Model-Based Synthesis of Nonlinear PI and PID Controllers

Raymond A. Wright
The Dow Chemical Company, Midland, MI 48667

Costas Kravaris
Dept. of Chemical Engineering, The University of Michigan, Ann Arbor, MI 48109

Nikolaos Kazantzis
Dept. of Chemical Engineering, Texas A&M University, College Station, TX 77843

PI and PID controllers continue to be popular methods in industrial applications. It is well known that linear PI and PID controllers result from the application of model-based controller design methods to linear first- and second-order systems. It is shown that nonlinear PI and PID controllers result from the application of nonlinear controller design methods to nonlinear first- and second-order systems. As a result, the controllers resulting from nonlinear model-based control theory are put in a convenient form, more amenable to industrial implementation. Additionally, the quantities used in the controller are useful for monitoring the process and quantifying modeling error. Chemical engineering examples are used to illustrate the resulting control laws. A simulation example further demonstrates the performance of the nonlinear controllers, as well as their useful process monitoring quantities.

Introduction

PI and PID controllers continue to be popular methods for controlling industrial processes, due to their simplicity in implementation and understanding. In general, as can be surmised from the names, a PI controller possesses proportional and integral action, and a PID controller possesses proportional, integral, and derivative action. The standard PI and PID transfer-function forms that are usually presented in textbooks (such as Stephanopoulos, 1984) are

$$\frac{\bar{u}(s)}{\bar{e}(s)} = k_c\left(1 + \frac{1}{\tau_I s}\right) \qquad (1)$$

$$\frac{\bar{u}(s)}{\bar{e}(s)} = k_c\left(1 + \frac{1}{\tau_I s} + \tau_D s\right), \qquad (2)$$

where u is the manipulated input, e is the error signal, s is the Laplace transform variable, and k_c, τ_I, and τ_D are the

tunable parameters of the controller. Viewed in its most basic form, these controllers are simply a linear combination of the 2 (or 3) modes of action. The inclusion of integral action necessarily implies that both controllers are dynamic systems; in a state space representation, there will be a state associated with the controller.

In practice, there are several variations of Eqs. 1 and 2 that can be used. For example, the derivative mode of a PID controller may be a numerical derivative, or a filtered value to prevent the amplification of noisy signals (Stephanopoulos, 1984; Morari and Zafiriou, 1989; Ogata, 1990). It may be the derivative of the error signal, or the derivative of the measured output, to prevent step changes in the setpoint value from directly effecting the derivative (Åström and Wittenmark, 1990). The controllers shown in Eqs. 1 and 2 operate solely on the error signal. It is also possible to operate differently on the setpoint value and the measured output. Such a case would be if the derivative was in terms of the measured output only and not based on the error signal (Åström and Wittenmark, 1990; Ogata, 1990). If the controller is solely dependent on the error signal, it can be called an error-feed-

back controller, or a one-degree-of-freedom controller. If the control action depends on both the error and the output measurement, it is a mixed error–output feedback controller, or a two-degree-of-freedom controller (Kailath, 1980).

The application of linear model-based controller synthesis methods to linear first- and second-order models can also give rise to a controller of the form of Eq. 1 or 2. As shown in Rivera et al. (1986), an internal model controller (IMC) derived from a first-order linear model is exactly a linear PI controller, and an IMC controller derived from a linear second-order model is exactly a linear PID controller. The resulting controller, while of the form of Eq. 1 or Eq. 2, has the controller gains, k_c τ_I, and τ_D, as a function of the model parameters and one tunable parameter, which is the closed-loop time constant. Thus, a direct connection can be established between model parameters and controller tuning.

In order to achieve tighter closed-loop control of nonlinear processes, it is natural to generalize Eqs. 1 or 2 and develop nonlinear PI and PID controllers. One way would be to add higher-order terms of the error and integral of the error to the control law. Another way suggested in prior work is to make the controller parameters functions of the error (Cheung and Luyben, 1980), or PID parameter scheduling (Rugh, 1987), or both (Jutan, 1989). In general, many different types of controllers can be classified as nonlinear PI or nonlinear PID, as long as the resulting controller can be shown to possess proportional, integral, and possibly derivative action. There is no particular form that has become standard in the literature.

The present work focuses on the development of nonlinear model-based controllers from nonlinear first- and second-order models. This will include nonlinear processes that are exactly first or second order, as well as those processes for which a reasonably accurate low-order model exists and is first or second order (Lindskog and Ljung, 1994; Ogunnaike et al., 1994; Balasubramhanya and Doyle, 1995; Pearson and Ogunnaike, 1996). In particular, it is shown that the resulting model-based controllers are exactly nonlinear PI and PID controllers, respectively. These results are in the same spirit as those of Rivera et al. (1986), for linear systems. The relation between the model parameters and the controller gains is directly shown, and the tuning of the resulting nonlinear PI or PID controller will be in terms of the closed-loop time constant. The resulting controllers also bring general nonlinear controller synthesis to a form that is directly applicable in practice. Furthermore, the terms used in calculating the control action enable the generation and the on-line evaluation of residual quantities that are directly applicable in monitoring not only the process but also the magnitude of error in the process model upon which the controller is based.

This work begins with the application of nonlinear controller synthesis methods to nonlinear first-order models, which results in nonlinear PI controllers. A similar development, starting with nonlinear second-order models and resulting in nonlinear PID controllers, is then presented. Issues arising at the implementation stage, including process monitoring, on-line measures of model error, and example applications to chemical engineering processes are then discussed. Finally, a CSTR with a nonelementary exothermic reaction is used to demonstrate the closed-loop performance of the proposed nonlinear controllers.

Model-Based Synthesis of Nonlinear PI Controllers

In developing general nonlinear controller synthesis algorithms, it is common to begin with a general representation of nonlinear systems. A standard control-affine state-space representation is generally used, and several results are available in the literature for the application of nonlinear model-based controller synthesis methods on models of this form (Isidori and Ruberti, 1984; Kravaris and Kantor, 1990; Bequette, 1991; Kravaris and Arkun, 1991; Allgöwer and Doyle, 1996). However, an interpretation of these general results in terms of controller structures that are more commonly used in practice is desirable.

In synthesizing nonlinear PI controllers from a nonlinear first-order dynamic model, it is convenient to express the model in terms of a single equation, where the scalar output y is the dependent variable. It is further helpful to use a model where the coefficient of the scalar input u is equal to 1. In this work, first-order models of the form

$$M(y)\frac{dy}{dt} + N(y) = u \qquad (3)$$

will be used as a starting point. It is important to note that all nonlinear control-affine first-order models with well-defined relative order can be put in the form of Eq. 3. In this representation, the steady-state behavior of the process is captured entirely by $N(y)$, and the function $M(y)$ is related solely to the dynamics of the process. These two functions will prove useful in later sections, where using the model for on-line process monitoring is discussed. For the special case where $M(y) = \tau/k$ and $N(y) = y/k$, the first-order model becomes the standard linear first-order model expressed in terms of a static gain, k, and a time constant, τ, which is used as the basis for deriving linear PI controllers in Rivera et al. (1986).

Consider the nonlinear first-order process described by Eq. 3 and the problem of synthesizing a feedback controller with the following specifications:

(1) Controller must be first order;
(2) Controller must possess integral action;
(3) Controller must induce the *linear* closed-loop dynamics

$$\tau_{cl}\frac{dy}{dt} + y = y_{sp}, \qquad (4)$$

where τ_{cl} is the time constant of the closed-loop system (tunable parameter).

A general nonlinear controller synthesis method, such as globally linearizing control (GLC) (Kravaris and Chung, 1987; Daoutidis and Kravaris, 1992) can be used for this purpose. When an open-loop state observer is used in the GLC structure, the resulting controller reduces to a model state-feedback controller (Kravaris et al., 1994, 1998). When applied to Eq. 3, this synthesis method results in the following controller that meets the specifications just given (see the Appendix for derivation)

$$\frac{d\hat{y}}{dt} = \frac{e}{\tau_{cl}}$$

$$u = M(\hat{y})\frac{e}{\tau_{cl}} + N(\hat{y}), \qquad (5)$$

where $e = y_{sp} - y$ is the error. The state of the controller $\hat{y}$ is the integral of the error scaled by the closed-loop time constant τ_{cl}. The controller given by Eq. 5 thus has integral action, and since the error appears explicitly in the calculation of the manipulated input, it also has proportional action. Since the only input in Eq. 5 is the error signal, this controller is a one-degree-of-freedom nonlinear PI controller.

