ELSEVIER

J. Proc. Cont. Vol. 7, No. 1, pp. 3–17, 1997

0959-1524/97 $17.00 + 0.00

PII: S0959-1524(96)00006-6

Full Papers

Input–output linearizing control of constrained nonlinear processes

Michael J. Kurtz and Michael A. Henson*

Department of Chemical Engineering, Louisiana State University, Baton Rouge, LA 70803-7303, USA

Received 25 September 1995; revised 14 February 1996

An input–output linearization strategy for constrained nonlinear processes is proposed. The system may have constraints on both the manipulated input and the controlled output. The nonlinear control system is comprised of: (i) an input–output linearizing controller that compensates for processes nonlinearities; (ii) a constraint mapping algorithm that transforms the original input constraints into constraints on the manipulated input of the feedback linearized system; (iii) a linear model predictive controller that regulates the resulting constrained linear system; and (iv) a disturbance model that ensures offset-free setpoint tracking. As a result of these features, the approach combines the computational simplicity of input–output linearization and the constraint handling capability of model predictive control. Simulation results for a continuous stirred tank reactor demonstrate the superior performance of the proposed strategy as compared to conventional input–output linearizing control and model predictive control techniques. Copyright 1996 Elsevier Science Ltd

Keywords: nonlinear processes; constrained control; feedback linearization; model predictive control; chemical reactor control

Many chemical processes require high performance control strategies due to increased demands on product quality, productivity, and environmental impact. However, most controller design techniques are based on linear models that do not account for process nonlinearities. For strongly nonlinear processes, significantly improved performance can be expected if a more accurate nonlinear model is employed directly in the controller design. In the past decade, a variety of nonlinear controller design strategies have been proposed. Nonlinear model predictive control and input–output linearizing control have emerged as the most widely studied design techniques for chemical process applications.

Nonlinear model predictive control (NMPC) is an optimization-based strategy in which a nonlinear process model is used to predict the effect of future manipulated input moves on future values of the controlled outputs[1–4]. At each time step, a sequence of input moves is calculated by solving an open-loop optimal control problem. A feedback controller is obtained by implementing only the first calculated input and resolving the optimization problem at the next sampling instant using new process measurements. NMPC offers many of the appealing features of linear model predictive control, including nominal stability in the presence of input and output constraints[5–7].

On the other hand, input–output linearizing control (IOLC) is an analytical design approach which aims to reduce the original nonlinear control problem to a simpler linear control problem[8–10]. The nonlinear control system is designed using a two-step procedure. First, a nonlinear process model is used to synthesize a nonlinear state feedback controller that linearizes the map between a 'new' manipulated input and the controlled output. In the second step, a linear pole placement controller is designed for the feedback linearized system. As compared to NMPC, IOLC offers several important advantages including a well defined setpoint response, transparent controller tuning, and low computational requirements[8]. However, conventional feedback linearization techniques do not have constraint handling capabilities[11,12]. As a result, linearizing controllers often are tuned to avoid input constraints, thereby yielding unnecessarily poor performance[13].

In this paper, an input–output linearization strategy for constrained nonlinear processes is presented. The linearizing controller is designed in the usual manner by neglecting constraints on input and output variables. At each sampling instant, the IOLC law and the current state measurement are used to map the original input constraints into constraints on the manipulated input of the feedback linearized system. This transformation

*To whom correspondence should be addressed: henson@onlc.che.lsu.edu

yields a *linear* dynamic system with constant output constraints and *time-varying* input constraints. The constraints are handled explicitly by designing a linear model predictive controller for the constrained linear system. Thus, the proposed strategy combines the benefits of the feedback linearization and model predictive control approaches.

The remainder of the paper is organized as follows. First, feedback linearizing control strategies for constrained nonlinear systems are reviewed. Then, the deleterious effects of process constraints on conventional IOLC controllers are demonstrated using a simple chemical reactor model. The input–output linearization strategy for constrained systems is then presented. The reactor model is then used to compare the proposed strategy with conventional IOLC and model predictive control techniques. Finally, a summary and conclusions are presented.

Feedback linearization strategies for constrained systems

Many processes exhibit significant nonlinear behavior and are subject to constraints on input and/or output variables. Consequently, controller design for constrained nonlinear systems is a problem of considerable theoretical and practical importance. Recent research has focused on feedback stabilization of constrained linear systems; a variety of controller design and analysis tools are now available[14,15]. However, few constrained control techniques have been proposed for nonlinear systems. Most of the available design methods are based on feedback linearization and only address input constraints.

Kapoor and Daoutidis[16] present a controller design strategy for unstable nonlinear systems that are state-space linearizable. The method is based on the construction of invariant sets in which closed-loop stability is guaranteed. Kendi and Doyle[17] propose a nonlinear anti-windup technique for constrained multivariable systems that can be input–output decoupled. The controller design utilizes an anti-windup scheme developed for constrained linear systems. An alternative anti-windup strategy is proposed by Soroush and Kravaris[18]. Calvet and Arkun[19] present a state-space linearization technique based on the internal model control structure. Possible mismatch between the process and model outputs is addressed by mapping the actual input constraint into a time-varying constraint on the input of the feedback linearized system.

Lee and Hedrick[20] present an input–output linearization technique based on minimizing an objective function that penalizes excessive manipulated input moves. An adaptive scheme in which the tuning parameters of the linearizing controller are adjusted on-line such that input constraints are avoided is proposed by Zhou *et al.*[21] Belchen and Sandrib[22] present a linearization strategy in which less important tracking objectives are sacrificed when inputs become saturated. The major disadvantage of these constrained control techniques is that they only provide indirect compensation for input constraints and cannot address output constraints whatsoever.

An alternative strategy is to map the input constraints into corresponding constraints on the feedback linearized system and then design a model predictive controller for the constrained linear system. This approach has been pursued independently by Nevistic and co-workers[23,24] and ourselves[25]. We believe that the control strategy presented in this paper provides several important advantages over the method developed by Nevistic and co-workers. In particular, the proposed technique offers:

1. A novel constraint mapping procedure that is computationally efficient and effective.
2. Explicit handling of output constraints.
3. A systematic predictive controller design strategy for the unstable, constrained linear system that results from input–output linearization.
4. A novel disturbance modeling technique that ensures offset-free setpoint tracking.

Input–output linearization for unconstrained systems

Controller design

We briefly outline the IOLC design procedure for unconstrained systems; more detailed descriptions are available elsewhere[8–10]. The nonlinear process model has the form,

$$\dot{x} = f(x) + g(x)u \qquad (1)$$
$$y = h(x)$$

where x is an n-dimensional vector of state variables, and u and y are the manipulated input and controlled output, respectively. We assume that the state vector is measured or estimated from available measurements. The Lie derivative of the scalar field $h(x)$ with respect to the vector field $f(x)$ is defined as:

$$L_f h(x) = \frac{\partial h(x)}{\partial x} f(x) \qquad (2)$$

Higher-order Lie derivatives are defined recursively:

$$L_f^k h(x) = \frac{\partial L_f^{k-1} h(x)}{\partial x} f(x) \qquad (3)$$

The nonlinear system (1) has relative degree r at the point x_0 if:

1. $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x_0 and for all $k < r - 1$.
2. $L_g L_f^{r-1} h(x_0) \neq 0$.

We assume that the relative degree is well defined throughout the region of operation.

The nonlinear state feedback control law that provides input–output linearization can be written as

$$u = \frac{v - L_f^r h(x)}{L_g L_f^{r-1} h(x)} \tag{4}$$

where v is the manipulated input for the feedback linearized system. Under this feedback law, there exists a nonlinear coordinate transformation $[\xi^T, \eta^T]^T = \Phi(x)$ such that (1) can be represented as a partially linear system of the form,

$$\begin{aligned} \dot{\xi} &= A\xi + Bv \\ \dot{\eta} &= q(\xi,\eta) \\ y &= C\xi \end{aligned} \tag{5}$$

where the triplet (A,B,C) is in Brunovsky canonical form:

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad C = [1\ 0 \dots 0\ 0] \tag{6}$$

We assume that the $(n-r)$-dimensional nonlinear subsystem (the *zero dynamics*) in (5) is bounded-input, bounded-output stable with respect to the ξ variables as inputs. This assumption ensures that the nonlinear system (1) is stabilized if the r-dimensional linear subsystem in (5) is stabilized.

In the absence of constraints, the linear subsystem can be stabilized using a pole-placement controller with integral action,

$$v = \left[\alpha_0 \ -\alpha_1 \dots -\alpha_r\right] \begin{bmatrix} z \\ \xi \end{bmatrix} \tag{7}$$

$$\dot{z} = y_{sp} - y$$

where z is the integral state, y_{sp} is the setpoint, and the α_k are controller tuning parameters chosen such that the polynomial $s^{r-1} + \alpha_r s^r + \dots + \alpha_1 s + \alpha_0$ is Hurwitz. In the original coordinates:

$$v = \left[\alpha_0 \ -\alpha_1 \dots -\alpha_r\right] \begin{bmatrix} z \\ h(x) \\ \vdots \\ L_f^{r-1} h(x) \end{bmatrix} \tag{8}$$

$$\dot{z} = y_{sp} - y$$

Effect of process constraints

We demonstrate that process constraints can severely degrade the performance of a conventional IOLC controller. Consider an irreversible, first-order chemical reaction $A \rightarrow B$ which occurs in a constant volume, continuous stirred tank reactor. The process model can be written as[26]:

$$\begin{aligned} \dot{C}_A &= \frac{q}{V}(C_{A_f} - C_A) - k_0 \exp\left(-\frac{E}{RT}\right) C_A \\ \dot{T} &= \frac{q}{V}(T_f - T) + \frac{(-\Delta H)}{\rho C_p} k_0 \exp\left(-\frac{E}{RT}\right) C_A \\ &+ \frac{UA}{V\rho C_p}(T_c - T) \end{aligned} \tag{9}$$

The nominal conditions in *Table 1* correspond to an unstable operating point. The manipulated input and controlled output are the coolant temperature (T_c) and reactor temperature (T), respectively. *Figure 1* shows the open-loop temperature response for ± 5 K step changes in the coolant temperature. The CSTR clearly exhibits highly nonlinear behavior in this operating regime.

