On-Line Nonlinear Model-Based Estimation and Control of a Polymer Reactor

Rajendra K. Mutha and William R. Cluett
Dept. of Chemical Engineering, University of Toronto, Toronto, Ont., Canada M5S 3E5

Alexander Penlidis
Dept. of Chemical Engineering, University of Waterloo, Waterloo, Ont., Canada N2L 3G1

Polymer reactor control problems often lack frequent measurements of polymer properties, while other reactor environment measurements, such as temperatures, are readily available. The experimental application of a new estimator for handling such multirate measurement systems is presented. The estimator, in conjunction with a detailed nonlinear model and all measurements at their respective sampling rates, is able to provide frequent predictions of the process variables related to the quality (weight-average molecular weight) and quantity (monomer conversion) of the polymer product. The experimental application of a multivariable, nonlinear model-based predictive controller using reactor temperature and initiator flow rate as the manipulated variables for controlling weight-average molecular weight and monomer conversion is presented. The control strategy is designed to handle control nonaffine systems (nonlinear in the manipulated variables), while allowing for the incorporation of rate and amplitude constraints on the manipulated variables. A continuous methyl methacrylate solution polymerization pilot plant was used for the experiment.

Introduction

The overall objective of our recent research has been to design an industrially relevant, model-based estimation and control strategy for complex chemical processes, with particular interest in polymer reactors. A number of articles have been written on the topic of estimation and control of polymer reactors, but we will limit our literature survey to recent articles involving experimental work. Reviews by MacGregor et al. (1984), Ray (1985), Elicabe and Meira (1988), and Dimitratos et al. (1994) provide more detailed surveys of the literature. Common to many of these recent articles is the use of state estimation and nonlinear control theory to address the challenges of polymer reactor control. Tzouanas and Shah (1989) used a solution methyl methacrylate (MMA) reactor for evaluation and comparison of various control strategies with monomer conversion as the main target. Ogunnaike and Gopalratnam (1991) presented a two-tier multirate control-system structure for an industrial terpolymerization reactor. Dimitratos et al. (1991) evaluated four different types of Kalman filter on a seeded semicontinuous emulsion copolymerization of vinyl acetate/butyl acrylate. Van Doremaele et al. (1992) studied copolymer composition control of a semicontinuous emulsion copolymerization process. Soroush and Kravaris (1992) evaluated a globally linearizing control method for achieving servo and regulatory temperature control of a batch solution MMA polymerization reactor. Evaluations were further extended to multivariable nonlinear control of monomer conversion and reactor temperature on a continuous MMA polymer reactor with a nonsingular (Soroush and Kravaris, 1993) and singular characteristic matrix (Soroush and Kravaris, 1994) by dynamic input/output linearizing state feedback. Ellis et al. (1994) achieved on-line estimation and control of molecular-weight distribution on a batch solution MMA polymerization process, using a Kalman-filter-based, two-time-scale estimation scheme along with a PID controller. Urretabizkaia et al. (1994) achieved on-line terpolymer composition control for a semicontinuous emulsion polymerization process, using a nonlinear adaptive

Correspondence concerning this article should be addressed to W. R. Cluett.
Current address of R. K. Mutha: Aspen Technology, Inc., 9896 Bissonnet, Houston, TX 77036.

controller based on a simplified mathematical model of the process. Finally, Elicabe et al. (1995) applied a Kalman filter to the data collected during the semibatch emulsion copolymerization of a ternary system, using concepts from systems theory to reduce the degrees of freedom of a Kalman filter and thereby provide an easier method for tuning the filter.

One of the remaining obstacles in the control of polymer reactors is the lack of adequate and frequent measurements. For example, temperature measurements are available instantaneously, whereas polymer samples are only taken infrequently for quality assessment and are associated with large analysis times. This necessitates the use of multirate measurement estimators for polymer reactor systems. Several state-estimation schemes based on the extended Kalman filter (EKF) have been proposed in the literature for process state estimation and subsequent inclusion in process-control schemes in order to attempt to compensate for this lack of adequate and frequent measurements. However, very few of these estimation algorithms are designed to handle systems with multirate measurements. In the first part of our research, we developed a practical estimator using a fixed-lag smoothing-based EKF for dealing with nonlinear systems having multirate measurements (Mutha et al., 1997a). The estimator is designed for systems with multirate measurements, and uses each measurement multiple times to achieve good convergence properties and robustness to state and measurement noise. It can handle several measurements with different amounts of delay and is also applicable to systems with variable measurement delays.

In the polymer industry, the quantity (monomer conversion) and quality (number and/or weight-average molecular weight) are the two most important controlled variables for homopolymer systems. Initiator flow rate and reactor temperature can be used as manipulated variables for controlling the quantity and quality of a homopolymer, although in most polymer reactor control studies reported in the literature, reactor temperature has been chosen as the controlled variable. Their use as manipulated variables leads to a control nonaffine (CNA) system because of the nonlinear relationship that exists between these inputs and the controlled outputs. In addition, the manipulated variables must be constrained to lie within some prespecified bounds, and, due to physical limitations of the process, their rate of change is often restricted. In the second part of our research, we developed a practical nonlinear model-based predictive controller (NLMPC) for handling CNA systems, which also allows for the incorporation of rate and amplitude constraints on the manipulated variables (Mutha et al., 1997b). The algorithm is based on a reinterpretation of the prediction equation as a Taylor series. The MPC quadratic cost function translates to a nonlinear cost function for CNA systems. The resulting nonlinear programming problem is translated to a quadratic programming problem by an iterative approximation and is thereby made feasible for real-time applications.

This article presents an experimental application of both our state estimation and nonlinear model-based predictive control schemes to a continuous MMA solution polymerization pilot plant. The MMA process model used by both the estimator and the controller is a special case of a more generic model for homopolymerization systems described in Gao and Penlidis (1996). The generic homopolymerization model accounts for mass balances for each reactant, and energy balances and molecular weight equations for both linear and branched systems. The next section of this article describes the MMA process and related measurements. This is followed by a brief review of the multirate measurement-based estimator and an extensive experimental evaluation of the estimator for both single- and multi-rate measurement systems. The final section of the article summarizes the unique features of the CNA-NLMPC algorithm, followed by an experimental evaluation of the control algorithm, highlighting the importance of accounting for the nonaffine system behavior.

MMA Process Description

The continuous MMA solution polymerization pilot-plant setup used in this research has been described in detail by Chien (1992), and Chien and Penlidis (1994a,b). The pilot plant consists of a 1.3 litre stainless steel overflow reactor. Three variable-speed, computer-controlled, high-precision remote setpoint pumps are used for pumping monomer, solvent, and initiator solution into the reactor. These pumps are solenoid-operated electric-diaphragm metering pumps by ProMinent (Model BX2i1006SB). The reaction product flows out of the reactor via an overflow line into a cooler. The cooled output stream is then passed through a deaerator (in order to eliminate air bubbles), and then through an on-line densitometer and viscometer. The densitometer measurements are used for inferring the percentage conversion (X) of the monomer. The viscosity measurements along with X and reactor-temperature measurements are used for inferring the weight-average molecular weight ($\overline{M}_w$) of the polymer product.

The three reagents used during an experimental run are MMA as monomer, toluene as solvent, and 2,2′-azobisisobutyronitrile (AIBN) as initiator. The MMA and toluene were both supplied from Aldrich and were both of commercial grade with 99% purity. The AIBN was supplied from Dupont. All chemicals were used as received and were *not* purified prior to an experimental run. Typical experimental operating conditions are shown in Table 1.

Relationships between process model, estimator, available measurements, and controlled variables

Consider that the MMA process can be described by the following mathematical model:

$$\frac{dx}{dt} = F(x,u), \tag{1}$$

where x contains the process states and u represents the manipulated variables. Assume that measurements y are avail-

Table 1. Experimental Conditions of the MMA Solution Polymerization Reactor

Reactor temperature	70–90°C
Reactor volume (V_R)	1.3 L
Initiator concentration in feed	0.032 mol·L^{-1}
Monomer feed rate	0.0133 L·min^{-1}
Solvent feed rate	0.02 L·min^{-1}
Residence time	≈ 40 min

able from this process and that these measurements are also estimated according to

$$\hat{y} = G(\hat{x}). \tag{2}$$

The aim of the estimator is to make these model predictions track the actual process measurements by updating the estimated process states $\hat{x}$ using the available measurements, that is, $\hat{x}$ is updated by an EKF using equations of the form:

$$\hat{x}(k|k-1) = \int_{(k-1)*T_s}^{k*T_s} F(\hat{x}(k-1|k-1), u(k))\, dt$$

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)(y - \hat{y}),$$

where k is the discrete-time index, T_s is the sampling interval, and $K(k)$ is the Kalman gain matrix.

In several process-control applications, the measured variables and controlled variables are not identical. Assume that the controlled variables y_c can be estimated as

$$\hat{y}_c = H(\hat{x}). \tag{3}$$

If on-line measurements of y_c are either not feasible or the measurements are associated with significant noise or large measurement delays, the model equations (Eqs. 1 and 3), with or without the estimator, can be used to predict $\hat{y}_c$, which can in turn be used as the actual controlled variables.

In the MMA system under consideration, the process measurements (y) are density (ρ), viscosity (η), and the rate of heat liberated during polymerization (H). The controlled variables (y_c) are conversions (X) and weight-average molecular weight ($\overline{M}_w$).

For this MMA system, a direct relationship between y and y_c exists:

$$y_c = L(y), \tag{4}$$

and is developed in the next subsection. This relationship will be used for inferring y_c from y for making comparisons with the model-predicted controlled variables $\hat{y}_c$ during the evaluation of the estimator. However, it must be noted that the type of relationship in Eq. 4 may not exist for all systems, in which case the estimator evaluation could only be performed by comparing y with $\hat{y}$.

Inferred values of X and $\overline{M}_w$

By utilizing the density difference between the polymer and the unreacted monomer, the monomer conversion can be readily calculated. The densities of pure solvent (ρ_s), MMA monomer (ρ_m), and MMA polymer (ρ_p) are functions of temperature and are well known from the literature. For a 40:60 feed ratio of monomer to solvent, the average density of the inflow stream (ρ_{in}) is

$$\rho_{in} = 0.4\rho_m + 0.6\rho_s. \tag{5}$$

Monomer conversion (X) can then be inferred using the measured density ρ of the product stream according to

$$X = \frac{\frac{\rho_{in}}{\rho} - 1}{1 - \frac{\rho_p}{\rho_m}}. \tag{6}$$

The viscosity of a polymer solution can be related to number-average/weight-average molecular-weight averages of the polymer. Jo and Bankhoff (1976) developed an empirical relationship between $\overline{M}_w$, X, and solution viscosity (η):

$$\ln \eta = \beta_1 + \beta_2 [X\overline{M}_w]^i. \tag{7}$$

A modified version of this correlation was adopted for the MMA pilot plant setup by Lawrence (1994):

$$\ln \eta = \left(\alpha_1 + \frac{\alpha_2}{T}\right) + \alpha_3 X \overline{M}_w^{1/2}, \tag{8}$$

where T is the reactor temperature in degrees Celsius. $\overline{M}_w$ has been scaled such that the predicted values of $\overline{M}_w$ obtained from Eq. 8 need to be multiplied by 10^3 to obtain the actual value. The structure of the preceding correlation was arrived at after evaluating variations of Eq. 7, based on experimental data collected on-line from the viscometer and densitometer, and off-line data for X using gravimetric analysis and for $\overline{M}_w$ using gel permeation chromatography (GPC). In the preceding correlation, $\overline{M}_w$ is preferred over $\overline{M}_n$ since $\overline{M}_w$ is a better indicator of the state of the polymerization at higher conversion levels compared to $\overline{M}_n$. A theoretical justification based on polymerization kinetics for selecting the preceding structure of the correlation is also provided in Lawrence (1994).

The manipulated variables of the process are initiator flow rate and reactor temperature. Both of these inputs affect both controlled variables, X and $\overline{M}_w$. Typical input sequences used for collecting X and $\overline{M}_w$ data in order to estimate the parameters of Eq. 8 are shown in Figure 1.