Before proceeding, it is important to note two other aspects of the state of the controller (Eq. 5). First, from Eq. 4, the desirable closed-loop input–output behavior is

$$\frac{dy}{dt} = \frac{y_{sp} - y}{\tau_{cl}}. \tag{6}$$

Thus, the controller state $\hat{y}$ represents the precalculated evolution of the output in closed-loop. Additionally, the control law (Eq. 5) forces $\hat{y}$ to match with y so that the closed-loop dynamics follows Eq. 4. Notice that this happens after cancellation at the process modes. Indeed, combining Eq. 3 with Eq. 5 results in

$$M(y)\frac{dy}{dt} + N(y) = M(\hat{y})\frac{d\hat{y}}{dt} + N(\hat{y}), \tag{7}$$

from which y will asymptotically approach $\hat{y}$, with the speed of the dynamics Eq. 3, as long as the dynamics (Eq. 3) is stable.

The application of the same nonlinear controller synthesis method to a linear first-order process in standard form results in the linear controller

$$\frac{\bar{u}(s)}{\bar{e}(s)} = \frac{\tau}{k\tau_{cl}}\left(1 + \frac{1}{\tau s}\right), \tag{8}$$

when put in transfer-function form. By comparing Eq. 8 with Eq. 1, it can clearly be seen that this is a standard linear one-degree-of-freedom PI controller with gain $k_c = \tau/k\tau_{cl}$ and reset time $\tau_I = \tau$. The resulting controller, Eq. 8, is identical to the one derived by Rivera et al. (1986) using the IMC method.

If it is desired that the location of the zero-pole cancellation be arbitrarily assigned instead of having it at the process mode, an alternative solution to the posed synthesis problem can be obtained. This can be accomplished using the GLC synthesis method (Kravaris and Chung, 1987; Daoutidis and Kravaris, 1992) and results in the following controller (see the Appendix for derivation)

$$\frac{d\hat{y}}{dt} = \frac{e}{\tau_{cl}}$$

$$u = M(y)\left[\frac{e}{\tau_{cl}} + \frac{\hat{y} - y}{\tau_c}\right] + N(y), \tag{9}$$

which has two tunable parameters, τ_{cl} and τ_c. The state of the controller represented by Eq. 9 is exactly the same as the state of the controller represented by Eq. 5; it is therefore proportional to the integral of the error and represents the

precalculated evolution of the output that the controller is trying to enforce. The controller clearly has both integral and proportional action. Notice, however, that both the error and the measured output directly enter the calculation of the manipulated input. Therefore, this control law is mixed error-output feedback, or a two-degree-of-freedom nonlinear PI controller.

With the controller given by Eq. 9, the closed-loop system follows the dynamics

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$\frac{dy}{dt} = \frac{y_{sp} - y}{\tau_{cl}} + \frac{\hat{y} - y}{\tau_c}, \tag{10}$$

from which

$$\frac{d}{dt}(\hat{y} - y) = -\frac{1}{\tau_c}(\hat{y} - y), \tag{11}$$

and therefore $\hat{y}$ will asymptotically approach y with time constant τ_c. Thus, the tunable parameter τ_c measures the "hidden dynamics" of the closed-loop system, or, stated another way, the closed-loop system has a zero-pole cancellation at $-1/\tau_c$.

The application of the same nonlinear controller synthesis method to a linear first-order process in standard form results in the linear controller

$$\bar{u}(s) = \frac{\tau}{k\tau_{cl}}\left(1 + \frac{1}{\tau_c s}\right)\bar{e}(s) - \frac{\tau}{k}\left(\frac{1}{\tau_c} - \frac{1}{\tau}\right)\bar{y}(s), \tag{12}$$

when put in transfer function form. This would be exactly the same result as if one were to design a controller via polynomial equations and solve a Diophantine equation (Kailath, 1980) for the given specifications of closed-loop input–output behavior and zero-pole cancellation. The two-degree-of-freedom controller has proportional and integral action in the error, but only proportional action in y. When the process is unstable, the P-action in y can be interpreted as a prestabilizing inner loop that shifts the pole to $-1/\tau_c$, and the PI-action in the error as an outer loop that enforces the desired trajectory. Notice also that if τ_c is selected to equal τ, the controller (Eq. 12) becomes identically equal to that of Eq. 8.

It has been shown in this section that if a general nonlinear controller synthesis method is applied to a nonlinear first-order process model, a nonlinear PI controller will naturally result. A slight variation of the method can yield either a one-degree-of-freedom controller, where a zero-pole cancellation will occur at the process mode, or a two-degree-of-freedom controller, where the zero-pole cancellation can be arbitrarily assigned. The closed-loop time constant is a tunable parameter, which may be selected in a manner consistent with standard performance/robustness trade-offs. The model parameters, or nonlinear functions, will automatically appear in the control law in a physically meaningful way. The resulting nonlinear controllers are directly linked to the pro-

cess, yet are easily understandable in the context of standard linear PI controllers and can be implemented within the same framework. These points and the applicability of the controller quantities for process monitoring will be discussed in the context of more specific chemical engineering processes in a later section.

Model-Based Synthesis of Nonlinear PID Controllers

While nonlinear PI controllers arise naturally from nonlinear first-order process models, in this section it will be shown that nonlinear PID controllers arise naturally from nonlinear second-order process models of relative order 2. A general second-order nonlinear control-affine model of relative order 2 can always be expressed in terms of a second-order differential equation of the form

$$M\left(y, \frac{dy}{dt}\right)\frac{d^2y}{dt^2} + N\left(y, \frac{dy}{dt}\right) = u \qquad (13)$$

The functions $M(y,(dy/dt))$ and $N(y,(dy/dt))$ can most easily be calculated by putting the system in normal form (Isidori, 1989; Nijmeijer and van der Schaft, 1990) through a coordinate transformation, converting the normal form into a second-order differential equation, and rearranging the coefficients. Notice that the steady-state behavior is captured by $N(y, 0)$ and that the part of the model described by $M(y(dy/dt))$ is solely related to the dynamics of the process. For the special case where $M(y,(dy/dt)) = \tau^2/k$ and

$$N\left(y, \frac{dy}{dt}\right) = \frac{y}{k} + \frac{2\zeta\tau}{k}\frac{dy}{dt},$$

the second-order model becomes the standard linear second-order model expressed in terms of a static gain k, a time constant τ, and a damping factor ζ, which is used as a basis for deriving linear PID controllers in Rivera et al. (1986).

Consider the nonlinear second-order process described by Eq. 13 and the problem of synthesizing a feedback controller with the following specifications:
(1) Controller must be second order.
(2) Controller must possess integral action.
(3) Controller must induce the *linear* closed-loop dynamics

$$\tau_{cl}\tau_{cl}^*\frac{d^2y}{dt^2} + (\tau_{cl} + \tau_{cl}^*)\frac{dy}{dt} + y = y_{sp} \qquad (14)$$

where τ_{cl} and τ_{cl}^* are the closed-loop time constants (tunable parameters). In the limit as $\tau_{cl}^* \to 0$, the closed-loop dynamics in Eq. 14 will approach the one described by Eq. 4. This limiting case will also be considered and will lead to ideal derivative action.

A solution to the posed problem can be derived using the GLC synthesis method in the form of the model state-feedback structure (Kravaris et al., 1994, 1998). When applied to Eq. 13, this structure results in the following controller (see the Appendix for derivation)

$$\frac{d\hat{y}}{dt} = \hat{s}$$

$$\frac{\tau_{cl}\tau_{cl}^*}{\tau_{cl} + \tau_{cl}^*}\frac{d\hat{s}}{dt} = \frac{e}{\tau_{cl} + \tau_{cl}^*} - \hat{s} \qquad (15)$$

$$u = M(\hat{y},\hat{s})\frac{\dfrac{e}{\tau_{cl} + \tau_{cl}^*} - \hat{s}}{\dfrac{\tau_{cl}\tau_{cl}^*}{\tau_{cl} + \tau_{cl}^*}} + N(\hat{y},\hat{s}).$$

One state of the controller $\hat{s}$ represents the result of filtering the time-scaled error $e/(\tau_{cl} + \tau_{cl}^*)$ with filter time constant $(\tau_{cl}\tau_{cl}^*)/(\tau_{cl} + \tau_{cl}^*)$. The other state of the controller $\hat{y}$ is exactly the integral of $\hat{s}$. Therefore, $\hat{y}$ is the integral of the filtered error scaled by the sum of the closed-loop time constants $\tau_{cl} + \tau_{cl}^*$. Furthermore, from Eq. 14, the desirable closed-loop dynamics of the output follows

$$\frac{dy}{dt} = s$$

$$\frac{\tau_{cl}\tau_{cl}^*}{\tau_{cl} + \tau_{cl}^*}\frac{ds}{dt} = \frac{y_{sp} - y}{\tau_{cl} + \tau_{cl}^*} - s. \qquad (16)$$

Thus, the controller states $\hat{y}$ and $\hat{s}$ represent the precalculated evolution of the output y and its derivative $s = (dy/dt)$, which the controller is trying to enforce. The controller given by Eq. 15 has integral and derivative action, and since the error appears explicitly in the calculation of the manipulated input, it also has proportional action. The only input into Eq. 15 is the error signal, therefore this controller is a one-degree-of-freedom nonlinear PID controller.