By placing the reactor model (9) in standard form (1), it is easy to show that the relative degree $r = 1$. Therefore, the IOLC law has the form (4) where:

$$\begin{aligned} L_f h(x) &= \frac{q}{V}(T_f - T) + \frac{(-\Delta H)}{\rho C_p} k_0 \exp\left(-\frac{E}{RT}\right) C_A \\ &- \frac{UA}{V\rho C_p} T \\ L_g h(x) &= \frac{UA}{V\rho C_p} \neq 0 \end{aligned} \tag{10}$$

In this case, the linear pole-placement controller (8) is $v = \alpha_0 z + \alpha_1(\bar{T} - T)$, where $\bar{T}$ is the nominal reactor temperature. The controller tuning parameters are chosen as $\alpha_1 = 8$ and $\alpha_0 = 16$, which roughly corresponds to a closed-loop time constant of 0.25 min. This value is approximately one half the open-loop time constant for the –5 K step change shown in *Figure 1*.

Figures 2 and *3* show the performance of the IOLC controller with and without input constraints. In the

Table 1 Nominal operating conditions for the CSTR

Variable	Value	Variable	Value
q	100 l/min	$\frac{E}{R}$	8750 K
C_{Af}	1 mol/l	k_0	7.2×10^{10} min^{-1}
T_f	350 K	UA	5×10^4 J/min K
V	100 l	T_c	300 K
ρ	1000 g/l	C_A	0.5 mol/l
C_p	0.239 J/gK	T	350 K
$(-\Delta H)$	5×10^4 J/mol		

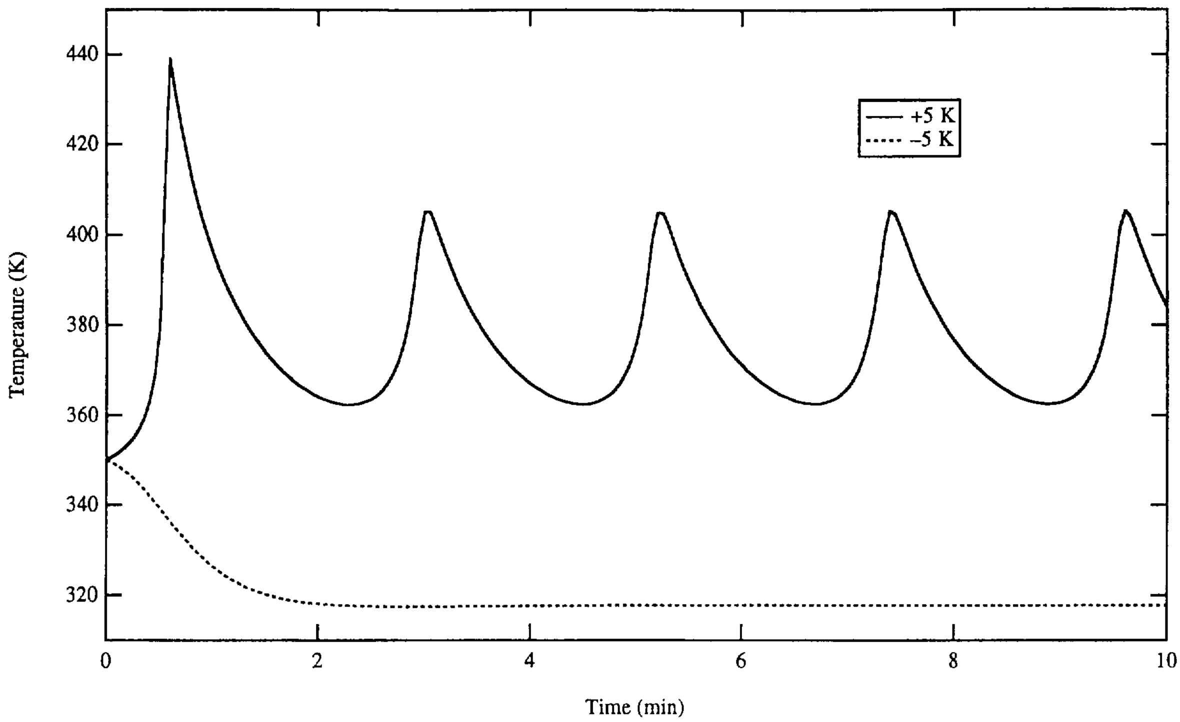

Figure 1 Open-loop response for coolant temperature changes

constrained case, the coolant temperature is bounded as 280 K ≤ T_c ≤ 380 K. *Figure 2* shows the effect of constraints on servo performance for a +25 K change in the temperature setpoint. The IOLC controller yields the prescribed setpoint response if T_c is unconstrained. However, controller performance is degraded significantly in the constrained case due to the input being 'clipped' by the lower constraint. *Figure 3* shows the effect of constraints on regulatory performance for a +35 K step change in the feed temperature (T_f). If T_c is unconstrained, the IOLC controller provides outstanding disturbance rejection. By contrast, the closed-loop system is unstable in the constrained case. The instability is attributable to a slight deviation of the input from its initial unconstrained trajectory. These results demonstrate that conventional IOLC controllers can be extremely sensitive to input constraints.

Input–output linearization for constrained systems

Motivated by the results presented above, we propose an input–output linearization strategy for constrained nonlinear systems. The basic idea is to map the actual input constraints into constraints on the manipulated input of the feedback linearized system. This transformation is performed at each sampling instant using the linearizing control law and the current state variables. The linear part of the input–output linearized system is discretized to yield a discrete-time linear model subject to *time-varying* input constraints and constant output constraints. This system is regulated with a linear model predictive controller (LMPC) with explicit constraint handling capability. The resulting control system is augmented with a disturbance model that ensures offset-free tracking.

At this point, it is important to note that the presence of constraints precludes feedback linearization in the traditional sense. The constraint mapping strategy yields a constrained linear system, which necessarily leads to a nonlinear control problem. However, the *linear* MPC design is much simpler than the *nonlinear* MPC design that would be required for the original constrained nonlinear system. In addition, exact input–output linearization is not actually achieved because the linearizing controller is discretized[27]. However, the discretized control law should provide 'approximate' linearization in most cases of practical interest. Therefore, we neglect the effects of discretization.

Input constraint mapping

The nonlinear process is assumed to have the following input and output constraints:

$$u_{\min} \le u \le u_{\max}, \quad \Delta u_{\min} \le \Delta u \le \Delta u_{\max}, \quad y_{\min} \le y \le y_{\max} \tag{11}$$

The objective is to transform these constraints into constraints on the feedback linearized system (5). First, the linear subsystem is discretized to facilitate the subsequent LMPC design. For a sampling period T, exact