The best set of parameters in Eq. 8 obtained by Lawrence (1994), with 95% confidence bounds, is

$$\alpha_1 = -1.048 \pm 3.84(10)^{-3}$$

$$\alpha_2 = 19.137 \pm 0.924$$

$$\alpha_3 = -9.095 \pm 2.819(10)^{-5} \tag{9}$$

Figure 2 shows that the inferred values of $\overline{M}_w$, obtained by combining the measurements of η, T, and the inferred values of X with Eq. 8, are in good agreement with the $\overline{M}_w$ values determined via GPC except for low $\overline{M}_w$. Figure 2 also compares the inferred values of X (via Eq. 6) with the X values determined from off-line gravimetric analysis. To account for the deviations between the inferred $\overline{M}_w$ and GPC-determined $\overline{M}_w$ in the low $\overline{M}_w$ region, a linear correction was added to the predicted $\overline{M}_w$, that is, whenever the inferred $\overline{M}_w$ (scaled value) was less than 35, the following relationship was activated:

$$\overline{M}_w = \frac{\overline{M}_w - 20}{0.41176}. \tag{10}$$

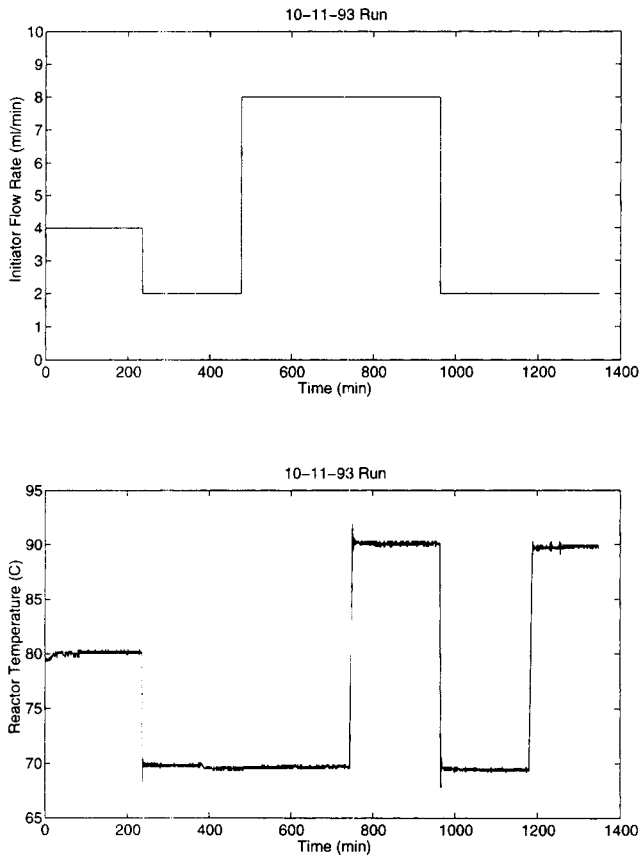


Figure 1. Typical input sequences used for developing $\overline{M}_w$ correlation (Lawrence, 1994).

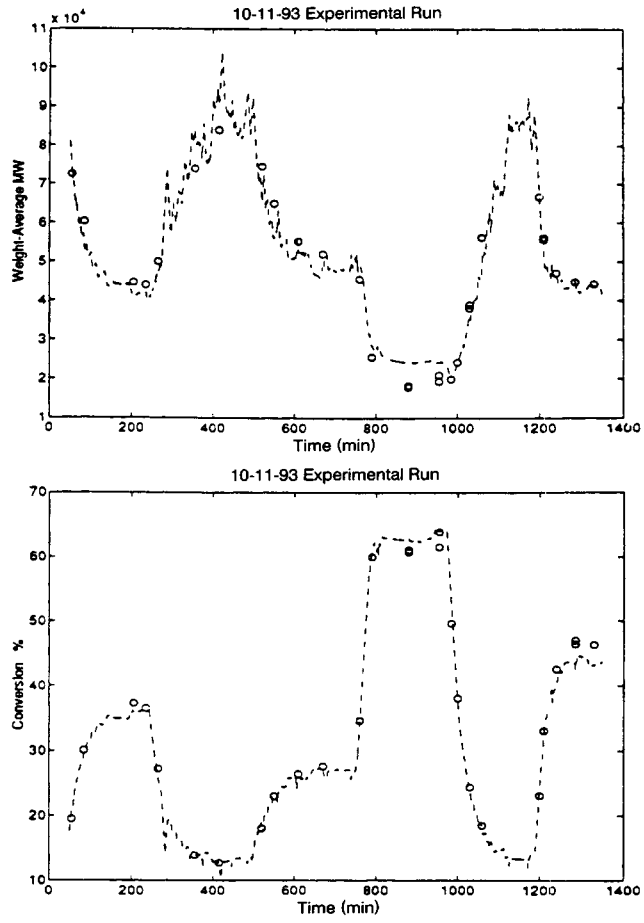


Figure 2. Inferred $\overline{M}_w$ and conversion (---) vs. off-line $\overline{M}_w$ and conversion (○) (Lawrence, 1994).

Equation 10 maps the inferred $\overline{M}_w$ in the low region from 28–35 to 18–35.

This modified correlation was validated using data from an independent experimental run performed by Lawrence (1994). The experiment was carried out using a constant initiator solution flow rate of 4 mL/min and reactor temperature of 80°C and with injections of known amounts of 2,2-diphenyl-1-picrylhydrazil (DPPH) and Topanol A into the system. DPPH and Topanol A are inhibitors, and their presence will reduce the polymerization rate and decrease molecular weight by consuming radicals. Figure 3 shows the inferred $\overline{M}_w$ along with the GPC data. It also shows the inferred conversion and the conversion values obtained from off-line gravimetric analysis. DPPH was injected into the reactor at time 187 min, and Topanol A was injected at time 382 min, both for 15-min periods. The inferred values of X and $\overline{M}_w$ closely match the X and $\overline{M}_w$ values determined using off-line analysis.

Measurements

Three measurement signals (y) are used by the estimator. In addition to the direct measurements obtained from the densitometer and viscometer, energy-balance equations are also solved on-line to infer the rate of heat liberated during polymerization.

As mentioned earlier, state estimators work on the principle of updating states based on the difference between measured variables (y) and their predicted values ($\hat{y}$) obtained from a model. This subsection describes the measurements to be taken from the MMA process and how these measurements will be predicted.

Process Measurements. An on-line Micro Motion Model D digital density meter is used for measuring density of the product stream and in turn for inferring monomer conversion X by Eq. 6. The densitometer is coupled with a Micro Motion Flow Transmitter (RFT9712). The densitometer measures density by measuring the change in the natural frequency of a hollow oscillator when filled with different liquids.

Viscosity of the product stream is measured by a Contraves Covimat 105-DD in-line rotational viscometer. The Covimat 105-DD is a continuous rotational viscometer with a concentric cylinder measuring system to determine the dynamic viscosity of a process fluid. The measuring bob rotates within a stationary measuring cell and is immersed in the liquid to be measured. The measuring bob is driven by a magnetic coupling by a synchronous motor suspended in a low-friction-bearing system from a calibrated top-mounted torsion bar. The viscous drag, which is proportional to viscosity, acting on the measuring bob displaces the motor in a direction opposite to the rotational direction of the bob until an equilibrium is established. This angle of deflection is measured by a variable displacement transformer, and the signal is transmitted

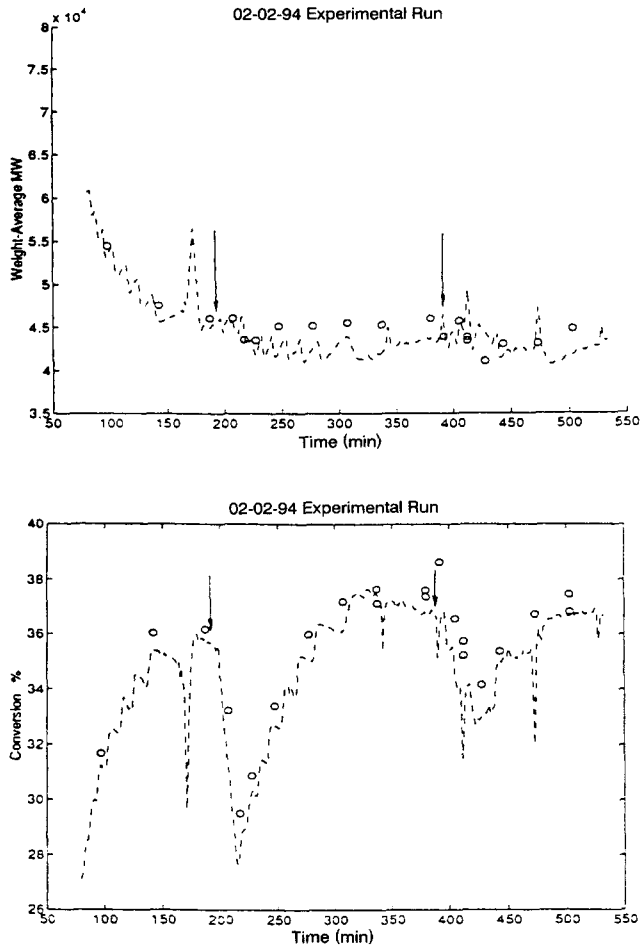


Figure 3. Inferred $\bar{M}_w$ and conversion (---) vs. off-line $\bar{M}_w$ and conversion (○) for validating Eq. 8 (Lawrence, 1994).

as an output signal. The correlation obtained using glycerol standards enables the output signal to be transformed to viscosity units. The correlation for converting the measured voltage signal (V in volts) to viscosity (η in cP) is

$$\eta = (V - 1.2262)/0.0865.$$

The measured η, T, and inferred X are then used in Eq. 8 to infer $\bar{M}_w$.

The rate of heat liberated from the polymerization (H) is inferred by performing an energy balance on the reactor and its contents:

$$H = \sum_i M_i Cp_i \frac{dT}{dt} + \sum_j F_j Cp_j (T - T_{\text{in}}) + Q_{\text{water}} - Q_{\text{mantle}}, \tag{11}$$

where T is the filtered reactor temperature, M_i are the mass of the reactor and its contents with Cp_i denoting the associated heat capacities, F_j are the mass flow rates of solvent and monomer/polymer with Cp_j as their heat capacities, and T_{in} is the temperature of the inlet solvent and monomer streams.

In all calculations, the filtered reactor temperature T was determined at each sampling instant from:

$$T(k) = \alpha T_{\text{meas}}(k) + (1 - \alpha) T(k-1), \tag{12}$$

where T_{meas} is the actual measurement from the reactor and α is a filter parameter between 0 and 1. (dT/dt) was calculated numerically by a second-order difference approximation. The Q_{water} in Eq. 11 is the rate of heat transferred to the cooling water (coils), and is calculated as:

$$Q_{\text{water}} = F_{\text{water}} Cp_{\text{water}} (T_{\text{water,out}} - T_{\text{water,in}}),$$

where $T_{\text{water,in}}$ is the temperature of the water entering the cooling water coil, $T_{\text{water,out}}$ is the temperature of the water leaving the cooling water coil, and F_{water} is the mass flow rate of the water through the cooling water coil. The Q_{mantle} is the rate of heat transferred from the electrical heating mantle to the reactor and is calculated as:

$$Q_{\text{mantle}} = U_{\text{mantle}} (T_{\text{mantle}} - T),$$

where T_{mantle} is the average temperature of the mantle and U_{mantle} is the overall heat transfer coefficient. U_{mantle} was determined experimentally.

The density and viscosity measurements involve a transportation lag of about 10 min, as the sensors are located some distance downstream of the reactor on the product line. This 10 min transportation lag is equivalent to taking a measurement directly at the reactor outlet with a sample analysis time of 10 min, that is, measurement delay. The measurement of the rate of heat liberated by the polymerization is not associated with any transportation lag or measurement delay.

Predicted Measurements. An estimator requires all of the measured values to be also predicted by a model. Monomer conversion is predicted by the model as:

$$\hat{X} = \frac{[P]}{[M] + [P]}, \tag{13}$$

where $[P]$ is the polymer concentration and $[M]$ is the monomer concentration.

The weight-average molecular weight is predicted by the model through the following equation:

$$\hat{\bar{M}}_w = (MW_m) \frac{\mu_2}{\mu_1}, \tag{14}$$

where MW_m denotes the monomer molecular weight, and μ_1 and μ_2 stand for the first and second moment of the dead polymer molecular-weight distribution, respectively. Expressions for μ_1 and μ_2 are given in Gao and Penlidis (1996).

Using $\hat{X}$ from Eq. 13 and $\hat{\bar{M}}_w$ from Eq. 14, density of the reacting mixture ρ is predicted by Eq. 6, and solution viscosity $\hat{\eta}$ is predicted by Eq. 8.

A linear correlation was developed to predict the heat liberated ($\hat{H}$) from the predicted conversion $\hat{X}$:

$$\hat{H}(\text{cal/min}) = 2{,}169.7\hat{X} + 1{,}267.245. \tag{15}$$

Reactor temperature control

The reactor itself is surrounded by a heating mantle and the heat load can be controlled remotely from a computer. For all the experimental work, the heat load was set to a constant value of 65% of its maximum power. The reactor is also equipped with a cooling coil. Temperature control is achieved by manipulating the water flow rate through the cooling coil. The water flow rate through the cooling coil is manipulated by the opening or closing of a valve, which in turn is manipulated by the computer signal sent to the valve. The signal sent to the cooling water valve is the number of seconds of clockwise or anticlockwise rotation (opening or closing) of the valve spindle.