It is interesting to examine the limiting case $\tau_{cl}^* \to 0$. In this case, the time constant of the second-state equation (Eq. 15) $(\tau_{cl}\tau_{cl}^*)/(\tau_{cl} + \tau_{cl}^*) \to 0$, and therefore $\hat{s} \to e/\tau_{cl}$. Moreover,

$$\left(\frac{e}{\tau_{cl} + \tau_{cl}^*} - \hat{s}\right)\Bigg/\left(\frac{\tau_{cl}\tau_{cl}^*}{\tau_{cl} + \tau_{cl}^*}\right) = \frac{d\hat{s}}{dt} \to \frac{1}{\tau_{cl}}\frac{de}{dt},$$

and the controller (Eq. 15) tends to

$$\frac{d\hat{y}}{dt} = \frac{e}{\tau_{cl}} \qquad (17)$$

$$u = M\left(\hat{y}, \frac{e}{\tau_{cl}}\right)\frac{1}{\tau_{cl}}\frac{de}{dt} + N\left(\hat{y}, \frac{e}{\tau_{cl}}\right),$$

which is an ideal one-degree-of-freedom nonlinear PID controller, as opposed to Eq. 15, which has filtered derivative action.

Note that with the controller represented in Eq. 15 or Eq. 17, the requested closed-loop input/output behavior in Eq. 14 or Eq. 4 is obtained after cancellations at the process

modes. For example, combining Eq. 17 with Eq. 13 results in

$$M\left(y,\frac{dy}{dt}\right)\frac{d^2y}{dt^2} + N\left(y,\frac{dy}{dt}\right) = M\left(\hat{y},\frac{d\hat{y}}{dt}\right)\frac{d^2\hat{y}}{dt^2}$$
$$+ N\left(\hat{y},\frac{d\hat{y}}{dt}\right), \quad (18)$$

from which y will asymptotically approach $\hat{y}$, with the speed of dynamics of Eq. 13, as long as Eq. 13 is stable.

The application of the same nonlinear controller synthesis method to a linear second-order process in standard form results in the controller

$$\frac{\bar{u}(s)}{\bar{e}(s)} = \frac{2\zeta\tau}{k\tau_{cl}}\left(1 + \frac{1}{2\zeta\tau s} + \frac{\tau}{2\zeta}s\right) \quad (19)$$

when put in transfer-function form. By comparing Eq. 19 with Eq. 2 it can clearly be seen that this is a standard linear one-degree-of-freedom PID controller with gain $k_c = (2\zeta\tau/k\tau_{cl})$, reset time $\tau_I = 2\zeta\tau$, and rate time $\tau_D = \tau/2\zeta$. The resulting controller (Eq. 19) is identical to the one derived by Rivera et al. (1986) using the IMC method.

If it is desired to arbitrarily assign the location of the zero-pole cancellations instead of having them at the process modes, an alternative solution to the posed synthesis problem can be obtained. This can be accomplished using the GLC synthesis method (Kravaris and Chung, 1987; Daoutidis and Kravaris, 1992) and results in the following controller (see the Appendix for derivation)

$$\frac{d\hat{y}}{dt} = \frac{e}{\tau_{cl}} \quad (20)$$

$$u = M\left(y,\frac{dy}{dt}\right)\left\{\frac{1}{\tau_{cl}}\frac{de}{dt} + \frac{\tau_{ca}+\tau_{cb}}{\tau_{ca}\tau_{cb}}\left(\frac{e}{\tau_{cl}} - \frac{dy}{dt}\right) + \frac{\hat{y}-y}{\tau_{ca}\tau_{cb}}\right\}$$
$$+ N\left(y,\frac{dy}{dt}\right),$$

with tunable parameters τ_{cl}, τ_{ca}, and τ_{cb}. Comparing Eq. 20 to Eq. 17, it can clearly be seen that both controllers have ideal derivative action and the same state $\hat{y}$, which is a time-scaled integral of the error. The manipulated input calculated in Eq. 20 depends directly on both the error and measured output. This controller is clearly mixed error-output feedback, or a two-degree-of-freedom nonlinear PID controller. However, control action is calculated differently in Eq. 20 than it is in Eq. 17, and this impacts the behavior of the closed-loop system. With the controller (Eq. 20) applied to Eq. 13, the closed-loop system follows the dynamics

$$\frac{d\hat{y}}{dt} = \frac{y_{sp}-y}{\tau_{cl}} \quad (21)$$

$$\frac{d^2y}{dt^2} = \frac{1}{\tau_{cl}}\frac{d(y_{sp}-y)}{dt} + \frac{\tau_{ca}+\tau_{cb}}{\tau_{ca}\tau_{cb}}\left(\frac{y_{sp}-y}{\tau_{cl}} - \frac{dy}{dt}\right) + \frac{\hat{y}-y}{\tau_{ca}\tau_{cb}}$$

Substituting $(y_{sp}-y)/\tau_{cl}$ by $d\hat{y}/dt$ and $1/\tau_{cl}[d(y_{sp}-y)]/dt)$ by $d^2\hat{y}/dt^2$ in Eq. 21, it follows that

$$\frac{d^2(\hat{y}-y)}{dt^2} + \frac{\tau_{ca}+\tau_{cb}}{\tau_{ca}\tau_{cb}}\frac{d(\hat{y}-y)}{dt} + \frac{\hat{y}-y}{\tau_{ca}\tau_{cb}} = 0, \quad (22)$$

and therefore $\hat{y}$ will asymptotically approach y with time constants τ_{ca} and τ_{cb}. Thus, the tunable parameters τ_{ca} and τ_{cb} account for the "hidden dynamics" of the closed-loop system; the closed-loop system has zero-pole cancellations at $-1/\tau_{ca}$ and $-1/\tau_{cb}$.

The application of the same nonlinear controller synthesis method to a linear second-order process in standard form results in the linear controller

$$\bar{u}(s) = \frac{\tau^2}{k\tau_{cl}}\left(\frac{\tau_{ca}+\tau_{cb}}{\tau_{ca}\tau_{cb}} + \frac{1}{\tau_{ca}\tau_{cb}s} + s\right)\bar{e}(s)$$
$$- \frac{\tau^2}{k}\left(\frac{1}{\tau_{ca}\tau_{cb}} - \frac{1}{\tau^2} + \left[\frac{\tau_{ca}+\tau_{cb}}{\tau_{ca}\tau_{cb}} - \frac{2\zeta}{\tau}\right]s\right)\bar{y}(s) \quad (23)$$

when put in transfer function form. This would be exactly the same result as if one were to design a controller via polynomial equations and solve a Diophantine equation (Kailath, 1980) for the given specifications of closed-loop input–output behavior and zero-pole cancellations. The two-degree-of-freedom controller has proportional, integral, and derivative action in the error, but only proportional and derivative action in y. When the process is unstable, the PD action in y can be interpreted as a prestabilizing inner loop that shifts the poles to $-1/\tau_{ca}$ and $-1/\tau_{cb}$ and the PID action in the error as an outer loop that enforces the desired trajectory. Notice also, that if τ_{ca} and τ_{cb} are chosen so that $\tau_{ca}\tau_{cb} = \tau^2$ and $\tau_{ca} + \tau_{cb} = 2\zeta\tau$, the controller (Eq. 23) becomes identically equal to Eq. 19.

It has been shown in this section that if a general nonlinear controller synthesis method is applied to a nonlinear second-order process model of relative order 2, a nonlinear PID controller will result. A slight variation on the method can yield either a one-degree-of-freedom controller, where the zero-pole cancellations will occur at the process modes, or a two-degree-of-freedom controller, where the zero-pole cancellations may be arbitrarily assigned. The closed-loop time constant is a tunable parameter, which can be selected in a manner consistent with standard performance/robustness trade-offs. The model parameters, or nonlinear functions, will automatically appear in the control law in a physically meaningful way. The resulting nonlinear controllers are directly linked to the process, yet easily understandable in the context of standard linear PID controllers and can be implemented within the same framework. These points and the applicability of the controller quantities for process monitoring will be discussed in the next section in the context of more specific chemical engineering processes.

A Practical Interpretation of Results and Examples of Applications to Chemical Processes

As with all model-based controllers, their performance in the presence of modeling error, or in other words how accurate the models need to be for good closed-loop performance

is naturally a concern. Local robustness results are available in the literature (Kravaris and Palanki, 1988; Byrnes et al., 1997) for general nonlinear input–output linearization methods. Since these methods were used in deriving the controllers in the previous sections, the controllers obviously inherit these results. From the specific form of these controllers, however, it is straightforward to see how modeling error will be rejected. Relevant quantities that can be used on-line in a simple process-monitoring scheme, or as a measure of the accuracy of the model upon which the controller is based, are inherent in the controller structure.