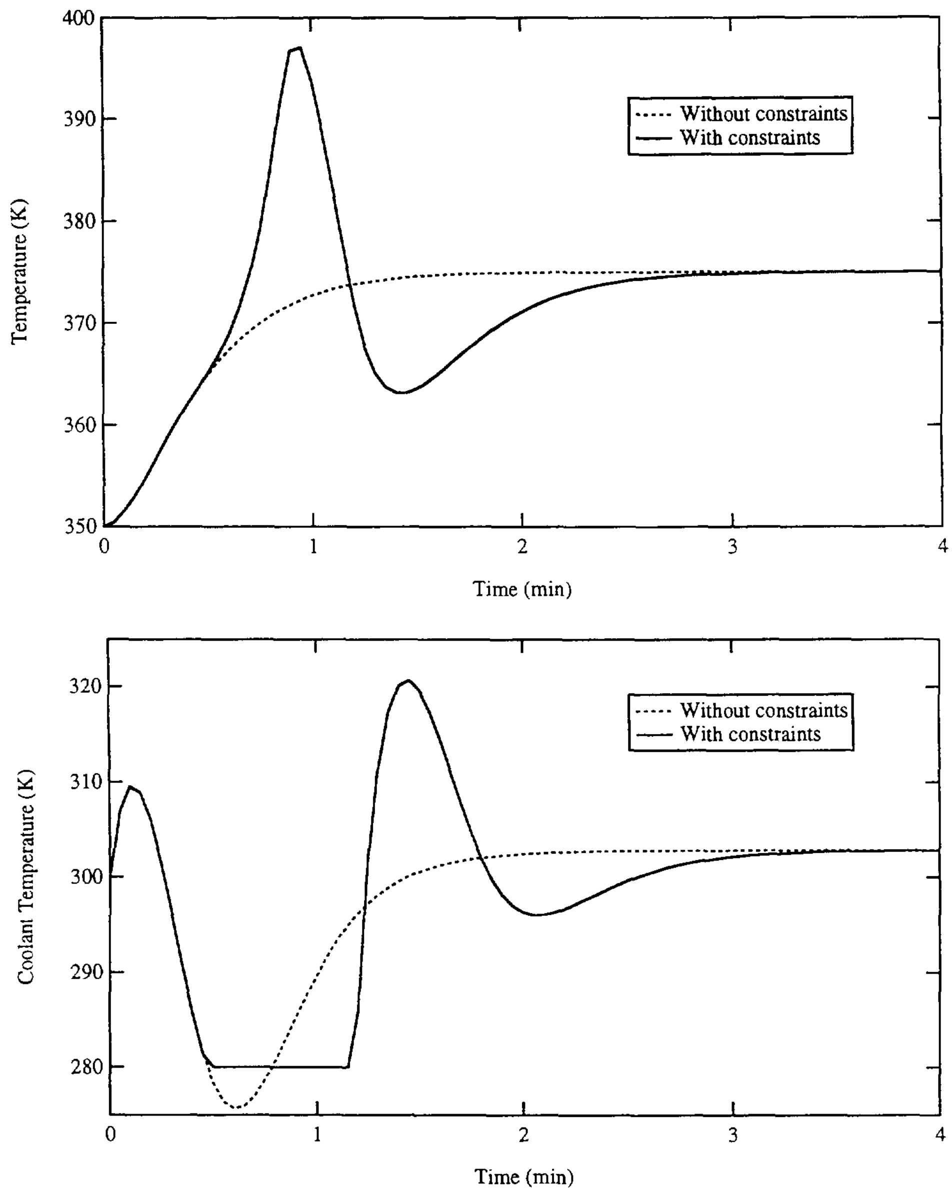

Figure 2 IOLC for a setpoint change

discretization yields[28]:

$$\xi(k+1) = A_d\xi(k) + B_d v(k) \tag{12}$$

$$y(k) = C\xi(k)$$

where the $r \times r$ matrix A_d and $r \times 1$ vector B_d have the form:

$$A_d = \begin{bmatrix} 1 & T & T^2 & \dots & T^{r-1} \\ 0 & 1 & T & \dots & T^{r-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \quad B_d = \begin{bmatrix} \dfrac{T^r}{r} \\ \dfrac{T^{r-1}}{r-1} \\ \vdots \\ T \end{bmatrix} \tag{13}$$

Because the linear subsystem in (5) is in Brunovsky canonical form, the pair (A_d, B_d) is controllable, but the eigenvalues of A_d are on the unit circle. We have investigated two ways of handling these unstable dynamics. In the first method, a linear state feedback controller is designed to stabilize the system prior to LMPC design. This method allows a comparatively simple LMPC technique for stable systems[14] to be employed. The stabilizing control law has the form,

$$v(k) = K\xi(k) + w(k) \tag{14}$$

where the feedback gain K is chosen such that the matrix $\bar{A}_d = A_d + B_d K$ has all its eigenvalues inside the unit circle. Consequently, the resulting system,

$$\xi(k+1) = \bar{A}_d\xi(k) + B_d w(k) \tag{15}$$

$$y(k) = C\xi(k)$$

is stable and the new input $w(k)$ can be used in the LMPC design[25]. A significant disadvantage of this tech-

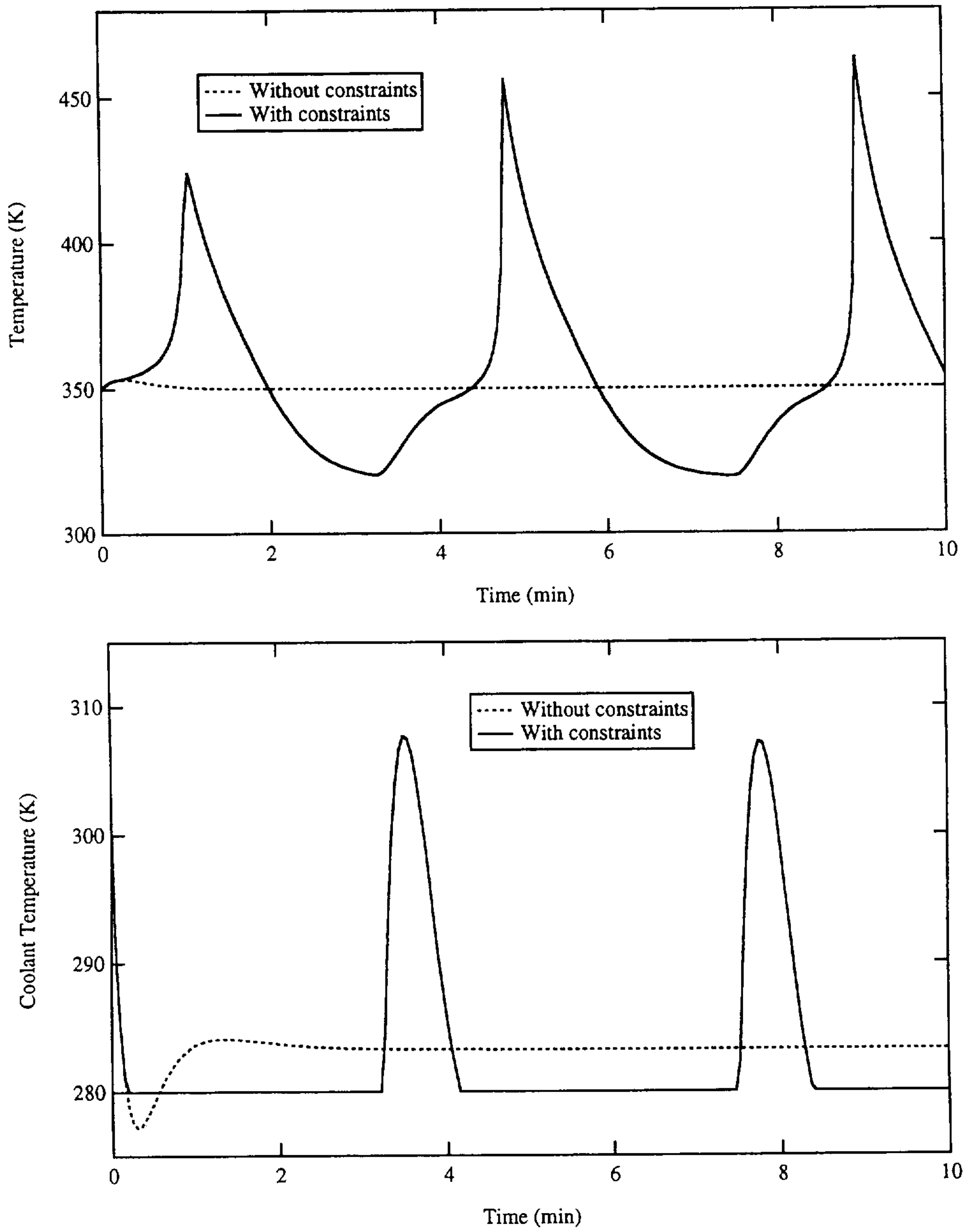

Figure 3 IOLC for a feed temperature disturbance

nique is that closed-loop performance depends on the eiganvalues of $\bar{A}_d$ and an appropriate choice cannot be determined *a priori*. In the technique pursued here, the unstable model (12) is employed directly in the LMPC design.

The next step is to map the constraints on the original nonlinear system (1) into constraints on the discretized linear system (12). The output constraints for the two systems are identical since the output y is not transformed as part of the IOLC design. By contrast, the constraints on the input u must be mapped into constraints on the new input v. This transformation must be performed at each sampling instant because the mapping is state dependent. Moreover, the transformation must be accomplished over the entire control horizon of the LMPC controller. Thus, at each time step k the objective is to find constraints of the form,

$$v_{min}(k+j|k) \leq v(k+j\,|\,k) \leq v_{max}(k+j\,|\,k), \quad 0 \leq j \leq N-1 \tag{16}$$

where: $v(k+j\,|\,k)$ is the value of the input $v(k+j)$ computed at time k; $v_{min}(k+j\ k)$ and $v_{max}(k+j\,|\,k)$ are the constraints $v_{min}(k+j)$ and $v_{max}(k+j)$, respectively, computed at time k; and N is the control horizon of the LMPC controller. Rate-of-change constraints on v are not shown explicitly because they can be converted into absolute constraints (shown below).

The input constraint mapping is performed using the IOLC law (4) and the current state measurement $x(k)$. The state-dependent relation between $u(k)$ and $v(k)$ follows from (4):

$$v(k) = L_f^r h[x(k)] + L_g L_f^{r-1} h[x(k)] u(k) \tag{17}$$

This mapping can be written as: $v(k) = b[x(k)] + a[x(k)]u(k)$. In the ideal case, the transformed constraints at time k are determined by solving the following optimization problem

$$v_{\min}(k+j \mid k) = \min_{u(k+j|k)} b[x(k+j \mid k)] + a[x(k+j \mid k)]u(k+j \mid k), \quad 0 \le j \le N-1 \tag{18}$$

$$v_{\max}(k+j \mid k) = \max_{u(k+j|k)} b[x(k+j \mid k)] + a[x(k+j \mid k)]u(k+j \mid k), \quad 0 \le j \le N-1$$

subject to the constraints:

$$\begin{aligned} &u_{\min} \le u(k+j \mid k) \le u_{\max} \\ &\Delta u_{\min} + u(k+j-1 \mid k) \le u(k+j|k) \\ &\qquad \le \Delta u_{\max} + u(k+j-1 \mid k) \end{aligned} \tag{19}$$

In (18)–(19), $u(k+j \mid k)$ represents the input $u(k+j)$ computed at time k, and $x(k+j \mid k)$ represents the state $x(k+j)$ computed at time k.