Reactor temperature control is achieved using a cascade control configuration. The outer loop controls the reactor temperature (T) and the inner loop controls the position of the cooling water valve spindle (V). The outer-loop manipulated signal is a setpoint for the position of the water-valve spindle. Based on this setpoint of the valve position (Vsp), the inner loop calculates the duration (DUR) in the number of seconds the valve spindle is to turn clockwise or anticlockwise (for opening or closing, respectively). This cascade strategy is illustrated in Figure 4.

The inner loop tuning used by Ogunye (1994) and Lawrence (1994) was adopted for this work. The inner loop is a variation of the standard discrete-time proportional-integral (PI) controller in the velocity form:

$$\mathrm{DUR}(k) = \frac{K_p}{Vsp(k)^2}[\epsilon(k) - \epsilon(k-1)] + K_i\epsilon(k),$$

where $\epsilon(k) = Vsp(k) - V(k)$, and K_p and K_i are the controller tuning parameters. In this form, the proportional gain K_p of the PI controller decreases inversely as the square of Vsp. This controller seems to perform reasonably well and therefore was used without modifications. This loop was executed every 20 s.

For the outer loop, which controls the reactor temperature, advanced control schemes like LQG and self-tuning control algorithms were designed and used by Ogunye (1994). For this work, a standard discrete-time, velocity form PI controller was used:

$$\Delta u(k) = K_p\left[\epsilon(k) - \epsilon(k-1) + \frac{T_s}{T_i}\epsilon(k)\right],$$

where T_i is the integral time constant; $\Delta u(k) = u(k) - u(k-1)$ and $\epsilon(k) = Tsp(k) - T_{\mathrm{meas}}$; and u refers to the valve spindle position setpoint (Vsp). The temperature-control loop was also executed every 20 s. The PI controller parameters were $K_p = 0.017$ (position of valve spindle per degree C) and $T_i = 3.75$ min. The reactor temperature setpoint of the outer loop (Tsp) is used as a manipulated variable in this study.

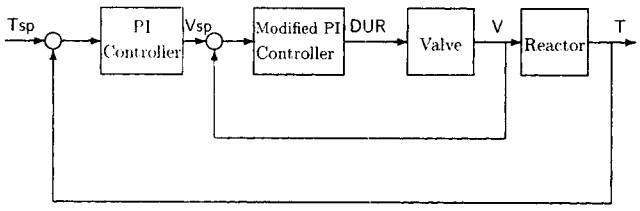


Figure 4. Cascade temperature controller setup for the MMA solution polymerization pilot plant.

Experimental Evaluation of the Estimator

EKF with fixed-lag smoothing for multirate systems

In Mutha et al. (1997a), we proposed an EKF algorithm with fixed-lag smoothing for multirate systems, where the term "extended" refers to the fact that the algorithm is applied in this case to a linearized version of the nonlinear MMA process model. The proposed algorithm is based on a "boot-strap" application of filtering and smoothing. The value of the lag used for smoothing is referred to as the smoothing interval (SI). Two EKFs are used in this scheme: one for filtering the states (EKF1) based on the process model, and the second for smoothing the states (EKF2). EKF1 is used for estimating the states at time k with the available measurements at time k. EKF2 is used for smoothing the states back to time $k - SI$. In general, the states of EKF2 are a subset of the states of EKF1, that is, it is often unnecessary to smooth all of the states of EKF1.

As for the convergence of the estimated states, which are observable only with the slow measurements, their true values depend on how frequently slow measurements are taken and on their respective measurement delay. With polymerization processes, measurements are often associated with long measurement delays and are taken infrequently. The convergence properties of these states can be enhanced by using the available measurements several times. In other words, a slow measurement should be used at each and every sampling interval until the next slow measurement becomes available, along with the new fast measurements, to estimate the states. The proposed algorithm tries to compensate for infrequent slow measurements by repeated use of the available slow measurements. This is accomplished by the coordinated action of EKF1 and EKF2.

The MMA process used in this study is examined as both a single-rate system and as a multirate (two-rate) system. Throughout all of the experimental work, the sampling interval was taken as 2 min. As a single-rate system, H is available at every sampling interval without measurement delay, and ρ and η are available at every sampling interval with a measurement delay of five sampling intervals (10 min). As a multirate system, H (the fast measurement) is available at every sampling interval without measurement delay, and ρ and η (the slow measurements) are available at every fifth sampling interval with a measurement delay of five sampling intervals. The key parameter of the estimator algorithm, SI, should be chosen at least equal to the largest of the sum of the measurement delay and the number of sampling intervals between consecutive slow measurements, both expressed in sampling intervals, for each slow measurement considered separately. By doing this, the slow measurements are used in the reintegration of EKF1 from $k - SI$ to k. If the SI is greater than the measurement delay by an integer n, then the slow measurements are used $(n+1)$ times by the algorithm. All of the fast measurements are used SI times by the algorithm.

Experimental results

The main objective with the experiments is to demonstrate the performance of the proposed estimator for single-rate and multirate measurements. Three different experimental runs were performed to collect data for this purpose. We refer to these experiments as runs 1, 2 and 3. The set of data from run 1 was used for modifying the termination rate constant (K_{to}) and initiator efficiency (f) values from the set of rate parameters presented by Gao and Penlidis (1996) in order to achieve reasonable predictions of the collected experimental data, for the complete operating range of temperatures and initiator flow rates. All the figures in this section show measurements at the time they become available, as opposed to when they were taken. The process was perturbed by making changes in the reactor temperature and flow rate of the initiator (AIBN). The process variables of interest are conversion (X) and weight-average molecular weight ($\overline{M}_w$). For numerical robustness, the flow rate of the initiator was scaled by a factor of 10,000 in order that both inputs were of similar magnitude.

The data collected from all three runs were used off-line several times for evaluating the performance of the estimator with different tuning parameters. Nonetheless, the implementation of the estimator algorithm was always carried out with the real-time application in mind. This was achieved by making sure that the data collected from the process was made available to the estimator algorithm at the appropriate sampling interval and by enforcing limits on computation time.

Model Predictions without Estimator. This subsection compares experimental data with the model predictions, without use of the estimator. The input sequences implemented on the process for run 1 are shown in Figure 5. The valve connected to the cooling water coil of the reactor broke down after 25 sampling intervals and, as a result, the reactor temperature ran away to almost 100°C. The problem was rectified by the 45th sampling interval of the run. The inferred trajectories of X and $\overline{M}_w$ from process measurements and Eqs. 6 and 8, respectively, and model-predicted trajectories of $\hat{X}$ and $\hat{\overline{M}}_w$ using Eqs. 13 and 14, respectively, for the implemented input sequences of Figure 5 are shown in Figure 6. It is seen from Figure 6 that there is a reasonable match between model predictions and measurement-based inferred values for X and $\overline{M}_w$, except for the steep peaks of $\overline{M}_w$, during the period between the 110–130 sampling intervals and at the end of the run.

The input sequences implemented on the process for run 2 are shown in Figure 7. The measurement-based inferred and model-predicted trajectories of X and $\overline{M}_w$ for the implemented input sequences of Figure 7 are shown in Figure 8. It is seen from Figure 8 that the overall trend of the model predictions is close to the experimental values for both X and $\overline{M}_w$. However, there is a steady-state mismatch for X around sampling interval 130. The process had not reached steady-state for most of the other step changes in the manipulated variables, and therefore no inference can be made about the steady-state mismatch at other operating points.

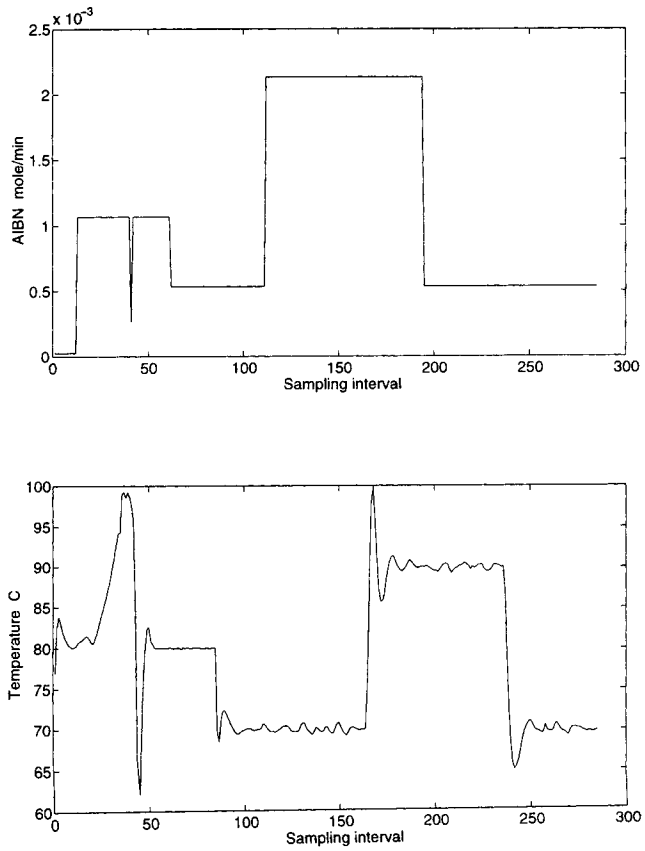


Figure 5. Input sequences for run 1.

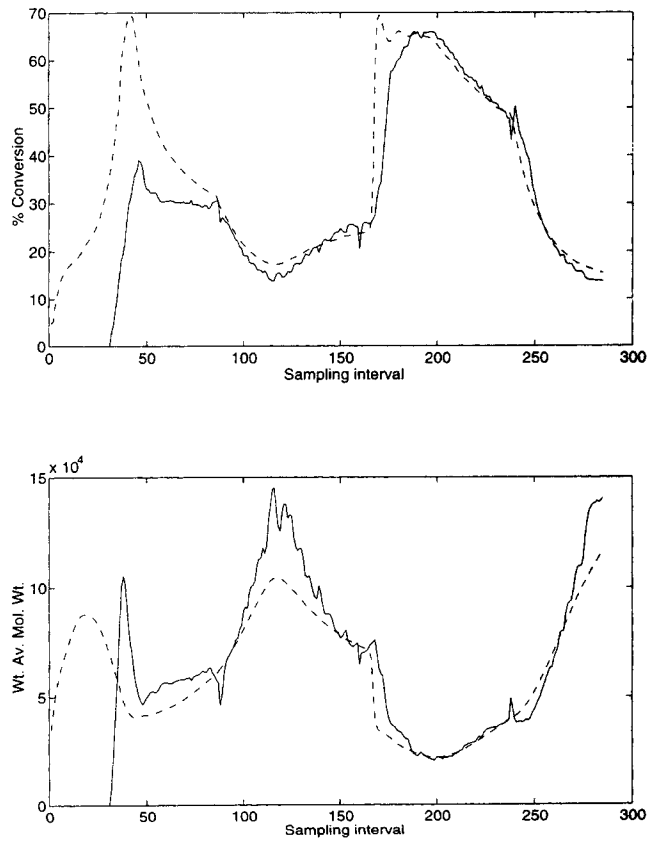


Figure 6. Measured values (—) and model predictions (---) for X and $\overline{M}_w$ (run 1).

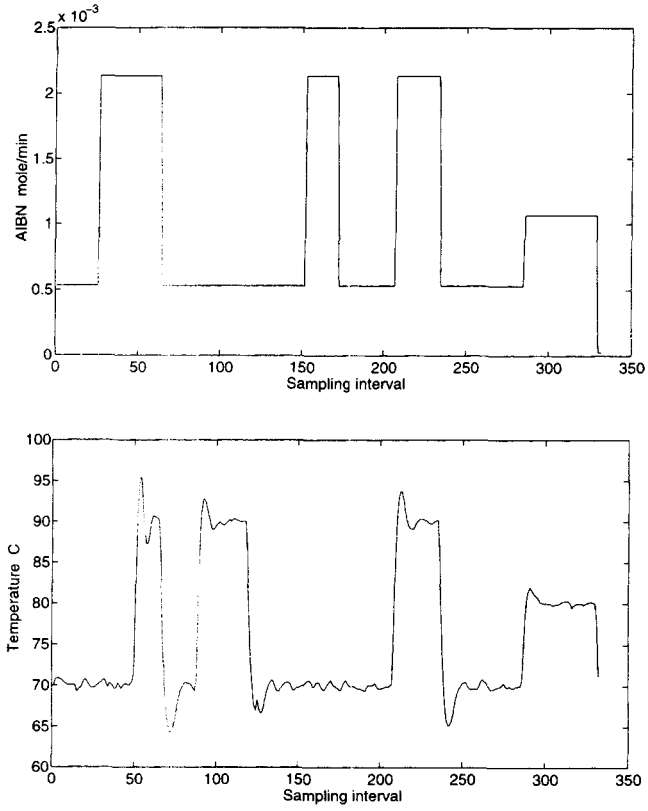


Figure 7. Input sequences for run 2.