Consider the case of a first-order nonlinear process, which is not exactly known, and is represented by

$$M_p(y)\frac{dy}{dt} + N_p(y) = u \qquad (24)$$

$$M_m(y)\frac{dy}{dt} + N_m(y) = u, \qquad (25)$$

where the subscript p denotes the true process and the subscript m denotes the model of the process. The closed-loop system under the one-degree-of-freedom nonlinear PI controller (Eq. 5) is

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$\frac{dy}{dt} = \frac{M_m(\hat{y})}{M_p(y)}\frac{y_{sp} - y}{\tau_{cl}} + \frac{N_m(\hat{y}) - N_p(y)}{M_p(y)}. \qquad (26)$$

When the closed-loop system is at steady state, the following relations will hold

$$y_s = y_{sp}$$

$$u_s = N_m(\hat{y}_s) = N_p(y_{sp}), \qquad (27)$$

where the subscript s denotes a steady-state value. A few key quantities can be used to quantify the difference between the model and the process or whether an unmeasured disturbance has entered the process. The state of all the controllers developed in this work $\hat{y}$ is also an estimate of the measured output y. In the absence of modeling error, $\hat{y}$ will track y. The difference $\hat{y} - y$ is a measure of the error in terms of the process output. Tracking the difference $N_m(y) - N_m(\hat{y})$ provides useful information in quantifying modeling error in terms of the process input. At steady state, the controller forces $N_m(\hat{y})$ to equal u, and the difference between these quantities and $N_m(y)$ is exactly the model error in the steady-state part of the process model.

If the two-degree-of-freedom nonlinear PI controller (Eq. 9) is used, the closed-loop system is

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$\frac{dy}{dt} = \frac{M_m(y)}{M_p(y)}\left[\frac{y_{sp} - y}{\tau_{cl}} + \frac{\hat{y} - y}{\tau_c}\right] + \frac{N_m(y) - N_p(y)}{M_p(y)}. \qquad (28)$$

When the closed-loop system is at steady state, the following relations will hold

$$y_s = y_{sp}$$

$$u_s = N_m(y_s) + M_m(y_s)\left[\frac{\hat{y}_s - y_s}{\tau_c}\right] = N_p(y_s). \qquad (29)$$

In this case, since the location of the zero-pole cancellation is tunable, the speed with which modeling error is rejected, as well as the difference $\hat{y} - y$, will be different than the one-degree-of-freedom controller case. The state of the controller is still an estimate of the measured output, but the difference $\hat{y} - y$ will depend on τ_c. Tracking $N_m(y)$ with u will provide exactly the same useful information in quantifying modeling error as it did with the one-degree-of-freedom controller. The difference $N_m(y_s) - u_s$ is exactly the modeling error in the steady-state part of the model.

This discussion has focused entirely on first-order systems and the use of nonlinear PI controllers. The analysis for second-order systems and nonlinear PID controllers is essentially the same and will be omitted here for brevity. The quantities that are useful to track on-line will be the same, only using $N(\cdot, 0)$ in place of $N(\cdot)$. The use of these quantities will be demonstrated for PID controllers in the simulated example in the next section.

Another concern with the implementation of model-based controllers is the degree to which controller code can be standardized to allow for minimum long-term maintenance costs. The form in which controllers were derived in this work is a good form for direct implementation. The calculation of the state variable is straightforward and very similar to the calculation of the integral in a standard linear PI or PID controller. The calculation of the manipulated input remains the same for each controller, in terms of the M and N functions. These two functions are all that vary as the model is changed. Visualizing these two functions as subroutines means that changing a model is as simple as replacing a subroutine. Moreover, the rest of the controller will be very similar to the standard linear controllers that are usually prevalent in industrial plants. Finally, the M and N functions are directly tied to a model of the process, which can be included in the documentation for the controller. These points will be further illustrated in a series of examples.

Example 1. Consider a batch reactor with negligible jacket dynamics and an irreversible, zero-order reaction, $A \rightarrow B$. An energy balance around the reactor yields

$$\frac{dT}{dt} = \frac{UA}{\rho c_p V}(T_j - T) + \frac{k_0 e^{-E/RT}(-\Delta H)}{\rho c_p V}. \qquad (30)$$

The objective is to control the reactor temperature, T, by manipulating the jacket temperature, T_j. Either the one-degree-of-freedom nonlinear PI (Eq. 5), or the two-degree-of-freedom nonlinear PI (Eq. 9) can be applied. For this exam-

ple, Eq. 9 will be applied to Eq. 30, resulting in

$$\frac{d\hat{T}}{dt} = \frac{T_{sp} - T}{\tau_{cl}}$$

$$M(T) = \frac{\rho C_p V}{UA}$$

$$N(T) = T - \frac{k_0 e^{-E/RT}(-\Delta H)}{UA}$$

$$T_j = M(T)\left(\frac{T_{sp} - T}{\tau_{cl}} + \frac{\hat{T} - T}{\tau_c}\right) + N(T). \qquad (31)$$

For this example, $M(T)$ is a constant with units of time, and corresponds to what the time constant of the process would be if there were no reaction taking place. As stated previously, the function $N(T)$ corresponds to the steady-state value of the manipulated input, T_j, and contains the only nonlinearity of the process. It would be relatively easy to verify both these functions in practice. The first and fourth parts of these controllers are not affected by changing the model, as long as the order of the model is preserved.

Implementation in the form given by Eq. 31 also results in the quantities needed for on-line monitoring of the process and model error. The state of the controller would obviously be available and could easily be compared to the measured output. For this two-degree-of-freedom controller, $N(T)$ is calculated in determining the control action and can be compared on-line to T_j as a measure of model error at steady state.

Example 2. Now consider the same batch reactor and reaction as in Example 1, but with jacket dynamics included:

$$\frac{dT}{dt} = \alpha_1(T_j - T) + r(T) \qquad (32)$$

$$\frac{dT_j}{dt} = \alpha_3(T - T_j) + \alpha_2(T_{ij} - T_j).$$

The substitutions

$$\alpha_1 = \frac{UA}{\rho c_p V}, \qquad \alpha_2 = \frac{F_j \rho_j}{m_j}, \qquad \alpha_3 = \frac{UA}{m_j c_{p_j}}$$

$$r(T) = \frac{k_0 e^{-E/RT}(-\Delta H)}{\rho c_p V}$$

have been made to simplify the development of the controller. By eliminating the state variable T_j, the system given by Eq. 32 can be put into the input−output form realized by the following second-order differential equation:

$$\frac{d^2 T}{dt^2} + \left[\alpha_1 + \alpha_2 + \alpha_3 - \frac{dr}{dT}(T)\right]\frac{dT}{dt}$$

$$-(\alpha_2 + \alpha_3)r(T) + \alpha_1 \alpha_2(T - T_{ji}) = 0. \qquad (33)$$

The objective now is to control the reactor temperature, T, by manipulating the jacket inlet temperature, T_{ji}. Either a one-degree-of-freedom nonlinear PID controller or a two-degree-of-freedom nonlinear PID controller can be applied to this process. For this example, the one-degree-of-freedom nonlinear ideal PID controller (Eq. 17) will be applied, to Eq. 33, resulting in

$$\frac{d\hat{T}}{dt} = \frac{T_{sp} - T}{\tau_{cl}}$$

$$M\left(\hat{T}, \frac{d\hat{T}}{dt}\right) = \frac{1}{\alpha_1 \alpha_2}$$

$$N\left(\hat{T}, \frac{d\hat{T}}{dt}\right) = \left[\frac{1}{\alpha_1} + \frac{1}{\alpha_2} + \frac{\alpha_3}{\alpha_1 \alpha_2} - \frac{1}{\alpha_1 \alpha_2}\frac{dr}{d\hat{T}}(\hat{T})\right]\frac{d\hat{T}}{dt}$$

$$-\left(\frac{1}{\alpha_1} + \frac{\alpha_3}{\alpha_1 \alpha_2}\right)r(\hat{T}) + \hat{T}$$

$$T_{ji} = M\left(\hat{T}, \frac{d\hat{T}}{dt}\right)\frac{1}{\tau_{cl}}\frac{d[T_{sp} - T]}{dt} + N\left(\hat{T}, \frac{d\hat{T}}{dt}\right) \qquad (34)$$

Again, in this example, M is a constant. Here it has units of time squared and is related to time constants for certain parts of the process. The function $N(T,0)$ corresponds to the steady-state value of T_{ji}. The nonlinearity of the process is reflected in the controller exclusively through N. As would be expected for a second-order system of relative order 2, both the reaction rate and the derivative of the reaction rate are important for controlling the process. Notice that the first and fourth parts of these controllers are not affected by changing the model, as long as the order of the model is preserved. In Eq. 34 a derivative of the error $T_{sp} - T$ is required in the controller. This is analogous to the derivative, which is required in the standard linear PID controller (Eq. 2), and can be implemented in a similar manner.