The obvious problem with the ideal mapping technique is that estimates of future values of the input and state variables are not available until the LMPC problem is solved, and the LMPC problem cannot be solved until the input constraints are specified. As a result, the solution of the ideal optimization problem requires a nonlinear programming strategy[23] or an iterative scheme[24]. Both techniques effectively eliminate the computational advantage of the proposed approach as compared to nonlinear MPC. Because exact mapping of future input constraints is impractical, it is necessary to approximate the constraints $v_{\min}(k+j \mid k)$ and $v_{\max}(k+j \mid k)$ for $j \ge 1$. Two approximate mapping techniques are discussed below.

Constant constraint technique. The most straightforward way to handle the constraint mapping problem is to simply extend the first input constraint over the entire control horizon[23]. The resulting optimization problem is

$$v_{\min}(k+j \mid k) = v_{\min}(k) = \min_{u(k)} b[x(k)] + a[x(k)]u(k), \quad 0 \le j \le N-1 \tag{20}$$

$$v_{\max}(k+j \mid k) = v_{\max}(k) = \max_{u(k)} b[x(k)] + a[x(k)]u(k), \quad 0 \le j \le N-1$$

where: $u_{\min} \le u(k) \le u_{\max}$, $\Delta u_{\min} + u(k-1) \le u(k) \le \Delta u_{\max} + u(k-1)$. Note that this problem is trivial to solve since $x(k)$ is known and the objective function is affine in $u(k)$. An important property of this method is that the first pair of constraints, $v_{\min}(k \mid k)$ and $v_{\max}(k \mid k)$, map *exactly* to the actual input constraints (11). As a result, the *implemented* input,

$$u(k) = \frac{v(k) - L_f^r h[x(k)]}{L_g L_f^{r-1} h[x(k)]} \tag{21}$$

is guaranteed to satisfy the *actual* constraints. On the other hand, the constraints $v_{\min}(k+j \mid k)$ and $v_{\max}(k+j \mid k)$ may not lead to inputs $v(k+j \mid k)$ that satisfy the actual constraints. Although these inputs are not implemented, this property is a potential disadvantage of the constant mapping technique since incorrect future constraints may lead to implemented control moves that are unnecessarily conservative or aggressive.

Variable constraint technique. The future input constraints can be calculated if an estimate of the future inputs is available. However, computationally intensive nonlinear programming or iterative solution strategies are required if future inputs calculated at the *current* sampling time are utilized for constraint mapping. A much simpler approach is to use inputs calculated at the *last* sampling time to determine future constraints at the current sampling time. This method is outlined below.

As shown in the next section, solution of the LMPC problem at time $k-1$ yields the input sequence

$$V(k-1 \mid k-1) = \left[v(k-1 \mid k-1) \;\; v(k \mid k-1) \ldots v(k+N-2 \mid k-1)\right]^T \tag{22}$$

The first input $v(k-1 \mid k-1)$ is used to calculate the implemented input $u(k-1)$. We use the remaining inputs as an estimate of the control sequence at the current sampling time:

$$V(k \mid k-1) = \left[v(k \mid k-1) \;\; v(k+1 \mid k-1) \ldots v(k+N-2 \mid k-1) \;\; v_a\right]^{\mathrm{T}} \tag{23}$$

where the value v_a is arbitrary since it is not actually utilized. The current measurement $x(k)$ is used to calculate the transformed state variables $\xi(k)$ and $\eta(k)$ via the nonlinear change of coordinates $\Phi(x)$. The normal form (5) is integrated with the piecewise constant input sequence $V(k \mid k-1)$ to yield predicted values of the transformed state variables

$$Z(k \mid k-1) = [\xi^{\mathrm{T}}(k \mid k-1) \;\; \xi^{\mathrm{T}}(k+1 \mid k-1) \ldots \xi^{\mathrm{T}}(k+N-1 \mid k-1)]^{\mathrm{T}} \tag{24}$$

$$N(k \mid k-1) = [\eta^{\mathrm{T}}(k \mid k-1) \;\; \eta^{\mathrm{T}}(k+1 \mid k-1) \ldots \eta^{\mathrm{T}}(k+N-1 \mid k-1)]^{\mathrm{T}}$$

where: $\xi(k \mid k-1) = \xi(k)$, $\eta(k \mid k-1) = \eta(k)$. It is important to note that the second time index denotes that the predictions are based on the input sequence at time $k-1$, even though the current measurement $x(k)$ is used. The state sequences and the inverse transformation $\Phi^{-1}(\xi,\eta)$ are used to compute future values of the actual state vector:

$$X(k \mid k-1) = [x^T(k \mid k-1) \quad x^T(k+1 \mid k-1) \ldots \\ x^T(k+N-1 \mid k)]^T \tag{25}$$

If the system is subject to rate-of-change constraints, predicted values of the actual input sequence are required to calculate the constraints (19). By utilizing the vectors $X(k \mid k-1)$ and $V(k \mid k-1)$ in the discretized version of the IOLC control law (4), the predicted input vector can be computed as

$$U(k \mid k-1) = [u(k \mid k-1) \quad u(k+1 \mid k-1) \ldots \\ u(k+N-2 \mid k-1) \quad u_a]^T \tag{26}$$

where the value u_a is arbitrary. The rate-of-change constraints are handled by substituting these estimates for $u(k+j-1 \mid k)$ in (19). The optimization problem (18) is solved by substituting the predicted state variables $x(k+j \mid k-1)$ in place of $x(k+j \mid k)$. The solution yields the transformed constraints:

$$V_{\min}(k \mid k-1) = [v_{\min}(k \mid k-1) \quad v_{\min}(k+1 \mid k-1) \ldots \\ v_{\min}(k+N-1 \mid k-1)]^T \tag{27}$$

$$V_{\max}(k \mid k-1) = [v_{\max}(k \mid k-1) \quad v_{\max}(k+1 \mid k-1) \ldots \\ v_{\max}(k+N-1 \mid k-1)]^T$$

These variable constraints are used in the LMPC design in place of the constant constraints $v_{\min}(k)$ and $v_{\max}(k)$. The procedure is repeated at the next time step with the input sequence $V(k \mid k)$ and the measurement $x(k+1)$. The algorithm is initialized by using the constant constraint mapping scheme during the first iteration.

Note that the first set of constraints, $v_{\min}(k \mid k-1)$ and $v_{\max}(k \mid k-1)$, map exactly to the actual input constraints since they are calculated using the current state measurement. Therefore, the implemented input (21) necessarily satisfies the actual constraints. As compared to the constant mapping technique, the major advantage of this method is that calculated constraints are more likely to agree with the actual constraints. As a result, the control system should exhibit improved perform-ance and robustness.

Linear model predictive controller design

The LMPC design is based on the linear model

$$\xi(k+j+1 \mid k) = A_d\xi(k+j \mid k) + B_d v(k+j \mid k) \tag{28}$$
$$y(k+j \mid k) = C\xi(k+j \mid k)$$

subject to the constraints:

$$v_{\min}(k+j \mid k) \le v(k+j \mid k) \le v_{\max}(k+j \mid k) \tag{29}$$
$$y_{\min} \le y(k+j \mid k) \le y_{\max}$$

It is important to reiterate that the input constraints vary with respect to the sampling time k, and they also vary over the control horizon if the variable constraint mapping technique is employed. The linear model is used by the LMPC controller to predict the effects of future control moves on future outputs. To obtain improved predictions in the presence of plant/model mismatch, at each time step the linear model is initialized with the current plant state as follows:

$$\xi(k \mid k) = \left[h[x(k)] \quad L_f h[x(k)] \ldots L_f^{r-1} h[x(k)]\right]^T \tag{30}$$

Recall that the matrix A_d is unstable because all its eigenvalues are located at $z = 1$. Thus, we utilize an infinite horizon LMPC design technique specifically developed for unstable systems[29]. The open-loop optimal control problem can be expressed as

$$\min_{V(k|k)} \sum_{j=0}^{\infty} [\xi(k+j \mid k) - \xi_s]^T Q [\xi(k+j \mid k) - \xi_s] \\ + r[v(k+j \mid k) - v_s]^2 + s[v(k+j \mid k) \\ - v(k+j-1 \mid k)]^2 \tag{31}$$

where: ξ_s and v_s are target values for ξ and v, respectively; $r > 0$ and $s \ge 0$ are scalar tuning parameters; and Q is a positive semidefinite tuning matrix. The decision vector is defined as: $V(k|k) = [v(k|k), v(k+1|k), \ldots, v(k+N-1|k)]^T$. To obtain a finite set of decision variables, inputs beyond the control horizon are set equal to the target value: $v(k+j \mid k) = v_s, j \ge N$.

