle," *Proc. Int. Symp. on Information Theory, Suppl. to Problems of Control and Information Theory*, IEEE, p. 267 (1972).
Baillagou, P. E., and D. S. Soong, "Major Factors Contributing to the Nonlinear Kinetics of Free-radical Polymerization," *Chem. Eng. Sci.*, **40**(1), 75 (1985a).
Baillagou, P. E., and D. S. Soong, "Molecular Weight Distribution of Products of Free Radical Nonisothermal Polymerization with Gel Effect. Simulation for Polymerization of Poly(methyl Methacrylate)," *Chem. Eng. Sci.*, **40**(1), 87 (1985b).
Balas, G. J., J. C. Doyle, K. Glover, A. Packard, and R. Smith, *μ-Analysis and Synthesis Toolbox User's Guide*, The Mathworks, Natick, MA (1995).
Bartee, J. F., and C. Georgakis, "Identification and Control of Bilinear Systems," *Proc. Amer. Control Conf.*, Chicago, p. 2576 (1992).
Billings, S. A., and I. J. Leontaritis, "Parameter Estimation Techniques for Nonlinear Systems," *Proc. IFAC Symp. on Identification and Systems Parameter Estimation*, Washington, DC, p. 505 (1982).
Boyd, S., and L. O. Chua, "Fading Memory and the Problem of Approximating Nonlinear Operators with Volterra Series," *IEEE Trans. Circuits Syst.*, **CAS-32**(11), 1150 (1985).
Bristol, E. H., "On a New Measure of Interaction for Multivariable Process Control," *IEEE Trans. Automat. Contr.*, **AC-11**(1), 133 (1966).
Chien, I. L., and B. A. Ogunnaike, "Modeling and Control of High-Purity Distillation Columns," AIChE Meeting, Miami (1992).
Chow, T., E. Eskow, and R. Schnabel, "Algorithm 739: A Software Package for Unconstrained Optimization Using Tensor Methods," *ACM Trans. Math. Softw.*, **20**(4), 518 (1994).
Congalidis, J. P., J. R. Richards, and W. H. Ray, "Feedforward and Feedback Control of a Solution Copolymerization Reactor," *AIChE J.*, **35**(6), 891 (1989).
Daoutidis, P., M. Soroush, and C. Kravaris, "Feedforward/Feedback Control of Multivariable Nonlinear Processes," *AIChE J.*, **36**(10), 1471 (1990).
Davies, W. D. T., *System Identification for Self-Adaptive Control*, Wiley-Interscience, London (1970).
Dennis, J. E., Jr., and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall, Englewood Cliffs, NJ (1983).
Doyle, J. C., "Analysis of Feedback Systems with Structured Uncertainties," *Proc. IEEE*, **6**, 242 (1982).
Doyle, F. J., III, B. A. Ogunnaike, and R. K. Pearson, "Nonlinear Model-based Control Using Second-order Volterra Models," *Automatica*, **31**(5), 697 (1995).
Doyle, F. J., III, A. K. Packard, and M. Morari, "Robust Controller Design for a Nonlinear CSTR," *Chem. Eng. Sci.*, **44**(9), 1929 (1989).
Eaton, J. W., and J. B. Rawlings, "Feedback Control of Chemical

Processes Using On-line Optimization Techniques," *Comput. Chem. Eng.*, **14**(4/5), 469 (1990).
Economou, C. G., and M. Morari, "Internal Model Control. 6. Multiloop Design," *Ind. Eng. Chem. Process Des. Dev.*, **25**, 411 (1986).
Economou, C., M. Morari, and O. Palsson, "IMC. 5. Extension to Nonlinear Systems," *Ind. Eng. Chem. Process Des. Dev.*, **25**, 403 (1986a).
Eskinat, E., S. H. Johnson, and W. L. Luyben, "Use of Hammerstein Models in Identification of Nonlinear Systems," *AIChE J.*, **37**(2), 255 (1991).
Fruzzetti, K., A. Palazoglu, and K. McDonald, "Nonlinear Model Predictive Control Using Hammerstein Models," *J. Proc. Cont.*, (1995).