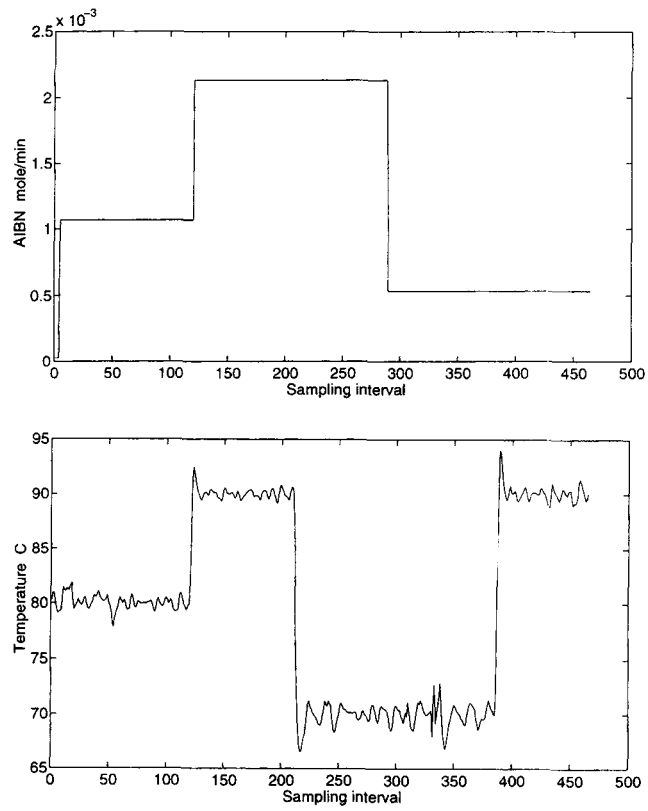


Figure 9. Input sequences for run 3.

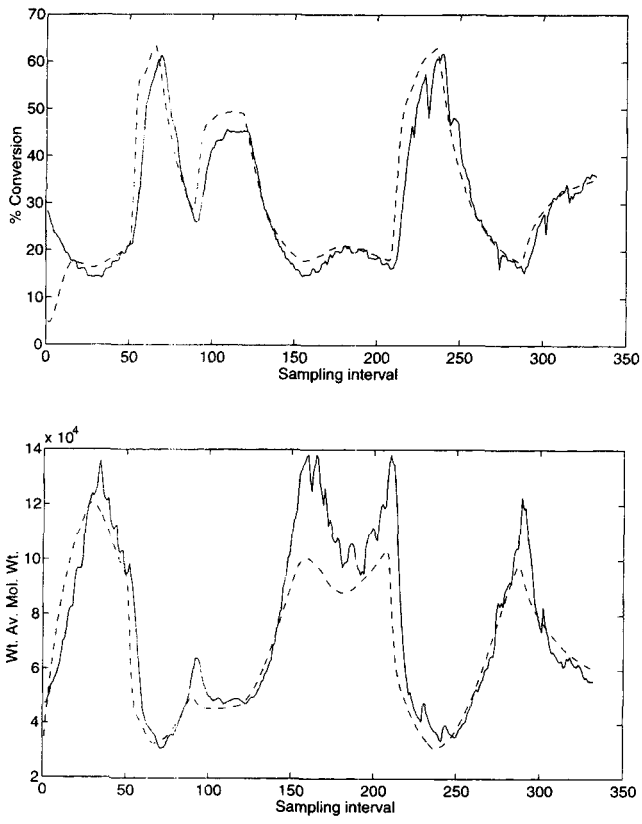


Figure 8. Measured values (—) and model predictions (---) for X and $\overline{M}_w$ (run 2).

The trend of not being able to track the steep peaks of $\overline{M}_w$ is again visible in this run, especially between sampling intervals 150–165, 200–215, and 285–300.

The input sequences implemented on the process for run 3 are shown in Figure 9. The measurement-based inferred and model-predicted trajectories of X and $\overline{M}_w$ for the implemented input sequences of Figure 9 are shown in Figure 10. It is seen from Figure 10 that the overall trend of the model predictions is close to the measurement-based inferred values for both X and $\overline{M}_w$. However, there are steady-state mismatches in X at four different ranges of sampling intervals: 180–210, 285–295, 370–385, and 450–465. The mismatch is particularly large in the range 430–465. For this run, the model underpredicts for large values of $\overline{M}_w$ and there is a large mismatch between predicted and measured $\overline{M}_w$ at sampling intervals around 380.

The inferred measurement of the rate of heat liberated from the reactor based on Eq. 11 and its predicted value through Eq. 15 are shown in Figure 11, for all three runs. For run 1, the predictions are poor, particularly for the first half of the run. For run 2 and run 3, the predictions are reasonable and show a good trend. However, for some ranges of sampling intervals, the prediction error is significant. Nonetheless, this measurement will be used by the estimator. As there is lower confidence in this measurement, a large variance will be associated with it.

Estimator states

The predictions by the process model without the estimator were able to approximately follow the changes in the pro-

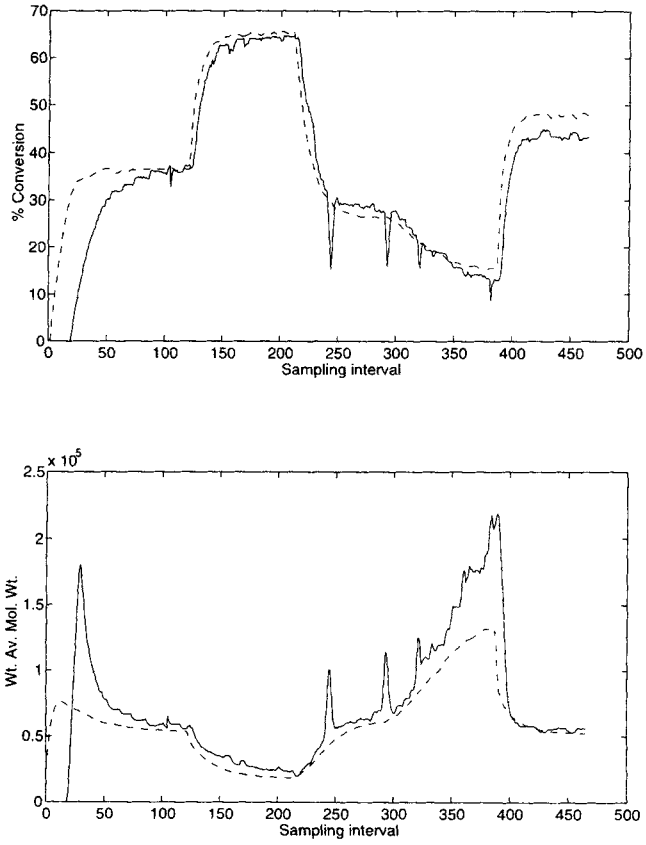


Figure 10. Measured values (—) and model predictions (---) for X and $\bar{M}_w$ (run 3).

cess outputs. However, there were several regions with substantial offset between the model predictions and the process measurements. Subsequently, the estimator has been implemented on this MMA solution polymerization process to update the process states and parameters of the model in order to achieve better predictions.

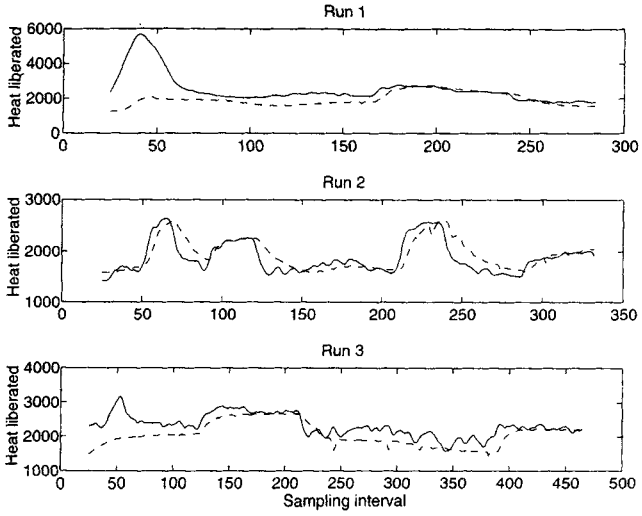


Figure 11. Measured rate of heat liberated (—) and the predicted rate of heat liberated (---), for all three runs.

The stationary states selected for EKF1 were the total moles of polymer in the reactor ($V_R[P]$) and the product of V_R and the second moment of the polymer molecular weight distribution ($V_R \mu_2$). These states directly affect X and $\bar{M}_w$, respectively. Their final selection was based on trying several different combinations in the estimator. The presence of mismatch between the measured data and predicted values from the model indicates the need for incorporating nonstationary states in the MMA model to achieve integral action in the estimator.

The MMA pilot-plant setup has only two independent measurements in density and viscosity, as the heat of reaction is also related to conversion. Therefore, two nonstationary states were incorporated in the model, to meet the consistency requirement of having the number of nonstationary states equal to the number of measurements (Parrish and Brosilow, 1988). The selected nonstationary states for this system were:

1. Multiplicative nonstationary state (x_1^s) with radical concentration ($[R^{\cdot}]$);
2. Multiplicative nonstationary state (x_2^s) with $V_R \mu_2$.

An increase in (x_1^s) increases X and decreases $\bar{M}_w$ via $[R^{\cdot}]$, and an increase in (x_2^s) increases $\bar{M}_w$ via $V_R \mu_2$. The nonstationary states have been selected such that they can absorb the biases between process measurements and model predictions. Both of these states were initialized at unity. The estimator compensates for the mismatch between model predictions and process measurements by varying the values of these nonstationary states to achieve better predictions. For more information on nonstationary states, the reader is referred to Stephanopoulos and San (1984) and Kozub and MacGregor (1992).

The states selected for EKF2 were also $V_R[P]$ and $V_R \mu_2$. No nonstationary states were used with EKF2. In summary, EKF1 consisted of four states, two stationary ($V_R[P]$ and $V_R \mu_2$) and two nonstationary (x_1^s and x_2^s), and EKF2 consisted of only two stationary states ($V_R[P]$ and $V_R \mu_2$). The minimum requirement in selecting the estimator states is state observability by the available measurements. Each state in the smoother (EKF2) affects the prediction of the slow measurements directly, that is, $V_R[P]$ affects X and $V_R \mu_2$ affects $\bar{M}_w$. Therefore, whenever a slow measurement is available, the states at time $k - SI$ are updated by the smoother EKF2, assuming k is the current sampling interval. The states selected in the smoother EKF are based on enhancing the convergence properties of the weakly observable states. The values of the estimator tuning parameters are shown in the Appendix. Tuning of the estimator was carried out in a time-consuming, trial-and-error manner.

Estimator performance with single-rate measurements

In this section, the performance of the estimator is illustrated for single-rate measurements, that is, all three measurements were available at each and every sampling interval of 2 min. However, note that the measurements from the densitometer and viscometer are associated with a transportation lag of five sampling intervals (10 min).

The estimator was implemented on the data from run 1 with an SI value of 10. The suggested minimum for this case is 6. The performance of the estimator along with model pre-

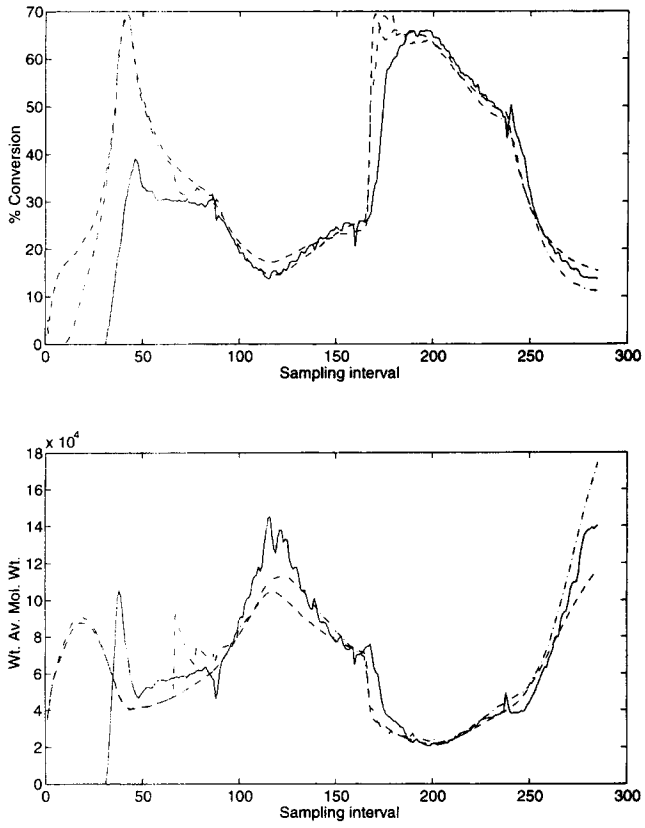


Figure 12. Measured values (—), single-rate measurement-based estimates with *SI* = 10 (- · -), and model predictions without the estimator (---) for *X* and $\overline{M}_w$ (run 1).