A necessary condition for the optimization problem to have a solution is that $\xi(k)$ converges to ξ_s. This requires that the unstable modes are driven to their steady-state values by the end of the control horizon. Because all the eigenvalues of A_d are on the unit circle, the following equality constraint must be satisfied at each k: $\xi(k+N / k) = \xi_s$. Otherwise, the system evolves in open-loop with an initial condition $\xi(k+N|k) \ne \xi_s$ and the state variables will not converge to their target values. Thus, the optimization must be solved subject to the following constraints:

$$v_{\min}(k+j \mid k) \le v(k+j \mid k) \le v_{\max}(k+j \mid k)$$
$$y_{\min} \le C\xi(k+j \mid k) \le y_{\max} \tag{32}$$
$$\xi(k+N \mid k) = \xi_s$$

The targets ξ_s and v_s are calculated from the steady-state form of (28) under the condition that $y = y_{sp}$, where y_{sp} is the setpoint. Under nominal conditions, it is easy to show that:

$$\xi_s = [y_{sp}\ 0 \ldots 0]^T, \quad v_s = 0 \tag{33}$$

The target values must lie within the feasible region defined by the input and output constraints for the LMPC problem to have a solution. As discussed in the next section, the targets can be shifted to eliminate steady-state offset caused by plant/model mismatch.

The infinite horizon LMPC problem (31) can be written as a finite horizon problem[14]

$$\min_{V(k|k)} s[v(k+N-1\mid k)-v_s]^2+\sum_{j=0}^{N-1}[\xi(k+j\mid k) -\xi_s]^{\mathrm{T}}Q[\xi(k+j\mid k)-\xi_s]+r[v(k+j\mid k)-v_s]^2 + s[v(k+j\mid k)-v(k+j-1\mid k)]^2 \tag{34}$$

subject to the constraints (32). This finite horizon problem can be manipulated to yield the following quadratic program for the input sequence $V(k) = V(k \mid k)$,

$$\min_{V(k)} V^{\mathrm{T}}(k)HV(k)+2V^{\mathrm{T}}(k)[G\xi(k)-Fv(k-1)] \tag{35}$$

The quadratic program is solved subject to following constraints:

$$DV(k) \leq d_1(k)\xi(k) + d_2(k) \tag{36}$$
$$EV(k) = e_1\xi(k)$$

The construction of H, G, F, D, d_1, d_2, E, and e_1 is discussed elsewhere[14]. In the present case, it is important to note that d_1 and d_2 are functions of time. A state feedback control law is obtained by implementing only the first calculated input $v(k) = v(k \mid k)$, and resolving the problem at the next sampling time with the new measurement $x(k + 1)$. The actual input $u(k)$ is calculated from $v(k)$ as in (21).

There exists a control horizon N such that the quadratic program (35)–(36) is feasible if the linear system is constrained stabilizable. For a given value of N and initial condition $\xi(k)$, feasibility can be checked with a linear program[30]. If the constraints are constant, feasibility at $k = 0$ implies feasibility at all future times. This property no longer holds if input constraints vary with time, in which case constrained stabilizability has to be checked at each time step. We address this problem in the following way. If the LMPC problem is infeasible, constraints are dropped on the last input in the control horizon, $v(k + N - 1 \mid k)$, and the problem is resolved. If the problem remains infeasible, constraints are dropped on the last two inputs, $v(k + N - 2 \mid k)$ and $v(k + N - 1 \mid k)$. The process is continued until feasibility is achieved. Also, the variable constraint mapping strategy is modified so unconstrained inputs are not used for constraint prediction. In this case, we extend the last constrained input over the control horizon to obtain the input sequence $V(k \mid k - 1)$. It is important to note that output constraints also can be relaxed to ensure that the optimization problem is feasible[30].

Disturbance model

The LMPC controller generates a proportional state feedback. As a result, offset will generally occur in the presence of plant/model mismatch. Offset is eliminated by introducing a disturbance model that shifts the target values v_s and ξ_s in the LMPC optimization problem (34). Available disturbance modeling techniques focus on the output feedback case[14,29]. A method that ensures offset-free performance in the state feedback case is presented below.

The disturbance model is obtained by augmenting (12) with a q-dimensional disturbance vector $d(k)$:

$$\xi(k+1) = A_d\xi(k) + B_dv(k) + Gd(k)$$
$$d(k+1) = Pd(k) \tag{37}$$
$$y_{\mathrm{m}}(k) = \xi(k)$$

The vector of measured outputs y_{m} is the entire state vector ξ since state feedback is assumed. Recall that $\xi(k)$ is determined from the actual state variables $x(k)$ as in (30). The first step is to use the augmented model to build an observer that provides an estimate of the disturbance vector d. It is easy to show that the augmented model is observable if $q = r$ and $P = G = I$. The observer has the form

$$\hat{\xi}(k+1) = A_d\hat{\xi}(k) + B_dv(k) + \hat{d}(k) + L_1[\xi(k) - \hat{\xi}(k)]$$
$$\hat{d}(k+1) = \hat{d}(k) + L_2[\xi(k) - \hat{\xi}(k)] \tag{38}$$

where $\hat{\xi}$ and $\hat{d}$ are estimates of ξ and d, respectively, and L_1 and L_2 are $r \times r$ observer gain matrices. If the estimation error is defined as

$$e(K) = \begin{bmatrix} \xi(K)-\hat{\xi}(K) \\ d(k)-\hat{d}(k) \end{bmatrix} \tag{39}$$

the observer error dynamics can be written as:

$$e(k+1) = \begin{bmatrix} A_d - L_1 & I \\ -L_2 & I \end{bmatrix} e(k) = Pe(k) \tag{40}$$

Because the disturbance model is observable, the eigenvalues of the matrix P can be placed arbitrarily via appropriate choice of the gains L_1 and L_2.

The disturbance estimate is used to shift the target values v_s and ξ_s in the LMPC objective function. By appending the current estimate $\hat{d}(k)$ to the state equations, the linear model (28) has the following representation at steady-state:

$$\begin{bmatrix} I-A_d & B_d \\ C & 0 \end{bmatrix}\begin{bmatrix} \xi_s \\ v_s \end{bmatrix} = \begin{bmatrix} \hat{d}(k) \\ y_{\mathrm{sp}} \end{bmatrix} \tag{41}$$

This set of linear algebraic equations is solved for the targets ξ_s and v_s; a unique solution always exists. Fol-

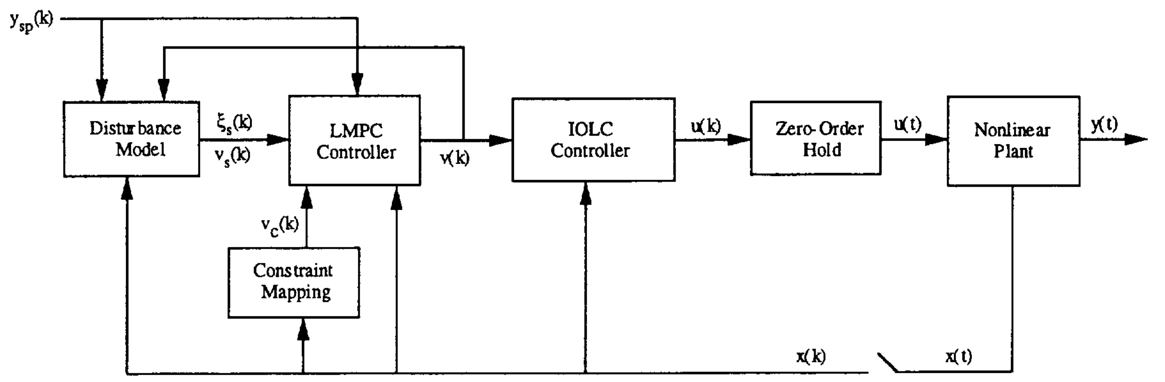

Figure 4 Block diagram of the input–output linearization strategy for constrained processes

lowing the proof of Rawlings *et al.*[12], it can be shown that the proposed scheme eliminates offset.

A block diagram of the proposed nonlinear control system is shown in *Figure 4*. The state vector is sampled