As with the nonlinear PI controller, implementation of the nonlinear PID controllers in the form given by Eq. 34 results in the quantities needed for on-line monitoring of the process and model error. The state of the controller is available and could easily be compared to the measured output. In the one-degree-of-freedom controller, $N(\hat{T},0)$ can be obtained from the controller calculations, and it would be one simple additional calculation to have $N(T,0)$ available on-line as well. These quantities can be tracked along with T_{ji} for detection and estimation of steady-state modeling error.

Simulation Example

The examples of the previous section illustrated the possible forms of nonlinear PI and PID controllers for some chemical systems where the application of the controllers was relatively straightforward. Since the systems actually were first- or second-order processes, simulation results for these examples are predictable. In this section, a more complex chemical system is used to illustrate model order reduction and application of the proposed controllers. Simulation re-

sults are presented to demonstrate the resulting closed-loop performance.

Consider a nonisothermal CSTR with the following homogeneous chemical reaction taking place inside the reactor

$$A \rightarrow B.$$

It is assumed that the preceding chemical reaction is taking place in the liquid phase. Moreover, it is considered to be nonelementary, with the overall reaction following the kinetic rate law
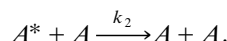
$$r = r_B = \frac{K(T)C_A}{1 + K'(T)C_A}. \tag{35}$$

This rate law is obtained when the reaction mechanism consists of the following elementary reaction steps (Fogler, 1992)

(1) An intermediate species A^* is formed from A

$$A \xrightarrow{k_1} A^*.$$
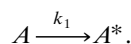
(2) The resulting active intermediate A^* collides with A

$$A^* + A \xrightarrow{k_2} A + A.$$

(3) Or it spontaneously decomposes to form B

$$A^* \xrightarrow{k_3} B.$$
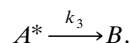
The active intermediate A^* has a very short lifetime because of its high reactivity and is considered to be present in low concentrations. The preceding conditions validate the pseudo-steady-state hypothesis, which states that the total rate of A^* should be equal to zero. Under these assumptions, the overall rate of formation of B, Eq. 35, results.

The controlled output is the reactor temperature and it is assumed that the jacket temperature can be directly manipulated. Under standard assumptions, the full-order dynamic model of the CSTR is

$$\frac{dC_A}{dt} = \frac{F}{V}(C_{A,\text{in}} - C_A) - \frac{K(T)C_A}{1 + K'(T)C_A} \tag{36}$$

$$\frac{dT}{dt} = \frac{F}{V}(T_{\text{in}} - T) + \frac{(-\Delta H)}{\rho c_p}\frac{K(T)C_A}{1 + K'(T)C_A} - \frac{UA(T - T_j)}{\rho c_p V}, \tag{37}$$

where

$$K(T) = k_0 \exp\left(\frac{-E}{RT}\right) \tag{38}$$

$$K'(T) = k'_0 \exp\left(\frac{-E'}{RT}\right) \tag{39}$$

The preceding process model, given by Eqs. 36–39, is clearly second order. However, in particular regions of opera-

tion, simplifications can be made that naturally lead to a first-order dynamic process model. At high concentrations of the reactant A, where $K'C_A \gg 1$, the overall kinetic rate law (Eq. 35) simplifies to:

$$r \approx \frac{K(T)C_A}{K'(T)C_A} = \frac{K(T)}{K'(T)}. \tag{40}$$

Thus, under these conditions, the apparent reaction-order is zero (Fogler, 1992). The dynamic process model for the CSTR can then be approximated by the first-order model

$$\frac{dT}{dt} = \frac{F}{V}(T_{in} - T) + \frac{(-\Delta H)}{\rho c_p}\frac{K(T)}{K'(T)} - \frac{UA(T - T_j)}{\rho c_p V} \tag{41}$$

along with Eqs. 38 and 39. This model can then be used to synthesize PI controllers as long as the CSTR operates in a region where the approximation given by Eq. 40 is valid. Simulations of one- and two-degree-of-freedom nonlinear PI controllers have been performed, and the results are available from the authors. Presentation of these results has been omitted from this article in favor of the results that follow.

If the jacket temperature cannot be directly manipulated, then the dynamics of a heating/cooling system must also be considered. If the reactor is a jacketed vessel, a jacket energy balance can be included

$$\frac{dT_j}{dt} = \frac{UA}{m_j c_{p_j}}(T - T_j) + \frac{F_j \rho_j}{m_j}(T_{ji} - T_j), \tag{42}$$

Using the simplification made to the reaction rate, an approximate model for the system with jacket dynamics is then given by Eqs. 41 and 42. This system is second order with relative order two, and can be used as the basis to synthesize PID controllers. Eliminating the state T_j from Eqs. 41 and 42, a second-order ODE is obtained that represents the input−output behavior of the process. It is of the form of Eq. 13 with

$$M\left(T, \frac{dT}{dt}\right) = \frac{\rho c_p V}{UA}\frac{m_j}{F_j \rho_j} \tag{43}$$

$$N\left(T, \frac{dT}{dt}\right) = \left\{ \frac{m_j}{F_j \rho_j} + \frac{\rho c_p V}{F_j \rho_j c_{p_j}} + \frac{\rho c_p V}{UA} + \frac{m_j}{F_j \rho_j}\frac{\rho c_p F}{UA} \right. $$
$$\left. - \frac{m_j}{F_j \rho_j}\frac{(-\Delta H)V}{UA}\frac{K(T)}{K'(T)}\left[\frac{E - E'}{RT^2}\right] \right\}\frac{dT}{dt} + T$$

$$- \left[\frac{1}{F_j \rho_j c_{p_j}} + \frac{1}{UA}\right](\rho c_p F)(T_{\text{in}} - T) - \left[\frac{1}{F_j \rho_j c_{p_j}} + \frac{1}{UA}\right]$$

$$\times (-\Delta H)V\frac{K(T)}{K'(T)} \tag{44}$$

The operating region over which Eq. 41 is a reasonable approximation to Eqs. 36 and 37 will depend on the values of the parameters in the models. For the purpose of this simula-

Table 1. Parameter Values for the Example Process

$V = 2\ m^3$	$-\Delta H = 40{,}000\ kJ/kmol$
$F = 0.0004\ m^3/s$	$R = 8.345\ kJ/(kmol \cdot K)$
$T_{in} = 295\ K$	$E = 50{,}000\ kJ/kmol$
$C_{A,in} = 10\ kmol/m^3$	$E' = 5{,}000\ kJ/kmol$
$UA = 3\ kJ/(s \cdot K)$	$K_0 = 6 \times 10^6\ s^{-1}$
$c_p = 4.2\ kJ/(kg \cdot K)$	$K_0' = 1200\ m^3/kmol$
$\rho = 1{,}000\ kg/m^3$	$m_i = 378.35\ kg$
$c_{pi} = 3.2\ kJ/(kg \cdot K)$	$F_j = 0.001\ m^3/s$
$\rho_i = 700\ kg/m^3$	

tion study, the parameter values given in Table 1 are used. The steady-state input–output behavior of both the true system (Eqs. 36–37, 42) and the approximate system (Eqs. 41 and 42) is shown in Figure 1. The approximate system tracks the true system well, until the reactor temperature approaches 365 K, at which point it diverges. With the jacket dynamics included, the reactor has three steady states, with an unstable middle steady state (see Russo and Bequette, 1995, for a dynamic analysis of jacketed CSTRs exhibiting nonlinear behavior). The one-degree-of-freedom PID controllers are necessarily restricted to operation at the lower-temperature steady state, which covers conversion up to approximately 60%. Naturally, in this region the nonlinear behavior of the process will be less pronounced. In all the simulations that follow, ideal derivatives of the error and/or output were used. In practice, a filter or approximation of the derivative would normally be used. Exactly the same ideas can be applied to the derivative modes in the linear and nonlinear controllers. For the sake of brevity and simplicity, these approximations are not used. Instead, to avoid a large spike of derivative action when step changes in the set point are introduced, only ramp changes in the setpoint will be used.