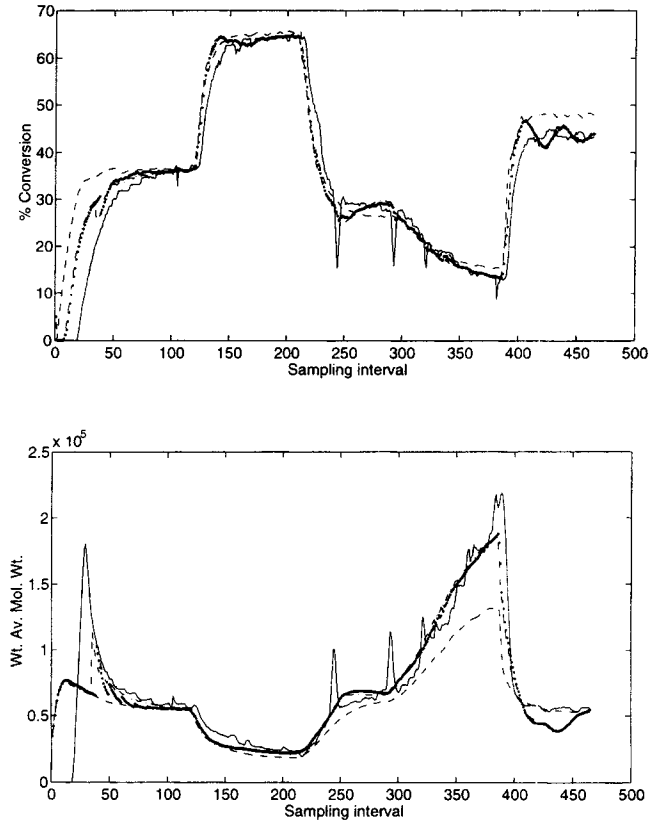


Figure 13. Measured values (—), single-rate measurement-based estimates with *SI* of 6 (- · -) and 10 (· · ·), and model predictions without the estimator (---) for *X* and $\overline{M}_w$ (run 3).

dictions without the estimator are shown in Figure 12. The estimator was turned on at about the 90th sampling interval (as the data until the 50th sampling interval were affected by the process problem described earlier). The estimator effectively eliminates the mismatch in X at sampling interval 120. Also, at the end of the run at sampling interval 270, the estimator is better able to track the steep increase in $\overline{M}_w$. This run illustrates that the estimator is able to improve the predictive capabilities of the model, by using the available information from the on-line measurements.

The estimator was implemented on the data from run 3 with *SI* values of 6 and 10. The performance of the estimators for these two smoothing intervals, along with model predictions without the estimator, are shown in Figure 13. The estimator was turned on at about the 35th sampling interval. Run 3 used all possible extrema of the reactor temperature and initiator flow rate. The performance of the estimator for both *SI* values of 6 and 10 is almost identical and shows far better predictions than the predictions without the estimator. The estimator minimizes the steady-state mismatch in X in the range of sampling intervals 180–210, 285–295, 370–385, and 450–465. Similarly, the estimator is able to track the $\overline{M}_w$ and follow the steep increase in the $\overline{M}_w$ in the range of sampling intervals 325–375. This run illustrates that the estimator is able to improve the prediction capabilities of the model, by using the available information from the on-line measurements. The almost identical performance comparison with *SI* of 6 and 10 makes the additional point that the performance of the estimator is insensitive to modest increases in *SI* above the suggested minimum value.

The removal of steady-state offset by the estimator has been achieved with the use of nonstationary states in the estimator. The trajectories of both nonstationary states with *SI* of 6 and 10 for run 3 are shown in Figure 14. For both values of *SI*, the nonstationary states show a similar trend. In the range of sampling intervals 285–295, the model predictions without the estimator underpredict the conversion data. An increase in $[R^{\cdot}]$ increases monomer conversion. During this interval, the estimator has increased the value of the nonstationary state (x_1^s) multiplicative with $[R^{\cdot}]$ to compensate for the mismatch and is able to track the measured X from the process. However, an increase in $[R^{\cdot}]$ also decreases $\overline{M}_w$. The second nonstationary state (x_2^s) multiplicative with μ_2 has therefore been increased by the estimator to offset the effect of x_1^s on $\overline{M}_w$ during the range of sampling intervals 285–295. A similar phenomenon is seen in the sampling interval range of 430–465, but in the opposite direction for both nonstationary variables.

Estimator performance with multirate measurements

In this case, the densitometer and viscometer measurements were made available to the estimator only once every five sampling intervals. These measurements still have the

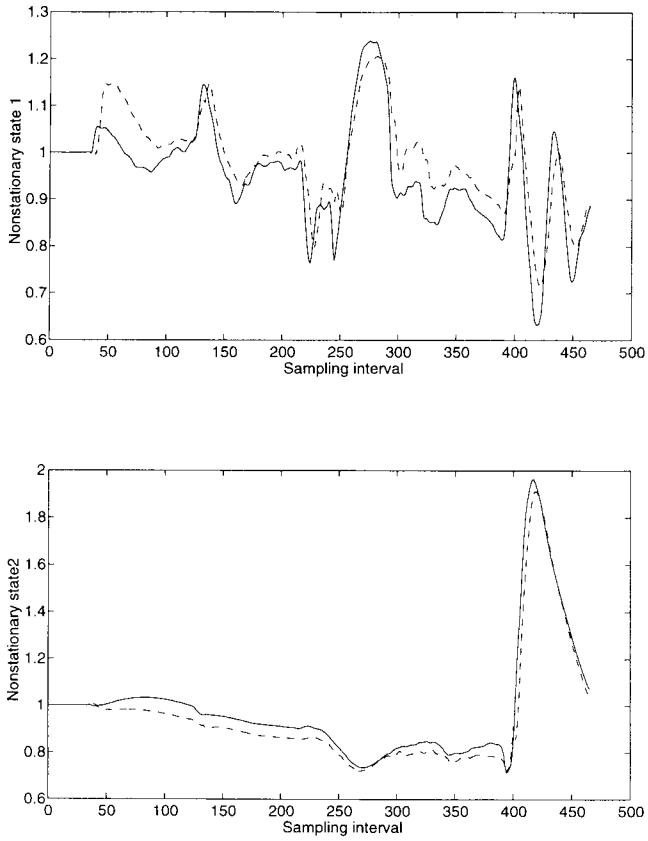


Figure 14. Nonstationary state trajectories for single-rate measurement-based estimator with *SI* of 6 (—) and 10 (---) for run 3.

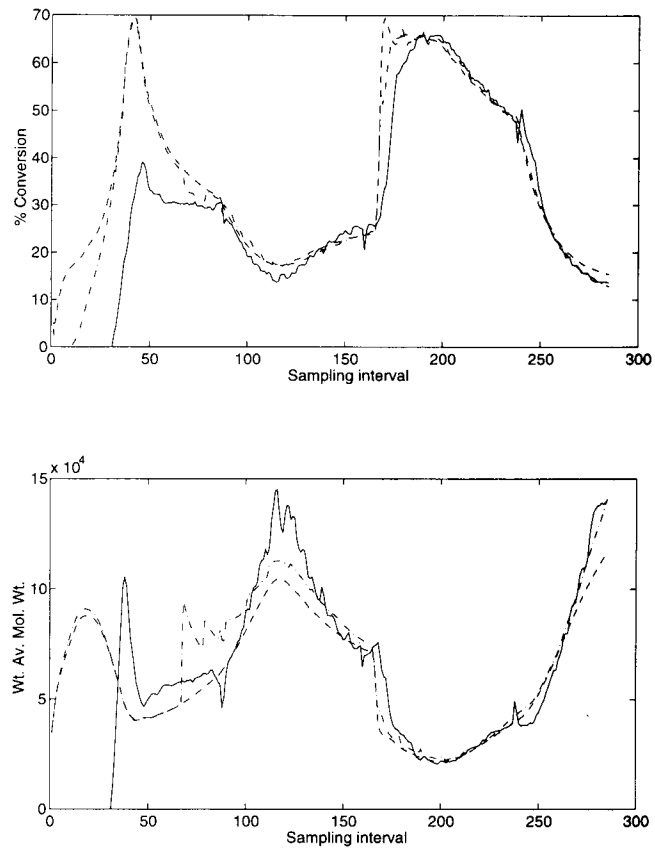


Figure 15. Measured values (—), multirate measurement-based estimates with *SI* = 10 (– · –), and model predictions without the estimator (---) for *X* and $\overline{M}_w$ (run 1).

measurement delay of five sampling intervals. Therefore, the slow rate measurements (density and viscosity) are taken once every five sampling intervals and are available with a delay of five sampling intervals. The measurement of the rate of heat released by the polymerization was available to the estimator at every sampling interval without any measurement delay.

The proposed estimator was implemented on the data from run 1 with an *SI* value of 10, which corresponds to the minimum suggested value. The performance of the estimator along with model predictions without the estimator are shown in Figure 15. The estimator was turned on at about the 70th sampling interval. The estimator tries to track the peak in $\overline{M}_w$ at about the 120th sampling interval. Also, at the end of the run at sampling interval 270, the estimator is better able to track the steep increase in $\overline{M}_w$ and to remove the bias in *X*. This run illustrates that the estimator can exhibit reasonable performance for the system even with slower measurement rates for density and viscosity.

The estimator was then implemented on the data from run 2 with an *SI* value of 10. The performance of the estimator along with model predictions without the estimator are shown in Figure 16. Note that this run was conducted in a manner such that the process outputs did not reach steady-state for most of the input changes. The estimator was turned on at about the 35th sampling interval. The estimator minimizes the steady-state mismatch in *X* in the range of sampling intervals 110–120 and 145–155. Similarly, the estimator is able to track the $\overline{M}_w$ and follow the steep increase in $\overline{M}_w$ in the range of sampling intervals 150–200 and 280–300. This run illustrates that the estimator exhibits reasonable performance for the system with slower measurement rates of density and viscosity, even when the input sequences are such that the process outputs are not allowed to reach their steady-state values.

The estimator performance was then evaluated on the data from run 3 with an *SI* value of 6, which is less than the recommended value of 10. The performance of the estimator along with model predictions without the estimator are shown in Figure 17. The estimator was turned on at about the 30th sampling interval. The performance of the estimator is generally poor when compared with the model predictions without the estimator. The estimator shows a large mismatch in predicting *X* in the range of sampling intervals 150–220 and 425–465. The performance in tracking $\overline{M}_w$ is reasonable, except for the mismatch toward the end of the run. Contrasting this performance to the single-rate measurements with an *SI* value of 6 in Figure 13, the performance of the estimator with single-rate measurements is seen to be much better.

The estimator was subsequently implemented on the data from run 3 with *SI* values of 10 and 15. The performance of the estimators for these two smoothing intervals, along with model predictions without the estimator, are shown in Figure 18. The estimator was turned on at about the 35th sampling

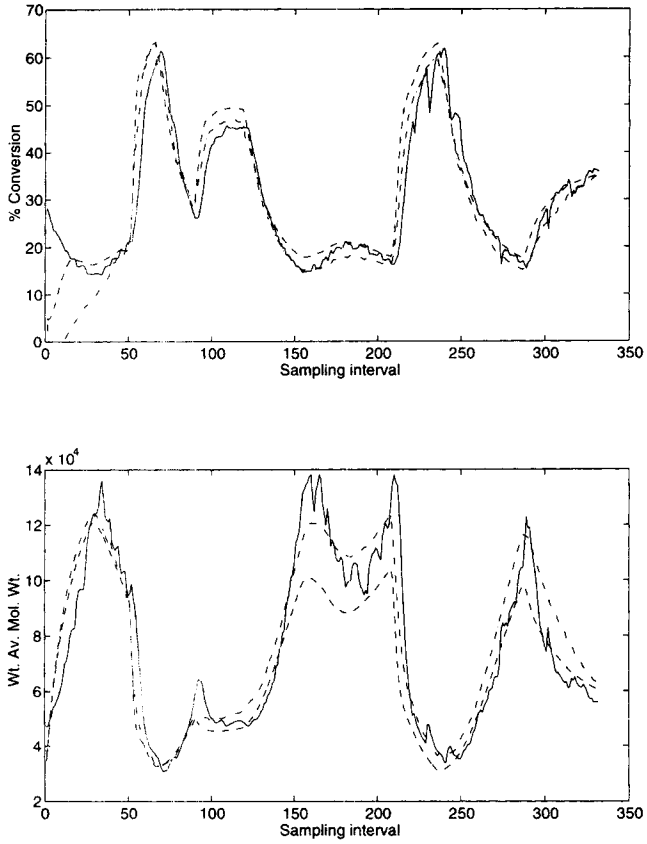


Figure 16. Measured values (—), multirate measurement-based estimates with $SI = 10$ (- · -), and model predictions without the estimator (---) for X and $\overline{M}_w$ (run 2).