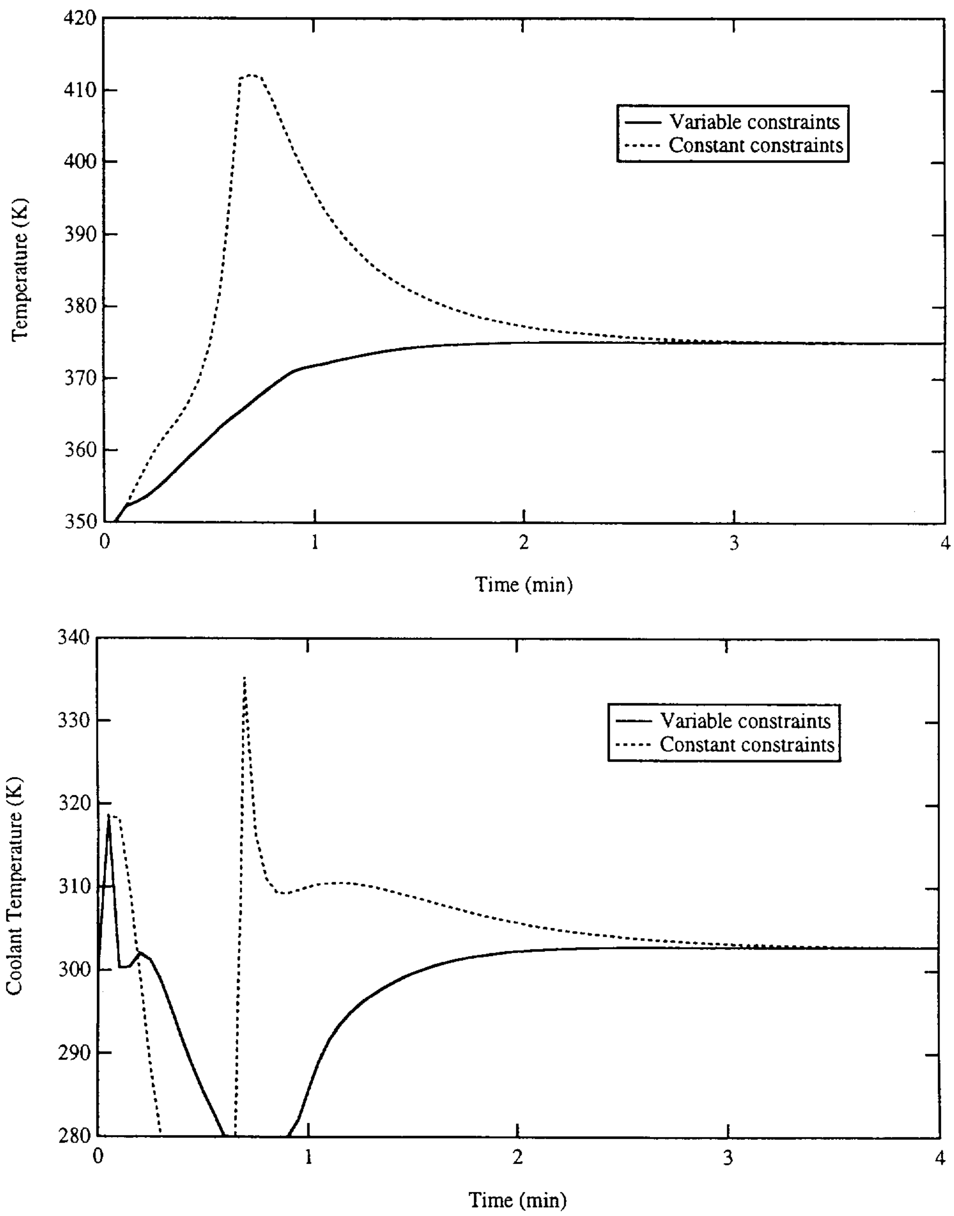

Figure 5 Comparison of constraint mapping techniques for a setpoint change

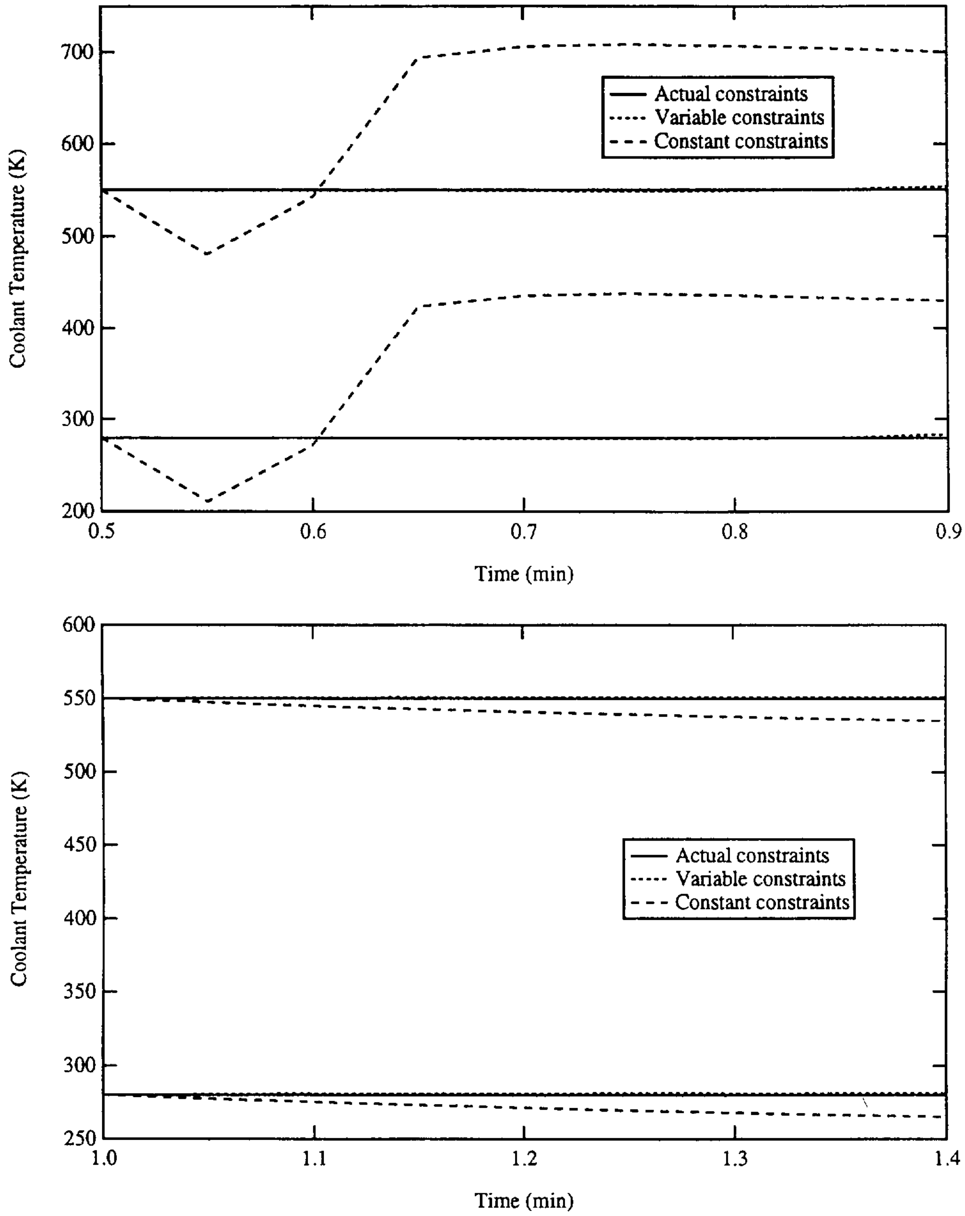

Figure 6 Constraint predictions at t = 0.5 min (top) and t = 1 min (bottom) for *Figure 5*

to yield the discrete value $x(k)$ used by the constraint mapping scheme, the disturbance model, the LMPC controller, and the IOLC controller. The disturbance model generates shifted targets $\xi_s(k)$ and $v_s(k)$ from $x(k)$, the setpoint $y_{sp}(k)$, and the transformed input $v(k)$. The LMPC controller computes $v(k)$ using the setpoint, the sampled state variables, the shifted targets, and the transformed constraints $v_c(k)$. The IOLC controller uses $v(k)$ to calculate the discrete input $u(k)$. The continuous input $u(t)$ injected into the nonlinear plant is obtained by applying a zero-order hold to $u(k)$.

Simulation study

Comparison of constraint mapping techniques

First, the two constraint mapping strategies are compared using the CSTR model described in the section on 'Effect of process constraints'. The CSTR is operated at the unstable operating point shown in *Table 1*, and the manipulated input is constrained as: 280 K $\leq T_c \leq$ 380 K. Output constraints are not considered in this example. The IOLC controller is designed as in the section on 'Effect of process constraints'. Because the relative degree r = 1, the feedback linearized system has the following form after discretization,

$$\xi(k+1) = \xi(k) + Tv(k) \quad (42)$$
$$y(k) = \xi(k)$$

where the sampling period T = 0.05 min. The LMPC controller is designed with a control horizon N = 10, which provides an effective control horizon NT = 0.5 min. The remaining tuning parameters q = 2, r = 1, and s = 1 are determined by trial and error. The target values are calculated as in the section 'Disturbance model'