García, C. E., "Quadratic Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process," AIChE Meeting, San Francisco (1984).
García, C. E., and M. Morari, "Internal Model Control: 3. Multivariable Control Law Computation and Tuning Guidelines," *Ind. Eng. Chem. Process Des. Dev.*, **24**, 484 (1985).
Gattu, G., and E. Zafiriou, "Nonlinear Quadratic Dynamic Matrix Control with State Estimation," *Ind. Eng. Chem. Res.*, **31**, 1096 (1992).
Hamer, J. W., T. A. Akramov, and W. H. Ray, "The Dynamic Behavior of Continuous Polymerization Reactors: II," *Chem. Eng. Sci.*, **36**(12), 1897 (1981).
Hernández, E., *Control of Nonlinear Systems Using Input-Output Information*, PhD Thesis, Georgia Institute of Technology, Atlanta (1992).
Hernández, E., and Y. Arkun, "Control of Nonlinear Systems Using Polynomial ARMA Models," *AIChE J.*, **39**(3), 446 (1993).
Hill, C. G., Jr., *An Introduction to Chemical Engineering Kinetics & Reactor Design*, Wiley, New York (1977).
Hirschorn, R. M., "Invertibility of Multivariable Nonlinear Control Systems," *IEEE Trans. Automat. Cont.*, **AC-24**(6), 855 (1979).
Kirnbauer, T., and H. Jörgi, "Nonlinear Predictive Control Using Volterra Series Models," *Proc. of the IFAC-NOLCOS 2*, Lake Tahoe, CA, p. 558 (1992).
Kortmann, M., K. Janiszowski, and H. Unbehauen, "Application and Comparison of Different Identification Schemes Under Industrial Conditions," *Int. J. Control*, **48**(6), 2275 (1988).
Li, W. C., and L. T. Biegler, "Process Control Strategies for Constrained Nonlinear Systems," *Ind. Eng. Chem. Res.*, **27**, 1421 (1988).
Li, W. C., and L. T. Biegler, "Multistep, Newton-type Control Strategies for Constrained, Nonlinear Processes," *Chem. Eng. Res. Des.*, **67**, 562 (1989).
Li, W. C., and L. T. Biegler, "Newton-type Controllers for Constrained, Nonlinear Processes with Uncertainty," *Ind. Eng. Chem. Res.*, **29**, 1647 (1990).
Ljung, L., *System Identification: Theory for the User*, Prentice Hall, Englewood Cliffs, NJ (1987).
Maner, B. R., "Polymerization Reactor Control Using Computationally Tractable Input-Output Models," PhD Thesis, Purdue Univ., West Lafayette, IN (1996).
Maner, B. R., F. J. Doyle III, B. A. Ogunnaike, and R. K. Pearson, "A Nonlinear Model Predictive Control Scheme Using Second Order Volterra Models," *Proc. Amer. Control Conf.*, Baltimore, p. 3253 (1994).
Marmarelis, P. Z., and V. Z. Marmarelis, *Analysis of Physiological Systems: The White-Noise Approach*, Plenum Press, New York (1978).
McFarlane, R. C., and D. E. Rivera, "Identification of Distillation Systems," *Practical Distillation Control*, Chap. 7, W. L. Luyben, ed., Van Nostrand Reinhold, New York, p. 96 (1992).
Moore, C. F., "Selection of Controlled and Manipulated Variables," *Practical Distillation Control*, Chap. 8, W. L. Luyben, ed., Van Nostrand Reinhold, New York, p. 140 (1992).
Morari, M., "Design of Resilient Processing Plants: III. A General Framework for Assessment of Dynamic Resilience," *Chem. Eng. Sci.*, **38**(11), 1881 (1983).
Morari, M., and E. Zafiriou, *Robust Process Control*, Prentice Hall, Englewood Cliffs, NJ (1989).
Murtagh, B. A., and M. Sauders, *MINOS 5.1 User's Guide*, Tech. Rep. SOL 83-20R, Stanford Univ., Stanford, CA (1987).
Narendra, K. S., and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. Neural Networks*, **1**(1), 4 (1990).