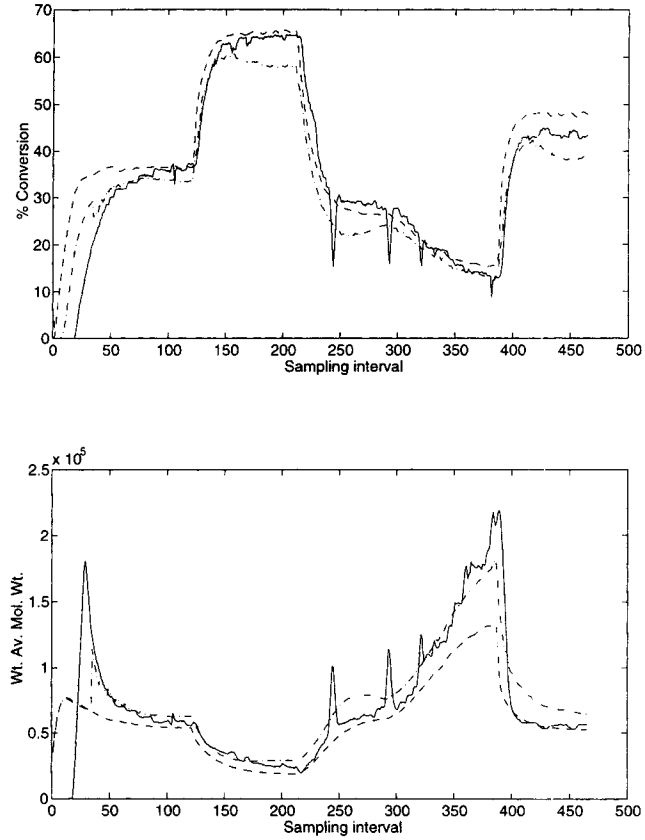


Figure 17. Measured values (—), multirate measurement-based estimates with $SI = 6$ (- · -), and model predictions without the estimator (---) for X and $\overline{M}_w$ (run 3).

interval. The performance of the estimator for both SI values of 10 and 15 is almost identical and exhibits far better predictions than the predictions without the estimator or for the case with an SI value of 6 in Figure 17. The estimator with both SI values minimizes the steady-state mismatch in X in the range of sampling intervals 285–295 and 450–465. Similarly, the estimator is able to track the $\overline{M}_w$ and also has good performance in tracking the steep increase in the $\overline{M}_w$ in the range of sampling intervals 325–375. However, when compared with the single-rate implementation in Figure 13, the response of the multirate measurement-based estimator is slower for X and $\overline{M}_w$. This can be expected, as information from the process is available less frequently for the multirate measurement-based estimator. The trajectories of both nonstationary states with SI values of 6, 10, and 15 for run 3, with multirate measurements, are shown in Figure 19. The trajectories of the nonstationary states for SI values of 10 and 15 are similar. However, the trajectories for the improper choice of $SI = 6$ are quite different. The steady-state mismatch in X and $\overline{M}_w$ with $SI = 6$ can be attributed to these differences.

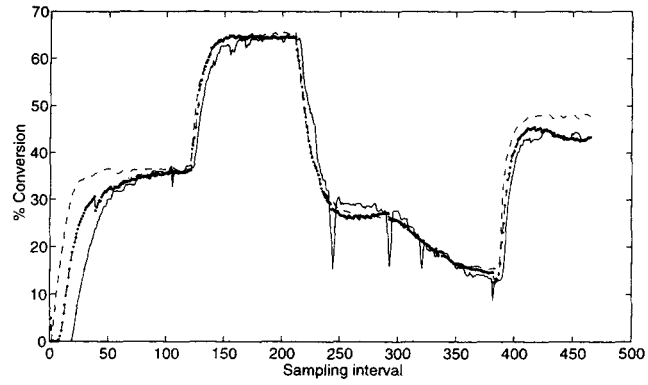
Experimental Evaluation of the Controller

The aim of the estimator is to make the model predictions track the actual process by updating process states and parameters using the available measurements. Once this is achieved, the estimator can predict variables that are not measured, and can predict the values of future output variables for use by a model-based predictive controller. Also, the estimator can predict the process states without delay at each sampling interval without having to wait until measurements are taken and are available. These are the key reasons for coupling the estimator with a controller. Use of just the nonlinear model without the estimator would result in biased predictions and could lead to poor control.

NLMPC for CNA systems

Model-predictive control algorithms use a model to predict the effect of past control inputs on the process output and then calculate the current and future control actions required to move the predicted process outputs along their desired setpoint trajectories. The CNA-NLMPC algorithm described by Mutha et al. (1997b) is based on the reinterpretation of the prediction equation used in linear MPC algorithms. For a SISO linear process, the prediction equation for p sampling intervals into the future can be written as

$$\hat{y}(k+p|k) = s_{p-1}\Delta u(k) + s_{p-2}\Delta u(k+1) + \cdots + s_{p-M}\Delta u(k+M-1) + y^{\text{past}}(k+p|k) + \hat{d}(k+p|k), \quad (16)$$



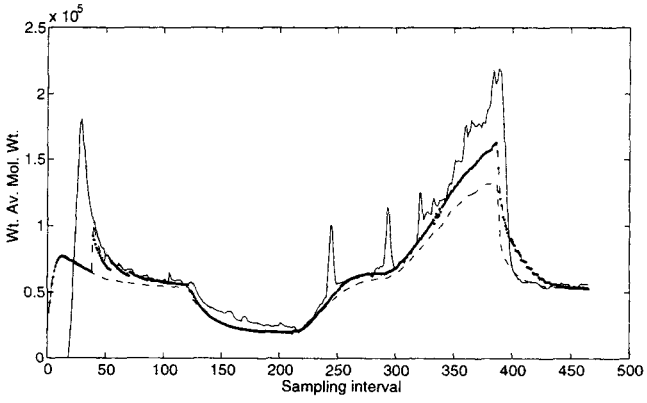
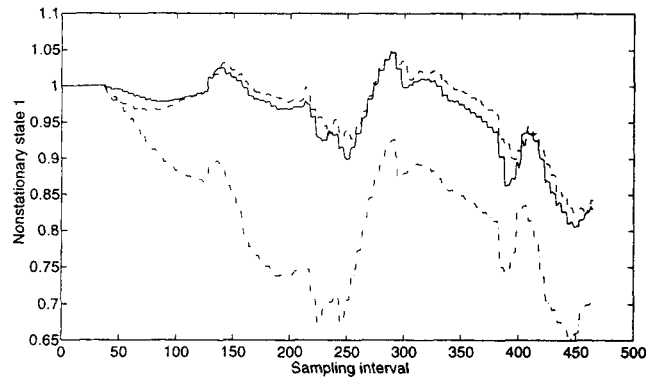


Figure 18. Measured values (—), multirate measurement-based estimates with *SI* of 10 (- · -) and 15 (· · ·), and model predictions without the estimator (---) for *X* and $\bar{M}_w$ (run 3).



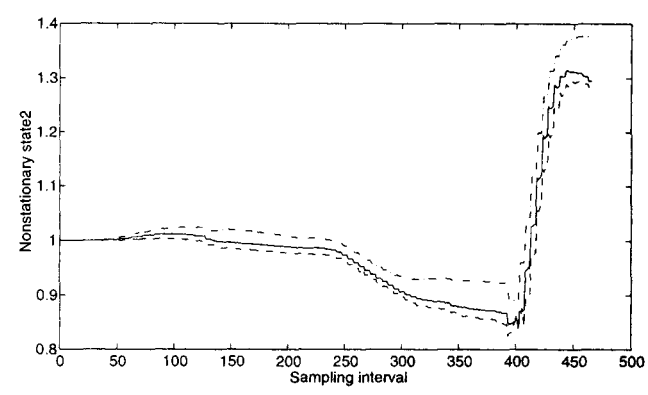


Figure 19. Nonstationary state trajectories for multirate measurement-based estimator with *SI* of 6 (- · -), 10 (—), and 15 (---) for run 3.

where Δ is the difference operator $(1 - z_d^{-1})$; p is the prediction horizon; M is the control horizon; $\hat{d}$ represents the predicted effect of unmeasured disturbances on the process output, and s_i denotes the value of the unit step response of the model at the ith sampling instant (with $s_i = 0\ \forall\ i < 0$). The output response of the process due to past control moves, y^{past}, can be calculated by finding the process response in the future with no further changes in the manipulated variable.

A reinterpretation of the prediction equation can be used to generate process output predictions that properly account for the system nonlinearities. For illustrative purposes, consider the prediction equation (Eq. 16) for a SISO static system (no dynamics) with both control and prediction horizons equal to one. Assuming that the operating point of the system is (y_0, u_0), this equation can be rewritten as

$$\hat{y} = \left.\frac{\partial y}{\partial(\Delta u)}\right|_{u_0} \Delta u + y_0 + \hat{d}, \tag{17}$$

where the step-response coefficient has been replaced by the partial derivative of the output with respect to Δu. For a linear system, the value of this partial derivative is a constant, and hence can be evaluated at any operating point. The preceding equation can be interpreted as a first-order Taylor series expansion of $\hat{y}$. The series expansion is truncated after the first term because higher-order derivatives are zero for a linear system.

Equation 17 provides the basis for extending the prediction equation for nonlinear systems. Consider a static, control nonaffine system. Let the operating point of the system be (y_0, u_0). The prediction equation for this nonlinear system is given by

$$\hat{y} = \left.\frac{\partial y}{\partial(\Delta u)}\right|_{u_0} \Delta u + \frac{1}{2!}\left.\frac{\partial^2 y}{\partial(\Delta u)^2}\right|_{u_0} (\Delta u)^2 + \cdots + y_0 + \hat{d}. \tag{18}$$

For a dynamic, control nonaffine system, the equation for predicting $\hat{y}(k+p|k)$, considering q terms in the Taylor series expansion and with a control horizon of $M(< p)$, will be

$$\hat{y}(k+p|k) = \sum_{i=1}^{q} \frac{1}{i!} \left.\frac{\partial^i y}{\partial[\Delta u(k)]^i}\right|_{\hat{y}(k+p|k),u_0} [\Delta u(k)]^i + \sum_{i=1}^{q} \frac{1}{i!} \left.\frac{\partial^i y}{\partial[\Delta u(k+1)]^i}\right|_{\hat{y}(k+p-1|k),u_0} [\Delta u(k+1)]^i + \cdots + r(k+p) + y^{\text{past}}(k+p|k) + \hat{d}(k+p|k), \tag{19}$$

where r is the contribution of the off-diagonal terms to the process output prediction. For a control horizon of $M = 1$, $\Delta u(k+i) = 0\ \forall\ i > 0$, and therefore $r(k+p) = 0$. For $M > 1$,

some of these contributions may be nonzero, but will be approximated with zero for the following reasons: (1) this algorithm makes use of the standard MPC receding horizon principle, and therefore only $\Delta u(k)$ is implemented at time k with $\Delta u(k+i)\ \forall\ i>1$ being discarded, and (2) y^{past} in the proposed implementation is based on a nonlinear model and therefore includes the effects of past implemented control moves. Setting $r(k+p)=0$, the preceding equation can be rewritten as

$$\hat{y}(k+p|k) = \sum_{j=0}^{M-1}\sum_{i=1}^{q}\frac{1}{i!}\frac{\partial^i y}{\partial[\Delta u(k+j)]^i}\bigg|_{\hat{y}(k+p-j|k),u_0}[\Delta u(k+j)]^i + y^{\text{past}}(k+p|k)+\hat{d}(k+p|k) \tag{20}$$

$$= \sum_{j=0}^{M-1}\sum_{i=1}^{q}\frac{s^{(i)}_{p-j-1}(k)}{i!}\bigg|_{u_0}[\Delta u(k+j)]^i + y^{\text{past}}(k+p|k)+\hat{d}(k+p|k), \tag{21}$$

where

$$s^{(i)}_j(k)=s^{(i)}_j=\begin{cases}0 & \text{if } j<0\\ \dfrac{\partial^i y}{\partial[\Delta u(k+j)]^i}\bigg|_{\hat{y}(k+j+1|k)} & \text{if } j\geq 0.\end{cases} \tag{22}$$

Equation 21 can be written in a more compact matrix form for the whole prediction horizon as

$$\hat{Y}=S^{(1)}|_{u_0}\Delta U+\frac{S^{(2)}|_{u_0}}{2!}(\Delta U)^2+\cdots+Y^{\text{past}}+\hat{D} = \sum_{i=1}^{q}\frac{S^{(i)}|_{u_0}}{i!}(\Delta U)^i+Y^{\text{past}}+\hat{D}, \tag{23}$$

where

$$\hat{Y}=[\hat{y}(k+p_1|k)\cdots\hat{y}(k+p_2|k)]^T$$

$$S^{(i)}=\begin{bmatrix} s^{(i)}_{p_1-1} & s^{(i)}_{p_1-2} & \cdots & s^{(i)}_{p_1-M}\\ s^{(i)}_{p_1} & s^{(i)}_{p_1-1} & \cdots & s^{(i)}_{p_1-M+1}\\ \vdots & & \ddots & \vdots\\ s^{(i)}_{p_2-1} & s^{(i)}_{p_2-2} & \cdots & s^{(i)}_{p_2-M}\end{bmatrix}$$

$$(\Delta U)^i=\{[\Delta u(k)]^i[\Delta u(k+1)]^i\cdots[\Delta u(k+M-1)]^i\}^T$$

$$Y^{\text{past}}=[y^{\text{past}}(k+p_1|k)\cdots y^{\text{past}}(k+p_2|k)]^T$$

$$\hat{D}=[\hat{d}(k+p_1|k)\cdots\hat{d}(k+p_2|k)]^T$$

with p_1 and p_2 denoting the lower and upper prediction horizons. The $S^{(1)}$ matrix in the preceding equation corresponds to the standard step-response matrix used in linear MPC. For an m-input, n-output MIMO system, assuming that $\partial y^2/(\partial\Delta u_i\partial\Delta u_j)=0$ for any two inputs, $i\neq j$, y is a vector of dimension n, Δu is a vector of dimension m, and each element of the matrix $S^{(i)}$ is of dimension $n\times m$.