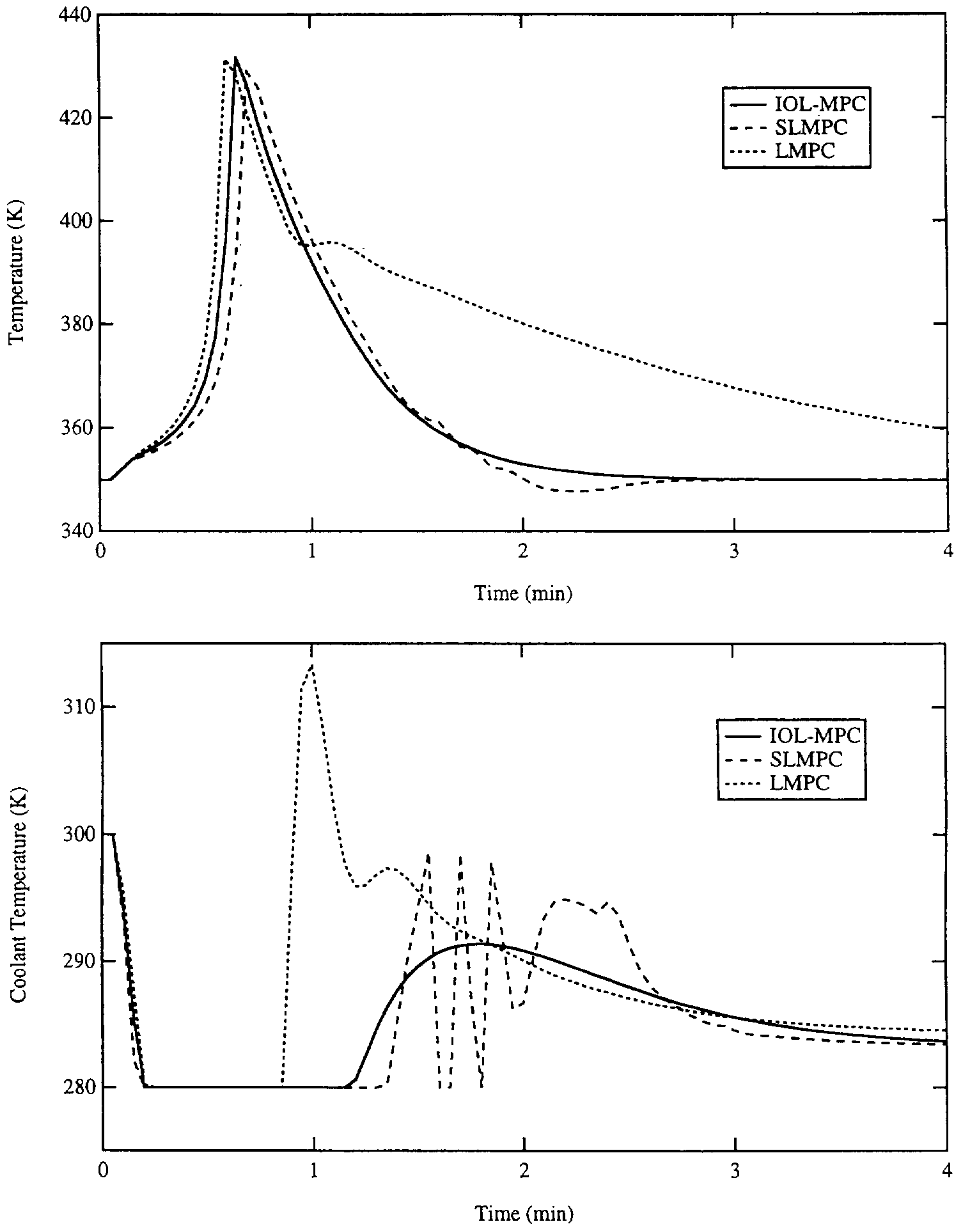

Figure 7 IOL-MPC, LMPC, and SMPC for a feed temperature disturbance

with the two observer poles placed at 0.8. The only difference between the two controllers compared is the method used to transform the input constraints into the feedback linearized space:

- Constant technique: $v_{min}(k) \leq v(k+j|k) \leq v_{max}(k)$
- Variable technique: $v_{min}(k+j|k) \leq v(k+j|k) \leq v_{max}(k+j|k)$

In each case, the constraint mapping is performed as in the section on 'Input constraint mapping'.

The two constraint mapping strategies are compared in *Figures 5* and *6* for a +25 K step change in the temperature setpoint. The controller which uses input constraints that are constant over the control horizon produces a large overshoot and aggressive control moves. By contrast, the controller which employs constraints that vary over the control horizon yields significantly improved setpoint tracking and more conservative control moves. A possible explanation for this behavior is shown in *Figure 6*, where the coolant temperature constraints generated by each controller are compared to the actual values at two particular time steps. The controller values are determined by mapping the constraints on the transformed input v to constraints on T_c using the *actual* state values that occur in the future. At $t = 0.5$ min, the constant technique produces extremely poor constraint predictions that appear to significantly degrade closed-loop performance (see *Figure 5*). By contrast, the variable method yields constraints that are almost identical to the actual values. The difference is less pronounced at $t = 1$ min since the state variables are changing more slowly, but the same general trend is observed. Based on these results we only consider the variable mapping technique in the sequel.

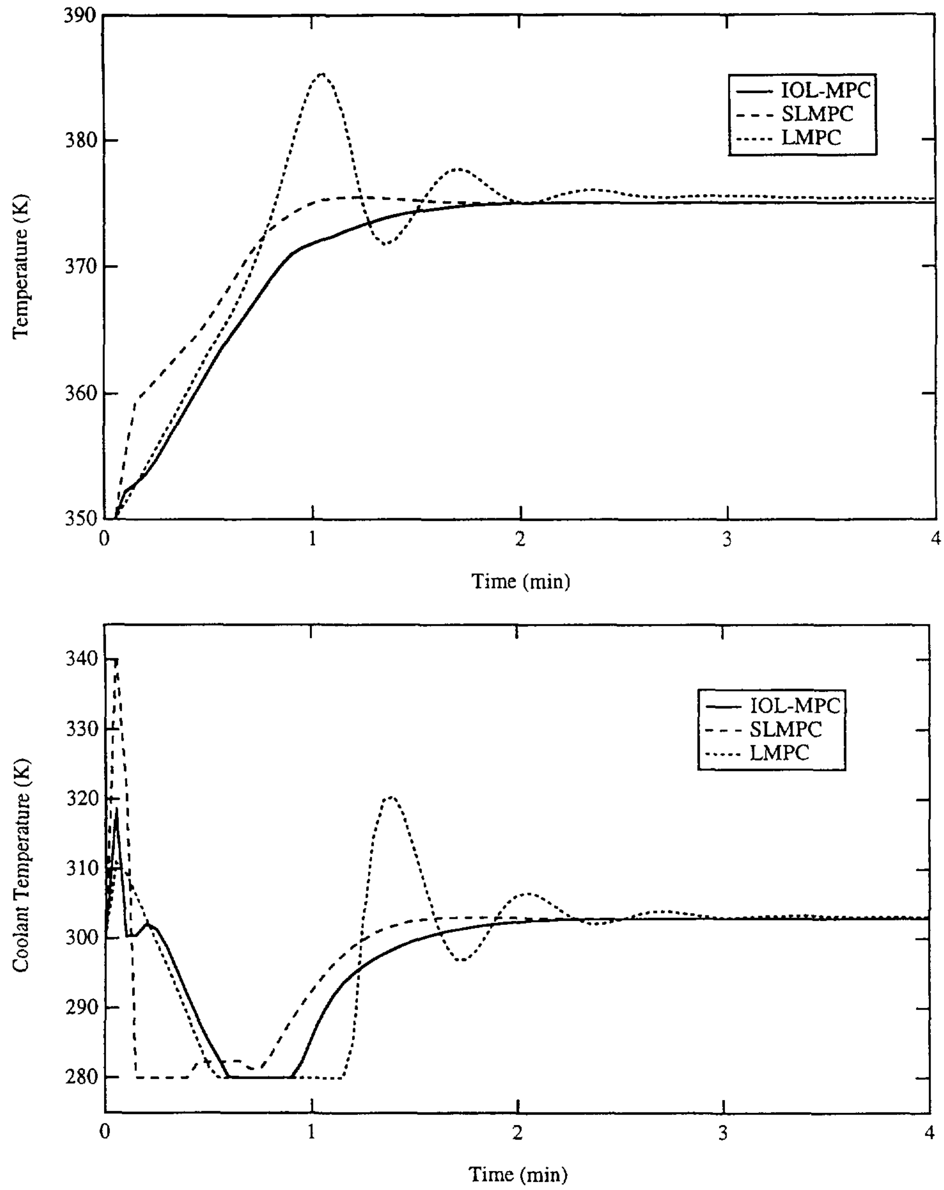

Figure 8 IOL-MPC, LMPC, and SMPC for a positive setpoint change

Comparison with model predictive control

We now compare the proposed control strategy to linear and nonlinear MPC techniques. The proposed controller employs the variable constraint mapping scheme and is tuned as before; it is called 'IOL-MPC' in the sequel. The LMPC design is based on a linear model that is obtained by linearizing (via Taylor series expansion) the CSTR model at the unstable operating point in *Table 1*; this controller is called 'LMPC.' The nonlinear MPC technique utilizes a linear model that is obtained by successively linearizing the CSTR model at the current operating point[31]; this controller is called 'SLMPC.' Note that conventional nonlinear MPC is not considered because we are interested in comparing control strategies that have comparable computational requirements.

The MPC controllers are designed by discretizing the respective linear model with a sampling period $T = 0.05$ min and solving an infinite horizon optimal control problem similar to that in the section on 'Linear model predictive controller design'. Each controller utilizes a disturbance model with the four observer poles placed at 0.5, 0.5, 0.6, and 0.6. The following controller tuning parameters are chosen by trial-and-error to provide a fast, smooth response to a positive setpoint change:

- LMPC: $N = 16$, $q = 1$, $r = 4$, $s = 1$
- SLMPC: $N = 16$, $q = 4$, $r = 0.3$, $s = 0.3$.

It is interesting to note that the three controllers require different values of the tuning parameters.