Nowak, R. D., and B. D. Van Veen, "Random and Pseudorandom Inputs for Volterra Filter Identification," *IEEE Trans. Signal Process.*, **42**(8), 2124 (1994).
Ogunnaike, B. A., R. K. Pearson, N. Samardzija, and J. D. Bomberger, "Low Order Empirical Modeling for Nonlinear Systems," *ADCHEM*, Kyoto, Japan, p. 41 (1994).
Pearson, R. K., "Nonlinear Input/Output Modeling," *ADCHEM*, p. 1 (1994).
Prett, D. M., and C. E. García, *Fundamental Process Control*, Butterworth-Heinemann, Boston (1988).
Priestley, M. B., *Nonlinear and Nonstationary Time Series Analysis*, Academic Press, San Diego, CA (1988).
Ray, W. H., "Polymerization Reactor Control," *IEEE Contr. Syst. Mag.*, **CSM-6**((4)S), 3 (1986).
Ricker, N. L., "Model Predictive Control with State Estimation," *Ind. Eng. Chem. Res.*, **29**, 374 (1990).
Ricker, N. L., "Model Predictive Control: State of the Art," *Chemical Process Control-CPCIV*, Y. Arkun and W. H. Ray, eds., AIChE, p. 271 (1991).
Ricker, N. L., and J. H. Lee, "Nonlinear Model Predictive Control of the Tennessee Eastman Challenge Process," *Comput. Chem. Eng.*, **19**(9), 961 (1995).
Rugh, W. J., *Nonlinear System Theory—The Volterra/Wiener Approach*, Johns Hopkins Press, Baltimore (1981).
Schmidt, A. D., A. B. Clinch, and W. H. Ray, "The Dynamic Behavior of Continuous Polymerization Reactors—III," *Chem. Eng. Sci.*, **39**(3), 419 (1984).
Schmidt, A. D., and W. H. Ray, "The Dynamic Behavior of Continuous Polymerization Reactors—I," *Chem. Eng. Sci.*, **36**, 1401 (1981).
Skogestad, S., and I. Postlethwaite, *Multivariable Feedback Control*, Wiley, New York (1996).
Soroush, M., and C. Kravaris, "Nonlinear Control of a Batch Polymerization Reactor: An Experimental Study," *AIChE J.*, **38**(9), 1429 (1992).
Sriniwas, G. R., Y. Arkun, I. L. Chien, and B. A. Ogunnaike, "Nonlinear Identification and Control of a High-purity Distillation Column: A Case Study," *J. Process Control*, **5**(3), 149 (1995).
Subba-Rao, T., "On the Theory of Bilinear Time Series Models II," Tech. Rep. 121, Univ. of Manchester Institute of Science and Technology, Manchester, UK (1979).
Weischedel, K., and T. J. McAvoy, "Feasibility of Decoupling in Conventionally Controlled Distillation Columns," *Ind. Eng. Chem. Fundam.*, **19**, 379 (1980).
Wellons, M. C., and T. F. Edgar, "The Generalized Analytical Predictor," *Ind. Eng. Chem. Res.*, **26**, 1523 (1987).
Yeo, Y. K., and C. Williams, "Bilinear Model Predictive Control," *Ind. Eng. Chem. Res.*, **26**, 2267 (1987).
Yu, C. C., and W. L. Luyben, "Design of Multiloop SISO Controllers for Multivariable Processes," *Ind. Eng. Chem. Process Des. Dev.*, **25**, 498 (1986).
Zames, G., "On the Input-Output Stability of Time-varying Nonlinear Feedback Systems: I. Conditions Derived Using Concepts of Loop Gain, Conicity, and Positivity," *IEEE Trans. Automat. Contr.*, **AC-11**(2), 228 (1966).
Zhu, X., and D. E. Seborg, "Nonlinear Model Predictive Control Based on Hammerstein Models," *Proc. Int. Symp. on Process Systems Engineering*, Vol. II, E. S. Yoon, ed., Korean Institute of Chemical Engineers, Seoul, p. 995 (1994).

Manuscript received May 20, 1996, and revision received Feb 20, 1997.