In Mutha et al. (1997b), the prediction Eq. 23 is rearranged as follows:

$$\hat{Y}=\bar{S}\Delta U+Y^{\text{past}}+\hat{D}, \tag{24}$$

where

$$\bar{S}=\left(S^{(1)}+\sum_{i=2}^{q}\frac{S^{(i)}}{i!}\text{diag}(\Delta U)^{i-1}\right)\bigg|_{u_0} \tag{25}$$

has been formed by factoring ΔU out of the first term on the righthand side of Eq. 23 and diag $(\Delta U)^{i-1}$ denotes a diagonal matrix consisting of the elements of vector $(\Delta U)^{i-1}$. Prediction Eq. 24 now has exactly the same structure as the prediction equation used in a standard LMPC algorithm. Therefore, the CNA-NLMPC algorithm reduces to an optimization problem with a quadratic cost function, subject to linear constraints. Further implementation details of the suboptimal CNA-NLMPC can be found in Mutha et al. (1997b).

Multivariable controller performance

During the closed-loop experiment, the proposed estimator was implemented on-line using single-rate measurements and an *SI* value of 10. The estimator states selected for EKF1 and EKF2 and their tuning parameters are shown in the Appendix and differ slightly from the states used for the estimator evaluation. The tuning matrices used for the estimator evaluation and the controller assessment are also not identical. The reason for these differences is that only preliminary tuning of the estimator was performed prior to the closed-loop experiment. More extensive off-line tuning of the estimator was carried out after these experiments and these are the results described previously in the third section.

The control objective was to make X and $\overline{M}_w$ track the requested setpoints. The manipulated variables were the reactor temperature and flow rate of the initiator solution to the reactor and the controlled variables were the predicted values $\hat{X}$ and $\hat{\overline{M}}_w$. In this case, the measurements (ρ, η, H) were only used to update the states that were then used to update $\hat{X}$ and $\hat{\overline{M}}_w$, that is, the measurement-based inferred values of X and $\overline{M}_w$ were not used by the controller. The performance of the estimator during the entire closed-loop experiment is shown in Figure 20. The estimator was turned on at about the 25th sampling interval.

Both the CNA-NLMPC algorithm proposed by Mutha et al. (1997b) and a standard CA-NLMPC (e.g., Garcia, 1984; Gangadhar and Zafiriou, 1992) algorithm were evaluated during this closed-loop experiment. To allow a fair comparison, both algorithms had identical settings in all respects, except for the use of $S^{(2)}$ by the CNA-NLMPC algorithm (see Eq. 23, with $q=2$). This allowed for the effect of including $S^{(2)}$ to be isolated and distinguished from the effect of the various controller tuning parameters. Both NLMPC algorithms are based on the full nonlinear model, that is, the y^{past} and $S^{(1)}$ for both algorithms have been calculated from the nonlinear model. The step-response matrix $S^{(1)}$ at each sam-

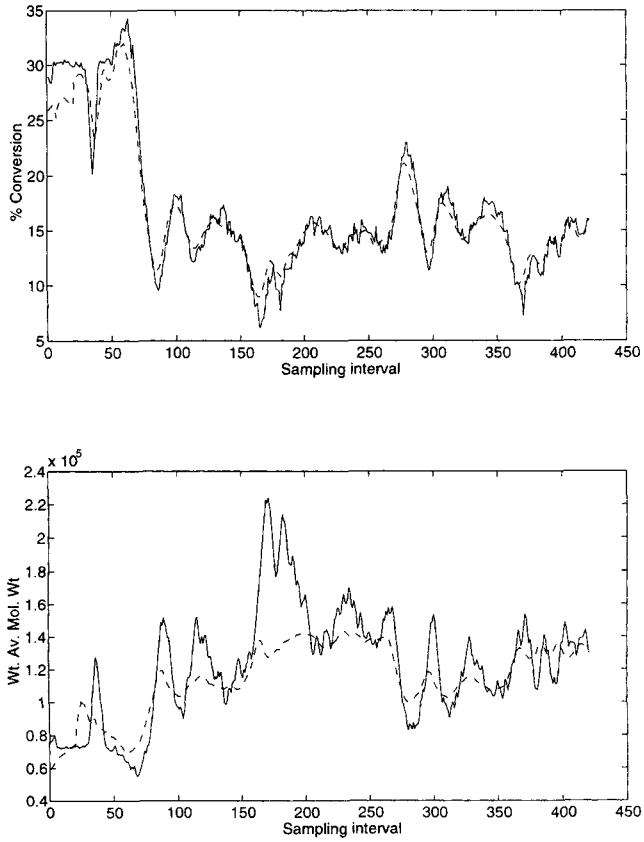


Figure 20. Measured values (—) and single-rate measurement-based estimates with $SI = 10$ (---) for X and $\overline{M}_w$ during closed-loop experiment.

pling interval was calculated by the method described in Peterson et al. (1992). The $S^{(2)}$ matrix at each sampling interval was determined using a forward difference type of approximation (see Mutha et al., 1997b, Eq. 24). Rate constraints were imposed on the reactor temperature with a maximum allowable change of 5°C per sampling interval. Greater changes in reactor temperature per sampling interval may not be achievable on this pilot-plant. Minimum and maximum temperature constraints were set at 50 and 100°C, respectively. Minimum and maximum flow rates of initiator were set at $2.8(10^{-5})$ and $2.4(10^{-3})$ moles/min, respectively. The controller tuning parameters used were $p_1 = 1$, $p_2 = 12$, $M = 1$, $\gamma_1 = 2.5$ (output weight on X), $\gamma_2 = 1$ (output weight on $\overline{M}_w$), $\lambda_1 = 1$ (input weight on initiator flow rate scaled by 10,000), and $\lambda_2 = 5$ (input weight on reactor temperature). Both control algorithms were under regulatory control for $\hat{X}$, at a setpoint value of 15% conversion and under servocontrol for $\hat{\overline{M}}_w$.

The performance of both NLMPCs in tracking the setpoints is shown in Figure 21, and the corresponding manipulated variable trajectories are shown in Figure 22. The CNA-NLMPC algorithm was started at the 57th sampling interval. The control algorithm was switched from CNA-NLMPC to CA-NLMPC at the 258th sampling interval. The switching of the algorithms was performed on-line, by changing the value of a "flag" from 1 to 0. The system was under CNA-NLMPC between sampling intervals 57 and 257 with $\hat{\overline{M}}_w$ setpoints of $1.1(10^5)$ from 57 to 145 sampling intervals and $1.4(10^5)$ from 146 to 257 sampling intervals. The system was under CA-NLMPC between sampling intervals 258 and 419 with $\hat{\overline{M}}_w$ setpoints of $1.1(10^5)$ from 258 to 349 sampling intervals and $1.4(10^5)$ from 349 to 419 sampling intervals. From Figure 21, it is clearly seen that the performance of CNA-NLMPC in tracking the $\hat{\overline{M}}_w$ setpoint is far superior to CA-NLMPC. CA-NLMPC was not able to reach the setpoint of $1.4(10^5)$, even after 70 sampling intervals (2 h and 20 min). By comparison, CNA-NLMPC was able to reach the same setpoint in less than 50 sampling intervals. A comparison of the time-normalized weighted sum of absolute error (SAE) in $\hat{X}$ and $\hat{\overline{M}}_w$ was conducted for both algorithms at both setpoint changes in $\hat{\overline{M}}_w$. For the $\hat{\overline{M}}_w$ setpoint change to $1.4(10^5)$, the SAE using CNA-NLMPC is 1.77%, and with CA-NLMPC is 1.73% for $\hat{X}$. For the same setpoint change, the SAE with CNA-NLMPC is $5.9(10^3)$ and with CA-NLMPC is $11.4(10^3)$ for $\hat{\overline{M}}_w$. Although the SAE for CA-NLMPC is marginally better than CNA-NLMPC for $\hat{X}$, it is much worse for $\hat{\overline{M}}_w$. The SAE for the $\hat{\overline{M}}_w$ setpoint of $1.1(10^5)$ was not straightforward, as the initial conditions for CA-NLMPC and CNA-NLMPC were completely different. From Figure 21, it can be qualitatively inferred that the magnitudes of the oscillations are higher for $\hat{X}$ and $\hat{\overline{M}}_w$ for CA-NLMPC than for CNA-NLMPC. If the first 20 sampling intervals after the step change in setpoints

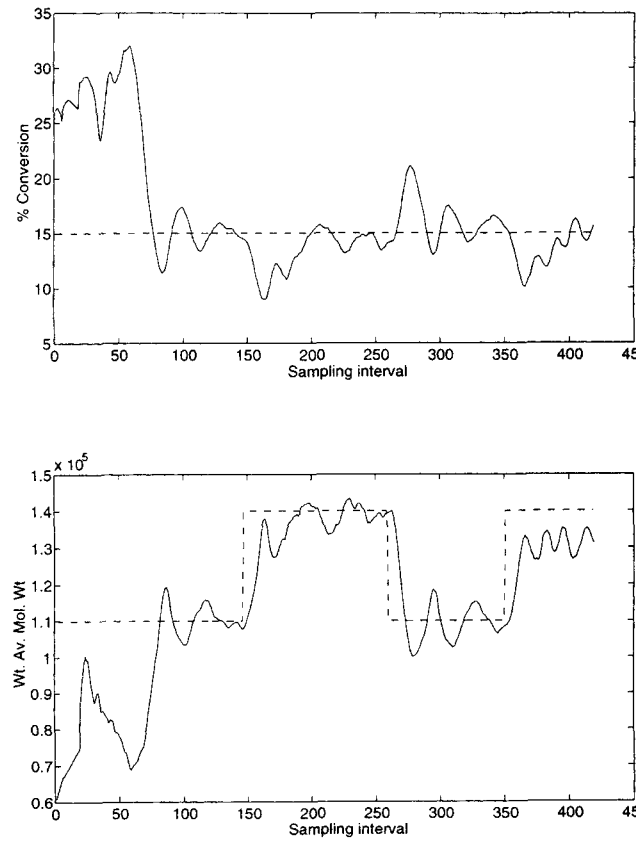


Figure 21. Controlled variables ($\hat{X}$ and $\hat{\overline{M}}_w$) during closed-loop experiment (CNA-NLMPC from 57 to 257 sampling intervals; CA-NLMPC from 258 to 419 sampling intervals); setpoints shown by (---).