The nonlinear PID controllers presented in this work are synthesized using Eqs. 43 and 44. For comparison purposes, simulations with appropriate linear PID controllers are also shown. The controllers are based on the linearization of the model of Eqs. 41 and 42 around an operating steady state. This is of the form of a standard linear second-order model,
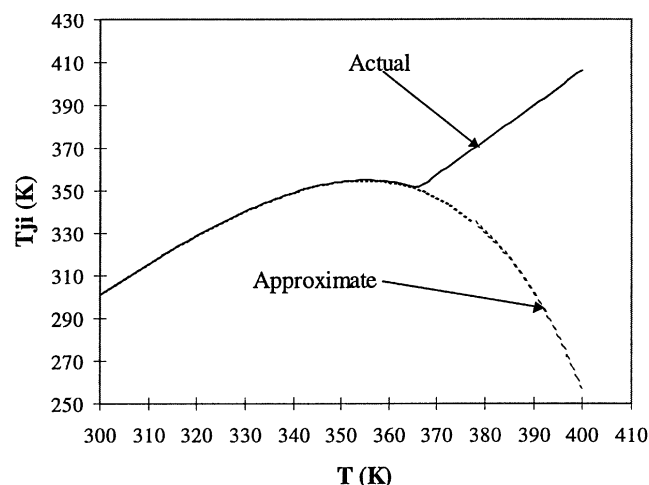

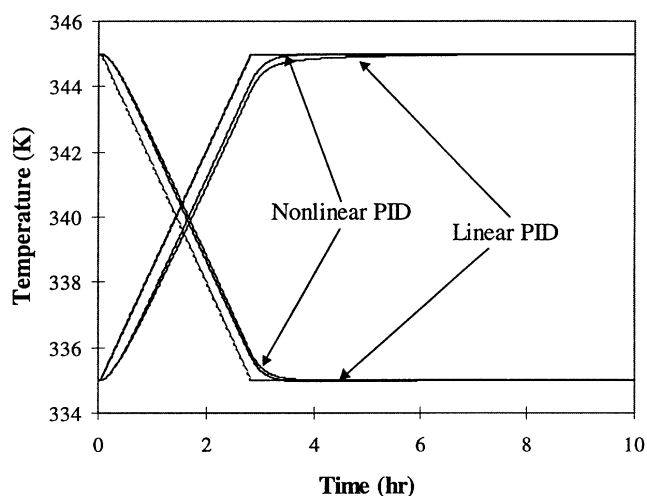Figure 1. Steady-state behavior of actual and approximate systems.

Figure 2. Response to changes in setpoint, both increase and decrease, for one degree-of-freedom linear and nonlinear PID controllers.

where the static gain, time constant, and damping factor are calculated from the following relations

$$\frac{\tau^2}{k} = M(T_s, 0) \tag{45}$$

$$\frac{2\zeta\tau}{k} = \frac{\partial N}{\partial\left(\dfrac{dT}{dt}\right)}(T_s, 0) \tag{46}$$

$$\frac{1}{k} = \frac{\partial N}{\partial T}(T_s, 0), \tag{47}$$

where T_s is the reactor temperature around which the system is being linearized.

The results using one-degree-of-freedom linear PID and nonlinear PID controllers are shown in Figure 2. The responses of an increase in setpoint, from 335 K to 345 K, and a decrease in setpoint, from 345 K to 335 K, are shown in the same figure. The desired closed-loop time constant for both controllers is $\tau_{cl} = 900$ s. Since these controllers are operating in the open-loop stable region, the conversion at these conditions is low and the nonlinearity is not very pronounced. With the linear controller, the response is slightly more sluggish for the increase in setpoint and overshoots slightly for the decrease in setpoint when compared to the nonlinear controller. The nonlinear controller quickly and smoothly reaches the final value of the setpoint regardless of direction.

Similar results for the two-degree-of-freedom linear and nonlinear PID controllers are shown in Figure 3. In this case the desired closed-loop time constant is $\tau_{cl} = 900$ s and the location of the zero-pole cancellations are assigned to $\tau_{ca} = 1800$ s and $\tau_{cb} = 1800$ s. This figure shows the two-degree-of-freedom controllers operating in an open-loop unstable region, with an increase in setpoint from 355 K to 360 K, and a decrease in setpoint from 360 K to 355 K. The linear con-
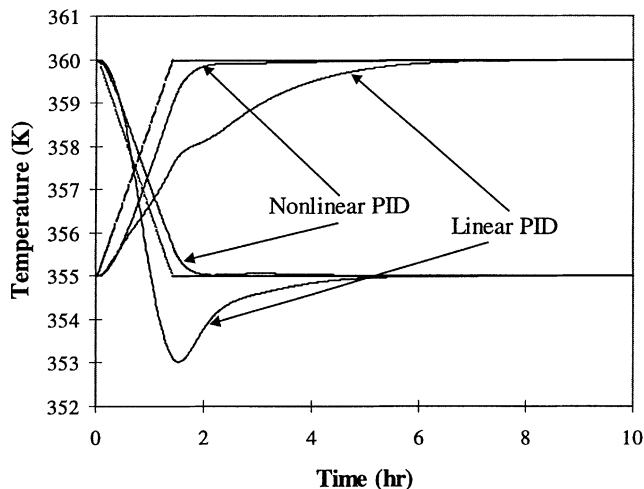
Figure 3. Response to step changes, both increase and decrease, for two-degree-of-freedom linear and nonlinear PID controllers.

troller is now noticeably sluggish for the increase in setpoint and overshoots the setpoint by several degrees for the de-
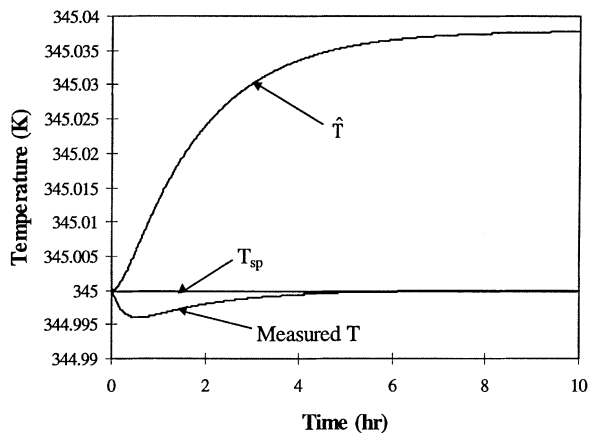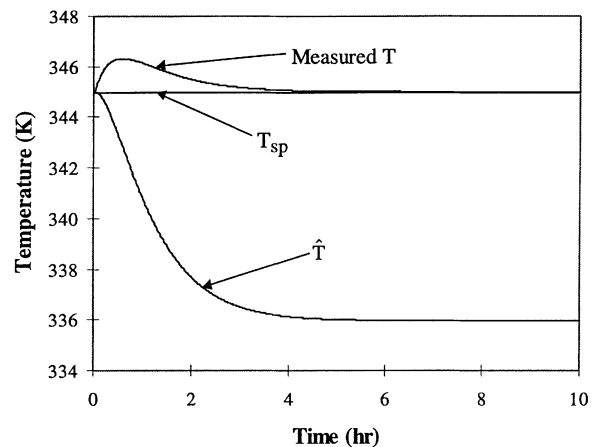

Figure 4. Response of one-degree-of-freedom nonlinear PID controller showing tracking of model error between approximate and true process.

Figure 5. Response of one-degree-of-freedom nonlinear PID controller to unmeasured 20% drop in F, demonstrating process monitoring and quantification of model error.

crease in setpoint. The nonlinear controller, however, smoothly reaches the final setpoint value regardless of the direction of the change.