The performance of the three controllers for a +35 K disturbance in the feed temperature (T_f) is shown in *Figure 7*. As before, the input is constrained as: 280 K $\leq T_c \leq$ 380 K. LMPC yields a very sluggish response and unnecessarily large control moves. SLMPC provides significantly improved disturbance rejection, but generates oscillatory control moves. IOL-MPC yields an output response that is very similar

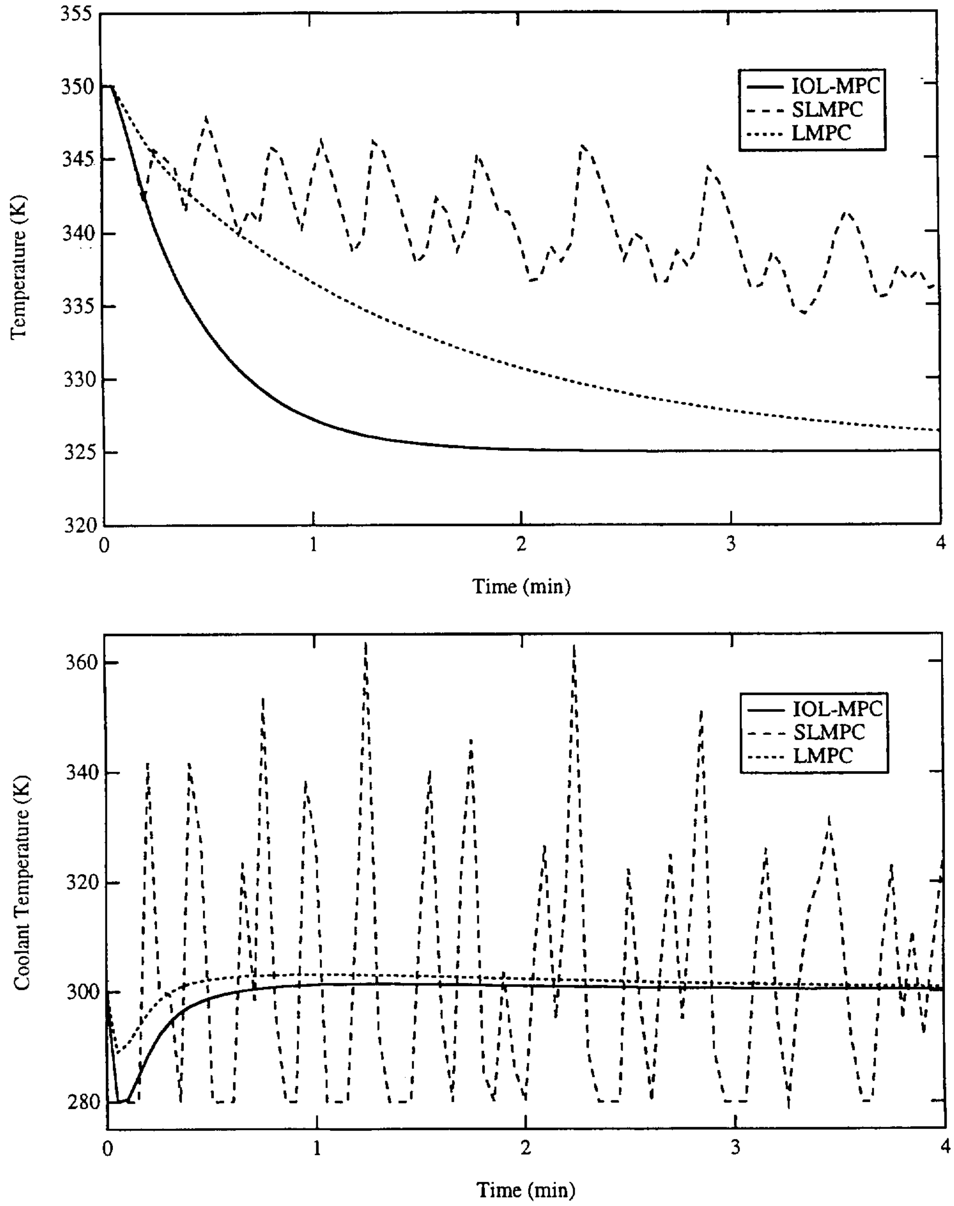

Figure 9 IOL-MPC, LMPC, and SMPC for a negative setpoint change

to that produced by SLMPC, but with much smoother control moves. In *Figure 8*, the three controllers are compared for a +25 K change in the temperature setpoint. LMPC yields poor setpoint tracking as the output oscillates and overshoots the setpoint. SLMPC provides improved servo performance with less aggressive control moves. IOL-MPC yields the best setpoint response yet produces the most conservative control moves.

The controllers are compared for a –25 K setpoint change in *Figure 9*. LMPC produces a very sluggish response since the linear model used is very inaccurate at the new setpoint. SLMPC had to be detuned with $r = s = 0.5$ to obtain a stable closed-loop response. Even with detuning, it is unable to attain the new setpoint and the control moves are extremely erratic. This behavior is attributable to the successive linearization repeatedly switching between stable and unstable models. Improved performance cannot be obtained unless the controller is detuned further. By contrast, IOL-MPC yields a fast, smooth setpoint response with little control effort.

Summary and conclusions

An input–output linearizing control strategy for constrained nonlinear processes has been developed and evaluated. The control system is comprised of: (i) an input–output linearizing controller that accounts for process nonlinearities; (ii) a constraint mapping scheme that transforms the actual input constraints into input constraints on the feedback linearized system; (iii) a linear model predictive controller that provides explicit compensation for input and output constraints; and (iv) a disturbance model that ensures offset-free performance. The control strategy retains the computational simplicity of input–output linearizing control while providing the constraint handling capability of model predictive control. Simulation results for a continuous stirred tank reactor show that the proposed method provides significantly improved performance as compared to conventional input–output linearizing control and model predictive control based on local and successive model linearization. Our future research efforts will focus on nominal stability analysis and additional applications of the proposed control strategy.

Acknowledgements

The authors would like to acknowledge the National Science Foundation (CTS-9501368) for supporting this research.

References

1 Biegler, L. T. and Rawlings, J. B. in 'Chemical Process Control *IV*' (Eds Y. Arkun and W.H. Ray), CACHE, AIChE, Austin, TX, 1991
2 Eaton, J. W. and Rawlings, J. B. *Comput. Chem. Engng.* 1990, **14**, 469
3 Henson, M. A. and Seborg, D. E. *Int. J. Control*, 1993, **58**, 1053
4 Soroush, M. and Kravaris, C. in 'Proc. American Control Conf.' San Francisco, 1993, 393
5 Keerthi, S. S. and Gilbert, E. G. *J. Optimiz. Theory Applic.* 1988, **57**, 265
6 Meadows, E. S., Henson, M. A., Eaton, J.W. and Rawlings, J.B. *Int. J. Control* 1995, **62**, 1217
7 Mayne, D. Q. and Michalska, H. *IEEE Trans. Autom. Control* 1990, AC-**35**, 814
8 Henson, M. A. and Seborg, D. E. *J. Process Control* 1991, **1**, 122
9 Isidori, A. 'Nonlinear Control Systems' Springer-Verlag, New York, NY, 1989
10 Kravaris C. and Kantor, J. C. *Ind. Eng. Chem. Res.* 1990, **29**, 2310
11 Bequette, B. W. *Ind. Eng. Chem. Res.* 1991, **30**, 1391
12 Rawlings, J. B., Meadows, E. S. and Muske, K. R. in 'Proc. IFAC Symposium on Advanced Control of Chemical Processes' Kyoto, Japan, 1994, 203
13 Alvarez, J., Alvarez, J. and Suarez, R. *Chem. Eng. Sci.* 1991, **46**, 3235
14 Muske, K. R. and Rawlings, J. B. *AIChE J.* 1993, **39**, 262
15 Saberi, A., Lin, Z. and Teel, A. in 'Proc. American Control Conf.' Seattle, WA, 1995, 285
16 Kapoor, N. and Daoutidis, P. in 'Proc. American Control Conf.' Seattle, WA, 1995, 3192
17 Kendi, T. A. and Doyle, F. J. in 'Proc. European Control Conf.' Rome, Italy, 1995, 2653
18 Soroush, M. and Kravaris, C. *AIChE J.* 1992, **38**, 1429
19 Calvet, J.-P. and Arkun, Y. *Ind. Eng. Chem. Res.* 1988, **27**, 1822
20 Lee, A. W. and Hedrick, J. K. *ASME J. Dynamic Systems, Measurement, and Control* 1994, **116**, 429
21 Zhou, W., Lee, P. L., Sullivan, G. R. and Brown, M. W. *Chem. Engng. Commun.* 1990, **97**, 109
22 Balchen, J.G. and Sandrib, B. *Model. Ident. Control* 1995, **16**, 95
23 Nevistic, V. and Del Re, L. In 'Proc. American Control Conf.' 1994, 2862
24 Oliveira, S. L., Nevistic, V. and Morari, M. in 'Proc. IFAC Symposium on Nonlinear Control Systems Design' Tahoe City, CA, 1995, 15
25 Henson, M. A. and Kurtz, M. J. in 'AIChE Annual Mtg' San Franciso, CA, 1994
26 Uppal, A., Ray, W. H. and Poore, A.B. *Chem. Engng. Sci.* 1974, **29**, 67
27 Grizzle, J.W. and Kokotovic, P.V. *IEEE Trans. Autom. Control* 1988, AC-**33** 857
28 Astrom, K.J. and Wittenmark, B. 'Computer Controlled Systems: Theory and Design' Prentice-Hall, Englewood Cliffs, NJ, 1984
29 Muske, K. R. and Rawlings, J. B. *J. Process Control* 1993, **3**, 85
30 Rawlings, J. B. and Muske, K. R. *IEEE Trans. Autom. Control* 1993 AC-**38**, 1512
31 Garcia, C. E. in 'AIChE Annual Mtg' San Francisco, 1984