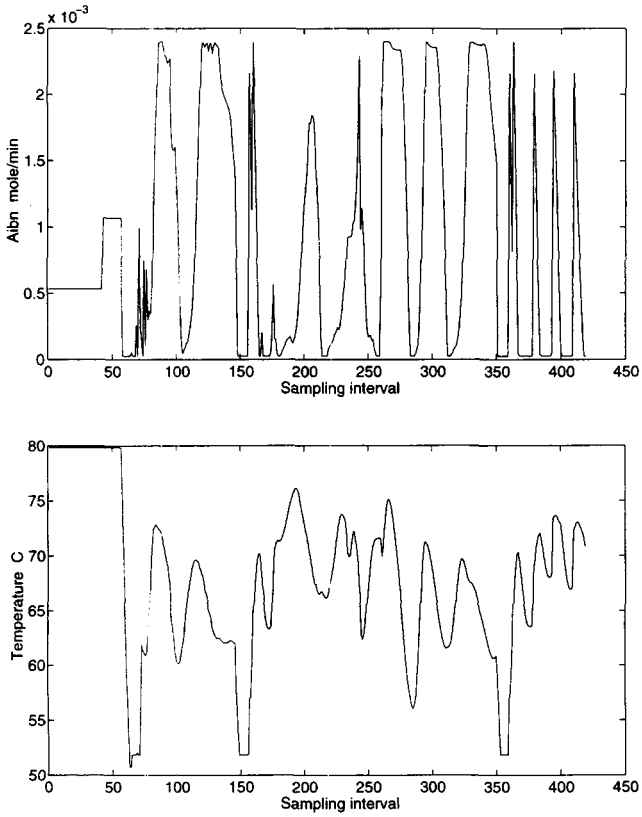


Figure 22. Manipulated variables during closed-loop experiment (CNA-NLMPC from 57 to 257 sampling intervals; CA-NLMPC from 258 to 419 sampling intervals).

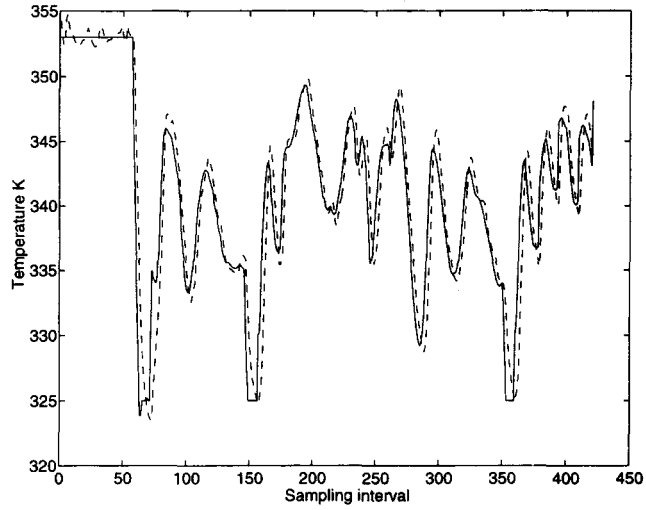


Figure 23. Temperature setpoint (—) to the cascade temperature controller and the actual reactor temperature (---) during closed-loop experiment.

are ignored for both NLMPCs, the SAE for $\hat{X}$ with CNA-NLMPC is 1.19% and with CA-NLMPC is 1.38%, whereas SAE for $\hat{M}_w$ with CNA-NLMPC is $3.19(10^3)$ and with CA-NLMPC is $4.06(10^3)$. The SAE is lower for CNA-NLMPC for both outputs.

Although SAE is adequate in quantifying the performance of the controllers, other methods like the sum of squares of errors (SSE) can also be used instead.

The manipulated variable trajectories are shown in Figure 22 for the CNA-NLMPC and CA-NLMPC and are seen to be quite different. The initiator flow-rate trajectory for the CNA-NLMPC (sampling intervals 57 to 257) are clearly nonperiodic. The initiator flow-rate trajectory for the CA-NLMPC (sampling intervals 258 to 419) shows some amount of periodicity. This kind of periodicity is typically seen when the controller is not able to properly decouple a MIMO system. The CA-NLMPC algorithm decouples this system through the use of $S^{(1)}$, and the CNA-NLMPC does so through the use of $\bar{s}$ (Eq. 25).

Reactor temperature controller performance

Reactor temperature is a manipulated variable with respect to the multivariable controller described in the previous subsection. The effective implementation of the calculated value of this manipulated variable is critical to the performance of the overall control strategy. In this closed-loop experiment, the desired value of reactor temperature as determined by the NLMPC was passed as a setpoint to the cascade temperature controller, as shown in Figure 4. This setpoint and the actual reactor temperature are shown in Figure 23, where it can be seen that the cascade controller was able to provide close tracking of the temperature setpoints generated by both NLMPC algorithms. The use of the rate constraints in the NLMPC algorithms was helpful in generating feasible setpoint changes for this system.

Concluding Remarks

This article presents an experimental evaluation of a nonlinear model-based estimator and predictive control strategy on a continuous MMA solution polymerization reactor. The proposed estimator was successfully implemented for both single-rate and multirate measurements. The computation time required for implementing the estimator algorithm was on the order of 3 s per sampling interval on a 486 DX Intel CPU with a 33-MHz clock frequency. From the performance of the estimator for both single-rate and multirate measurements, it can be concluded that the key parameter of the estimation algorithm, the smoothing interval, *SI*, should be chosen at least equal to the largest of the sum of the measurement delay and the number of sampling intervals between consecutive slow measurements, both expressed in sampling intervals, for each slow measurement considered separately. A much smaller *SI* value leads to poor performance of the estimator, and increasing *SI* beyond the previously suggested value does not necessarily improve the performance of the estimator. The control experimental run showed the improved performance that can be obtained by properly accounting for the control nonaffine nature of this system in the prediction equation. The computation time was on the order of 5 to 7 s per sampling interval for implementing the nonlinar control algorithm on the same 486-based computer. Most of the computation time was spent integrating the model equations for calculating the step-response coefficient matrices.

Literature Cited

Chien, D. C. H., "On-line Control of Conversion in Continuous Solution Methyl Methacrylate Polymerization Reactors," PhD Thesis, Dept. of Chemical Engineering, Univ. of Waterloo, Waterloo, Ont., Canada (1992).

Chien, D. C. H., and A. Penlidis, "Effect of Impurities on Continuous Solution Methyl Methacrylate Polymerization Reactors: I. Open-loop Process Identification Results," *Poly. React. Eng. J.*, **2**, 163 (1994a).

Chien, D. C. H., and A. Penlidis, "Effect of Impurities on Continuous Solution Methyl Methacrylate Polymerization Reactors: II. Closed-loop Real-time," *Chem. Eng. Sci.*, **49**, 1855 (1994b).

Dimitratos, J., C. Georgakis, M. S. El-Aasser, and A. Klein, "An Experimental Study of Adaptive Kalman Filtering in Emulsion Copolymerization," *Chem. Eng. Sci.*, **46**, 3203 (1991).

Dimitratos, J., G. Elicabe, and C. Georgakis, "Control of Emulsion Polymerization Reactors," *AIChE J.*, **40**, 1993 (1994).

Elicabe, G. E., E. Ozdeger, C. Georgakis, and C. Cordeiro, "On-line Estimation of Reaction Rates in Semicontinuous Reactors," *Ind. Eng. Chem. Res.*, **34**, 1219 (1995).

Elicabe, G. E., and G. R. Meira, "Estimation and Control in Polymerization Reactors," *Polym. Eng. Sci.*, **28**, 121 (1988).

Ellis, M. F., T. W. Taylor, V. Gonzalez, and K. F. Jensen, "On-Line Molecular Weight Distribution Estimation and Control in Batch Polymerization," *AIChE J.*, **40**, 445 (1994).

Gangadhar, G., and E. Zafiriou, "Nonlinear Quadratic Dynamic Matrix Control with State Estimation," *Ind. Eng. Chem. Res.*, **31**, 1096 (1992).

Gao, J., and A. Penlidis, "A Comprehensive Simulation Package/Database for Reviewing Free-radical Homopolymerization," *J. Macromol. Sci.-Rev. Macrmol. Chem. Phys.*, Vol. C 36, p. 199 (1996).

Garcia, C. E., "Quadratic Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process," AIChE Meeting, San Francisco (1984).

Jo, J. H., and S. G. Bankhoff, "Digital Monitoring and Estimation of Polymerization Reactors," *AIChE J.*, **22**, 361 (1976).

Kozub, D. J., and J. F. MacGregor, "State Estimation for Semi-batch Polymerization Reactors," *Chem. Eng. Sci.*, **47**, 1047 (1992).

Lawrence, A. D., "Multivariable Identification for the Control of a Continuous Solution Methyl Methacrylate Polymerization Reactor," Master's Thesis, Dept. of Chemical Engineering, Univ. of Waterloo, Waterloo, Ont., Canada (1994).

MacGregor, J. F., A. Penlidis, and A. E. Hamielec, "Control of Polymerization Reactors: A Review," *Poly. Process Eng.*, **2**, 179 (1984).

Mutha, R. K., W. R. Cluett, and A. Penlidis, "A New Multirate Measurement Based Estimator: Emulsion Copolymerization Batch Reactor Case Study," *Ind. Eng. Chem. Res.*, **36**, 1036 (1997a).

Mutha, R. K., W. R. Cluett, and A. Penlidis, "Nonlinear Model-based Predictive Control of Control Nonaffine Systems," *Automatica*, **33**, 907 (1997b).

Ogunnaike, B. A., and P. C. Gopalratnam, "A Two-Tier Multirate Control System Structure for an Industrial Terpolymerization Reactor," AIChE Meeting, Los Angeles (1991).

Ogunye, A. B., "On-line Multivariable Control of Continuous Solution Polymerization Reactors," PhD Thesis, Dept. of Chemical Engineering, Univ. of Waterloo, Waterloo, Ont., Canada (1994).

Parrish, J. R., and C. B. Brosilow, "Nonlinear Inferential Control," *AIChE J.*, **34**, 633 (1988).

Peterson, T., E. Hernandez, Y. Arkun and F. J. Schork, "A Nonlinear DMC Algorithm and Its Application to a Semibatch Polymerization Reactor," *Chem. Eng. Sci.*, **47**, 737 (1992).

Ray, W. H., "Polymerization Reactor Control," *Proc. ACC*, Boston, p. 842 (1985).

Soroush, M., and C. Kravaris, "Nonlinear Control of a Batch Polymerization Reactor: An Experimental Study," *AIChE J.*, **38**, 1429 (1992).

Soroush, M., and C. Kravaris, "Multivariable Nonlinear Control of a Continuous Polymerization Reactor: An Experimental Study," *AIChE J.*, **39**, 1920 (1993).

Soroush, M., and C. Kravaris, "Nonlinear Control of a Polymerization CSTR with Singular Characteristic Matrix," *AIChE J.*, **40**, 980 (1994).

Stephanopoulos, G., and K. Y. San, "Studies on On-line Bioreactor Identification," *Biotech. Bioeng.*, **26**, 1176 (1984).

Tzouanas, V. K., and S. L. Shah, "On-Line Monitoring of Polymer Quality in a Batch Polymer Reactor," *Chem. Eng. Sci.*, **44**, 183 (1989).

Urretabizkaia, A., J. R. Leiza, and J. M. Asua, "On-line Terpolymer Composition Control in Semicontinuous Emulsion Polymerization," *AIChE J.*, **40**, 1850 (1994).

Van Doremaele, G. H. J., H. A. S. SchoonBrood, J. Kurja, and A. L. German, "Copolymer Composition Control by Means of Semicontinuous Emulsion Copolymerization," *J. Appl. Polym. Sci.*, **45**, 957 (1992).

Appendix: Estimator Tuning Parameters

The tables show the diagonal elements of the respective tuning parameter matrices. All of the off-diagonal elements were intialized with a value of zero.

Measurement variances (R)

For all estimator and control runs:

Measurement	Variance
Rate of heat liberated (cal/min)	20
Percent conversion	1.0×10^{-4}
Viscosity (cp)	0.001

Estimator evaluation

EKF1

State	Initial Variance (P)	State Variance (Q)
$V_R[P]$	1.0×10^{-3}	1.0×10^{-6}
$V_R\mu_2$	1.0×10^{-1}	5.0×10^{-3}
x_1^s	5.0×10^{-7}	5.0×10^{-9}
x_2^s	5.0×10^{-6}	5.0×10^{-7}

EKF2

State	Initial Variance (P_2)	State Variance (Q_2)
$V_R[P]$	1.0×10^{-1}	1.0×10^{-7}
$V_R\mu_2$	1.0×10^{-1}	1.0×10^{-5}

Controller evaluation

EKF1

State	Initial Variance (P)	State Variance (Q)
$V_R[P]$	1.0×10^{-3}	1.0×10^{-6}
$V_R\mu_2$	1.0×10^{-1}	5.0×10^{-3}
x_1^s	5.0×10^{-7}	1.0×10^{-10}
x_2^s	5.0×10^{-6}	1.0×10^{-8}

EKF2

State	Initial Variance (P_2)	State Variance (Q_2)
$V_R[P]$	1.0×10^{-1}	1.0×10^{-7}
$V_R\mu_2$	1.0×10^{-1}	1.0×10^{-5}
x_1^s	1.0×10^{-6}	1.0×10^{-7}

Manuscript received Feb. 5, 1997, and revision received July 11, 1997.