The use of the model-based quantities for monitoring the process and a measure of model error is illustrated in Figures 4 and 5 for the one-degree-of-freedom nonlinear PID controller. In Figure 4, the state of the controller is initialized at the temperature set point of 345 K, leading to a steady-state value of the inlet jacket temperature that could be calculated from the approximate model. The movement in T_{ji} and $\hat{T}$ results from the model error between the approximate model (Eq. 41) for the reactor dynamics and the actual dynamics (Eqs. 36–37). The speed with which $\hat{T}$ moves corresponds to the location of the zero-pole cancellations in closed-loop. Similarly, the difference between $N(T,0)$ and T_{ji} corresponds to the model error at 345 K, and $N(\hat{T},0)$ approaches T_{ji} as it levels off. The same plots are shown in Figure 5 for an unmeasured 20% step decrease in the process flow rate, F. In the presence of this large disturbance, the controlled temperature changes by only a little more than a degree, but $\hat{T}$ and the difference between T_{ji} and $N(T,0)$ show large changes. If
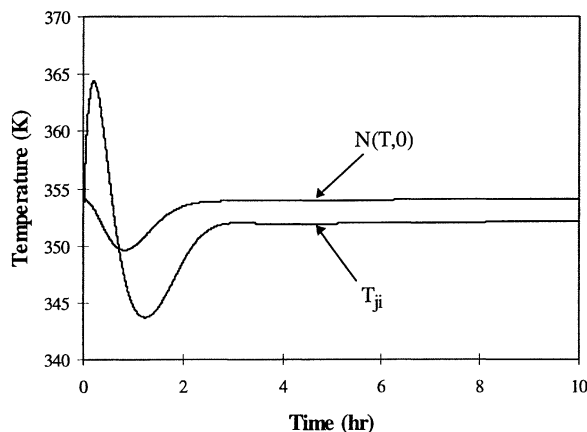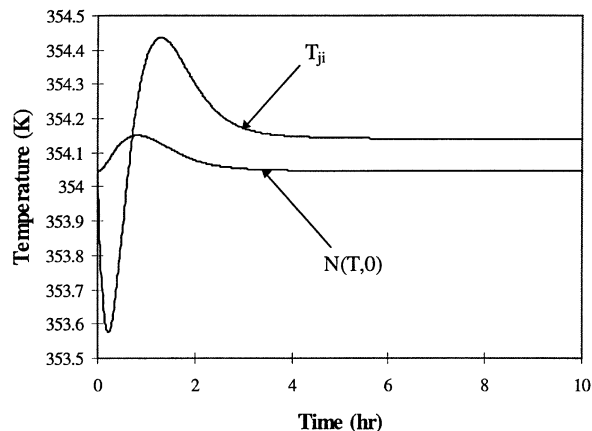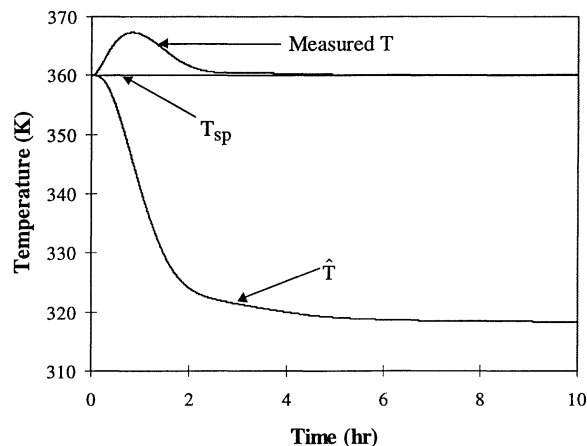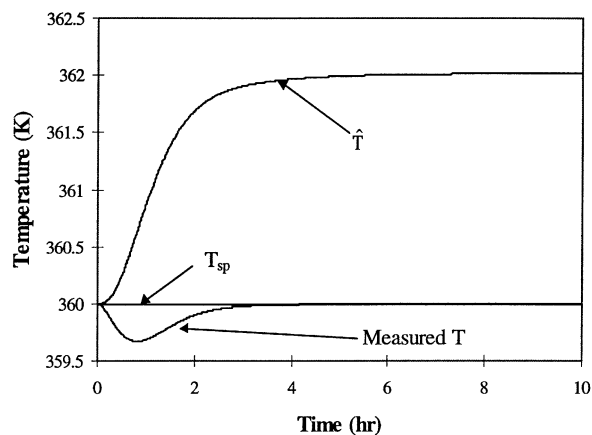
Figure 6. Response of two-degree-of-freedom nonlinear PID controller showing tracking of model error between approximate and true process.

Figure 7. Response of two-degree-of-freedom nonlinear PID controller to unmeasured 20% drop in *F*, demonstrating process monitoring and quantification of model error.

these quantities were used to monitor the process, it would be obvious that a large change in conditions had taken place.

The same scenarios are shown in Figures 6 and 7 for the two-degree-of-freedom nonlinear PID controller, but at the operating point of 360 K instead of 345 K. The tuning of τ_{ca} and τ_{cb} at 1800 s is faster than the normal modes of the process and the errors are rejected more quickly. Naturally, the difference between $\hat{T}$ and T in Figure 6 is much greater since the reactor is being operated at a point where there is more error between the approximate model and the actual process. This is also indicated, but to a lesser degree, by the difference between T_{ji} and $N(T,0)$. In Figure 7, the maximum deviation between the measured temperature and the setpoint in response to the unmeasured 20% decrease in F is a little over 7 degrees. However, the difference between T and $\hat{T}$ is over 40 degrees, clearly indicating a disturbance has entered the process. Notice, at this operating point that the process is more nonlinear and the conversion is higher. Quick detection and correction would be extremely important, and the use of the measures for on-line monitoring purposes could be very useful.

Conclusions

This work has shown that nonlinear PI and PID controllers result from the application of nonlinear controller design methods to nonlinear first- and second-order systems. The resulting controllers, based on nonlinear model-based synthesis methods, are in a convenient form for industrial implementation. Some of the quantities used in the controller have also been shown to be useful for monitoring the process and quantifying modeling error. Several chemical engineering examples were used to illustrate the resulting control laws. A simulation example further demonstrated the performance of these nonlinear controllers, as well as their useful process-monitoring quantities.

Notation

c_p = heat capacity of the reactor contents, kJ/(kg·K)
C_A = concentration of A in the exit stream, kmol/m^3
e = controller error signal
E = activation energy in $K(T)$, kJ/kmol
E' = activation energy for $K'(T)$, kJ/kmol
F = process flow rate, m^3/s
k = process static gain
k_c = controller proportional gain
k_0 = preexponential factor for $K(T)$, s^{-1}
k_0' = preexponential factor for $K'(T)$, m^3/kmol
m = mass of material in examples, kg
M = function of process output related to process dynamics
N = function of process output related to steady-state value of manipulated input

R = gas-law constant, kJ/(kmol·K)
s = Laplace transform variable
T = reactor temperature in examples, K
u = manipulated input
UA = overall heat-transfer times the area for heat transfer, kJ/(s·K)
V = volume of the reactor in examples, m^3
y = measured process output

Greek letters

$-\Delta H$ = overall heat of reaction, kJ/kmol
ρ = density of the reactor contents, kg/m^3
τ = time constant of first-order linear process, or natural period of second-order linear process
τ_c = time constant related to zero-pole cancellation in first-order systems
τ_{ca} = time constant related to zero-pole cancellation in second-order systems
τ_{cb} = time constant related to zero-pole cancellation in second-order systems
τ_{cl} = closed-loop time constant
τ_D = controller rate time
τ_I = controller reset time
ζ = damping factor of second-order linear process

Subscripts

in = inlet variable in examples
j = jacket variable in examples
ji = inlet jacket variable in examples
m = model variable
p = process variable
s = steady-state value
sp = set point

Literature Cited

Allgöwer, F., and F. J. Doyle, "Nonlinear Process Control—Which Way to the Promised Land?" CPC V, AIChE Symp. Ser., Vol. 93, p. 24 (1997).

Åström, K. J., and B. Wittenmark, *Computer-Controlled Systems, Theory and Design*, Prentice Hall, Englewood Cliffs, NJ (1990).

Balasubramhanya, L. S., and F. J. Doyle, "Low Order Modeling for Nonlinear Process Control," *Proc. 1995 Amer. Control Conf.*, Seattle, WA (1995).

Bequette, B. W., "Nonlinear Control of Chemical Processes: A Review," *Ind. Eng. Chem. Res.*, **30**, 1391 (1991).

Byrnes, C., F. D. Priscoli, and A. Isidori, *Output Regulation of Uncertain Nonlinear Systems*, Birkhuaser, Boston (1997).

Cheung, T. F., and W. L. Luyben, "Nonlinear and Nonconventional Liquid Level Controllers," *Ind. Eng. Chem. Fundam.*, **19**, 93 (1980).

Daoutidis, P., and C. Kravaris, "Dynamic Output Feedback Control of Minimum-Phase Nonlinear Processes," *Chem. Eng. Sci.*, **47**, 837 (1992).

Fogler, H. S., *Elements of Chemical Reaction Engineering*, Prentice Hall, Englewood Cliffs, NJ (1992).

Isidori, A., *Nonlinear Control Systems: An Introduction*, Springer, Berlin (1989).

Isidori, A., and A. Ruberti, "On the Synthesis of Linear Input-Output Responses for Nonlinear Systems," *Syst. Control Lett.*, **4**, 17 (1984).

Jutan, A., "A Nonlinear PI(D) Controller," *Can. J. Chem. Eng.*, **67**, 485 (1989).

Kailath, T., *Linear Systems*, Prentice Hall, Englewood Cliffs, NJ (1980).

Kravaris, C., and C. B. Chung, "Nonlinear State Feedback Synthesis by Global Input/Output Linearization," *AIChE J.*, **33**, 592 (1987).

Kravaris, C., and S. Palanki, "Robust Nonlinear State Feedback Synthesis Under Structured Uncertainty," *AIChE J.*, **34**, 1119 (1988).

Kravaris, C. P., Daoutidis, and R. A. Wright, "Output Feedback Control of Nonminimum-Phase Nonlinear Processes," *Chem. Eng. Sci.*, **49**, 2107 (1994).

Kravaris, C., and J. C. Kantor, "Geometric Methods of Nonlinear Process Control. I. Background," *Ind. Eng. Chem. Res.*, **29**, 2295

(1990); also, 2. Controller Synthesis, *Ind. Eng. Chem. Res.*, **29**, 2310 (1990).

Kravaris, C., and Y. Arkun, "Geometric Nonlinear Control—An Overview," *Chemical Process Control—CPCIV*, 447 (1991).

Kravaris, C., M. Niemiec, R. Berber, and C. B. Brosilow, "Nonlinear Model-Based Control of Nonminimum-Phase Processes," *Nonlinear Model Based Process Control*, R. Berber and C. Kravaris, eds., Kluwer Academic Publishers, Amsterdam, p. 115 (1998).

Lindskog, P., and L. Ljung, "Tools for Semi-Physical Modeling," *Preprints IFAC Symp Systems Identification*, **3**, 237 (1994).

Morari, M., and E. Zafiriou, *Robust Process Control*, Prentice Hall, Englewood Cliffs, NJ (1989).

Nijmeijer, H., and A. J. van der Schaft, *Nonlinear Dynamical Control Systems*, Springer, New York (1990).

Ogata, K., *Modern Control Engineering*, 2nd ed., Prentice Hall, Englewood Cliffs, NJ (1990).

Ogunnaike, B. A., R. K. Pearson, N. Samardzija, and J. D. Bomberger, "Low Order Empirical Modeling for Nonlinear Systems," *Proc. 1994 IFAC Conf.*, Kyoto, Japan (1994).

Pearson, R. K., and B. A. Ogunnaike, "Nonlinear Process Identification," *Nonlinear Process Control*, Chap. 4, M. A. Henson and D. E. Seborg, eds., Prentice Hall, Englewood Cliffs, NJ (1996).

Rivera, D. E., M. Morari, and S. Skogestad, "Internal Model Control. 4. PID Controller Design," *Ind. Eng. Chem. Process Des. Dev.*, **25**, 252 (1986).

Rugh, W. J., "Design of Nonlinear PID Controllers," *AIChE J.*, **33**, 1738 (1987).

Russo, L. P., and B. W. Bequette, "Impact of Process Design on the Multiplicity Behavior of a Jacketed Exothermic CSTR," *AIChE J.*, **41**, 135 (1995).

Stephanopoulos, G., *Chemical Process Control*, Prentice Hall, Englewood Cliffs, NJ (1984).

Appendix

Derivation of one-degree-of-freedom nonlinear PI controller

Consider the model state-feedback controller structure shown in Figure A1. The control law is derived by specifying the blocks of the control structure and combining the results. The blocks of the structure for the process given by Eq. 3 are as follows

Open-Loop Observer

$$\frac{d\hat{y}}{dt} = \frac{u - N(\hat{y})}{M(\hat{y})}. \tag{A1}$$

Corrected Setpoint Calculation
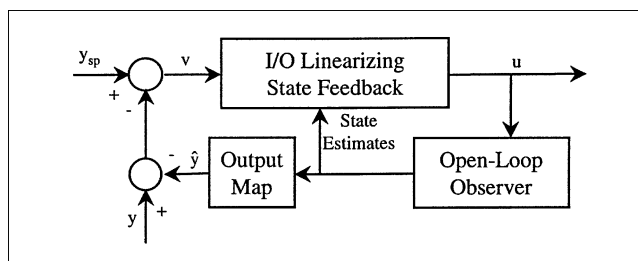
$$v = y_{sp} - y + \hat{y}. \tag{A2}$$

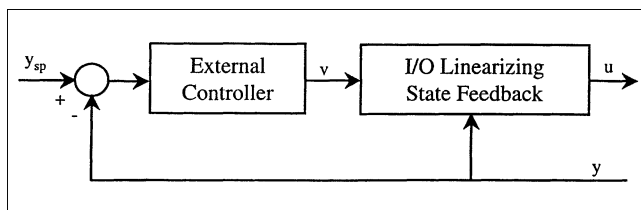**Figure A1. GLC-based model state-feedback controller.**

Figure A2. Basic GLC controller with state measurement.

Input−Output Linearizing State Feedback

$$u = M(\hat{y})\frac{v - \hat{y}}{\tau_{cl}} + N(\hat{y}). \qquad (A3)$$

Combining Eqs. A1−A3 leads directly to the overall control law

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - \hat{y}}{\tau_{cl}}$$

$$u = M(\hat{y})\frac{y_{sp} - \hat{y}}{\tau_{cl}} + N(\hat{y}), \qquad (A4)$$

which is exactly the result given by Eq. 5.

Derivation of two-degree-of-freedom nonlinear PI controller

Consider the basic GLC controller structure shown in Figure A2. The control law is derived by specifying the individual blocks and combining the results. For the process given by Eq. 3, the input−output linearizing feedback is given by

$$u = M(y)\frac{v - y}{\tau_c} + N(y). \qquad (A5)$$

This induces linear dynamics between v and y, with time constant τ_c

$$\tau_c\frac{dy}{dt} + y = v. \qquad (A6)$$

The external controller is a standard one-degree-of-freedom PI controller synthesized for the linear inner $v - y$ system (Eq. A6). Applying the general result (Eq. 5) shown earlier to Eq. A6 results in

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$v = \frac{\tau_c}{\tau_{cl}}(y_{sp} - y) + \hat{y}. \qquad (A7)$$

Combining Eq. 5 and Eq. A7 yields

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$u = M(y)\left[\frac{y_{sp} - y}{\tau_{cl}} + \frac{\hat{y} - y}{\tau_c}\right] + N(y), \qquad (A8)$$

which is exactly the controller given by Eq. 9.

Derivation of one-degree-of-freedom nonlinear PID controller

Consider the model-state feedback controller structure shown in Figure A1. The control law is derived by specifying the blocks of the control structure and combining the results. The blocks of the structure for the process given by Eq. 13 are as follows

Open-Loop Observer

$$\frac{d\hat{y}}{dt} = \hat{s}$$

$$\frac{d\hat{s}}{dt} = \frac{u - N(\hat{y},\hat{s})}{M(\hat{y},\hat{s})}. \qquad (A9)$$

Corrected Set Point Calculation

$$v = y_{sp} - y + \hat{y}. \qquad (A10)$$

Input−Output Linearizing State Feedback:

$$u = M(\hat{y},\hat{s})\frac{v - \hat{y} - (\tau_{cl} + \tau_{cl}^*)\hat{s}}{\tau_{cl}\tau_{cl}^*} + N(\hat{y},\hat{s}). \qquad (A11)$$

Combining Eqs. A9−A11 leads directly to the overall control law

$$\frac{d\hat{y}}{dt} = \hat{s}$$

$$\frac{d\hat{s}}{dt} = \frac{(y_{sp} - y) - (\tau_{cl} + \tau_{cl}^*)\hat{s}}{\tau_{cl}\tau_{cl}^*}$$

$$u = M(\hat{y},\hat{s})\frac{(y_{sp} - y) - (\tau_{cl} + \tau_{cl}^*)\hat{s}}{\tau_{cl}\tau_{cl}^*} + N(\hat{y},\hat{s}), \qquad (A12)$$

which is a rearrangement of the one given by Eq. 15.

Derivation of two-degree-of-freedom nonlinear PID controller

Consider the basic GLC controller shown in Figure A2. The control law is derived by specifying the individual blocks and combining the results. For the process given by Eq. 13,

the input−output linearizing feedback is given by

$$u = M\left(y, \frac{dy}{dt}\right) \frac{v - y - (\tau_{ca} + \tau_{cb})\dfrac{dy}{dt}}{\tau_{ca}\tau_{cb}} + N\left(y, \frac{dy}{dt}\right). \quad (A13)$$

This induces linear dynamics between v and y, with time constants τ_{ca} and τ_{cb}:

$$\tau_{ca}\tau_{cb}\frac{d^2y}{dt^2} + (\tau_{cl} + \tau_{cb})\frac{dy}{dt} + y = v. \quad (A14)$$

The external controller is a standard one-degree-of-freedom ideal PID controller synthesized for the linear inner $v - y$ system (Eq. A14). Applying the general result (Eq. 17) shown earlier to Eq. A14 results in

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$v = \frac{\tau_{ca}\tau_{cb}}{\tau_{cl}}\frac{d(y_{sp} - y)}{dt} + \frac{\tau_{ca} + \tau_{cb}}{\tau_{cl}}(y_{sp} - y) + \hat{y}. \quad (A15)$$

Combining Eq. A13 and Eq. A15 yields

$$\frac{d\hat{y}}{dt} = \frac{y_{sp} - y}{\tau_{cl}}$$

$$u = M\left(y, \frac{dy}{dt}\right)\left\{ \frac{1}{\tau_{cl}}\frac{d(y_{sp} - y)}{dt} + \frac{\tau_{ca} + \tau_{cb}}{\tau_{ca}\tau_{cb}}\left(\frac{y_{sp} - y}{\tau_{cl}} - \frac{dy}{dt}\right) \right.$$

$$\left. + \frac{\hat{y} - y}{\tau_{ca}\tau_{cb}} \right\} + N\left(y, \frac{dy}{dt}\right), \quad (A16)$$

which is exactly the controller given by Eq. 20.