Nonlinear Inferential Control

Nonlinear inferential control (NLIC) has been developed as a method for improving control of nonlinear systems. The controller is model-based, and allows for direct use of available measurements. This paper presents the structure of NLIC and the manner in which it is applied to processes when the controlled variables are measured. Also described is the improvement in the process control using NLIC. Two illustrative examples are presented, a laboratory heat exchanger process and a simulated neutralization process. The results indicate that a substantial improvement in control is possible using NLIC.

J. R. Parrish
Union Carbide Corp.
S. Charleston, WV

C. B. Brosilow
Case Western Reserve University
Cleveland, OH 44106

Introduction

Despite improvements in the theory and practice of process control over the past several decades, there remain classes of processes that either are left uncontrolled or are poorly controlled. These are processes that exhibit a very nonlinear response to controls. The effect of the control effort on the controlled process variable changes significantly due to disturbances or changes in the desired operating point. An example is the control of pH. pH control has frustrated many attempts at control (Gustafsson and Waller, 1983, 1984) because changes in buffering can radically change the amount of acid or base needed to achieve the desired pH. A single drop of strong acid or base can cause a change of several pH units in an unbuffered solution, whereas many thousand drops may be required to accomplish the same change in a strongly buffered solution. Since the degree of buffering depends on the concentration and types of salts in the solution, it is usually not feasible to measure it directly.

In recent years several nonlinear control strategies have been proposed to accomplish the control of nonlinear systems (Shinskey, 1962, 1973; Hunt et al., 1983; Meyer and Cicolani, 1980; Kravaris and Chung, 1987.) The last three references all present methods for transforming a nonlinear process into a completely or partially linear system. The most pertinent of the foregoing references is that of Kravaris and Chung. These authors achieve a linear input-output response for a reasonably broad class of systems by defining a new control effort that is linear in the actual controls and that depends on Lie derivatives and Lie brackets of the output and state equations. Our approach is to achieve a linear response of the output to the set point and disturbance effects by selecting the actual control effort so as to force the nonlinear process to follow a desired linear system trajectory. In those cases where the controls do not saturate and where the linear system responses and state estimation methods are the same, our methods and the approach of Kravaris and Chung should yield the same responses. By linear system response, we mean the "filter" response in our method, and the combined linear feedback controller, linear process response in the Kravaris and Chung method. The responses of the two methods should be the same even when there are modeling errors.

Nonlinear inferential control (NLIC) has the potential advantage over the transformation methods discussed above that control effort saturation can be accommodated readily and is not a potential source of instability as in the transformation methods. In addition, this paper treats the important issue of unknown inputs (i.e., disturbances), which is not treated in the linear transformation literature with which we are familiar. It is also likely that the implementation and design of nonlinear inferential control systems will be simpler than that of the transformation methods. However, the transformation methods—especially those of Karavaris and Chung—are important for the current work in that they provide its theoretical underpinning.

A major contribution of this paper is a structure proposed for the control of nonlinear processes. This structure focuses attention on the use of available measurements to infer how disturbances have influenced the process. It also provides a decomposition of the control problem, which facilitates the design of the nonlinear control system and provides a conceptually simple and practical method for tuning the control system to accommodate the inevitable modeling errors. On the other hand, we do not attempt to provide a rigorous, "turn the crank" method of designing all of the blocks within the proposed structure. Rather we present heuristics for the design of the various blocks that have worked well for the examples: control of a laboratory heat exchanger, a simulated pH process, and a simulated reactor process. The proposed design methods mainly follow those used for linear systems (Brosilow, 1979; Brosilow and Zhao, 1984; Parrish and Brosilow, 1985), with whatever extensions are necessary to accommodate the nonlinear process.

Structure of Nonlinear Inferential Control

The NLIC structure was selected to satisfy the following three criteria.

1. The structure should permit use of all process information (i.e., process measurements and process models) to accomplish control objectives.

2. The controller structure should provide for easy tuning. Tuning is "easy" when there is a single, physically meaningful parameter to adjust to obtain the desired performance-robustness trade-off.

3. The controller structure should reduce to that of a linear inferential controller (LIC) (Brosilow, 1979; Joseph and Brosilow, 1978; Parrish and Brosilow, 1985) or equivalently, a linear internal model controller (Garcia and Morari, 1982; Cutler and Ramaker, 1979) when the controlled variables are measured and the process model is linear.

The first criterion arises from the fact that unmeasured disturbance can profoundly change how the process responds to the controls. Frequently, all available process input and output measurements are needed to estimate the effects of the unmeasured disturbances.

The third criterion provides a link to existing theory and practice of linear model-based control. This link can assist in extrapolating intuition concerning linear control system design methods to the design of nonlinear systems.

The structure proposed for the nonlinear inferential controller with measurements of the controlled process variables is shown in Figure 1. If the controlled process variables are not measured, then a slightly modified version of Figure 1 can be used to infer the unmeasured variables as described by Parrish and Brosilow (1986). The following paragraphs give a brief overview of the function of each of the components of Figure 1. More detailed descriptions follow.

Process. The process block represents the physical system to be controlled. The process is a nonlinear continuous time dynamical system. Since a discrete time version of the controller is implemented, it is the sampled process that we attempt to regulate.

Estimator. The estimator block uses process and control effort measurements to estimate the unmeasured disturbances and the process state (if needed). The difference between the value of the controlled variable projected by the model and that inferred by the estimator is a measure of the error in projecting the effect of the disturbances across the sampling interval.

Model. The process model is either a continuous time or a discrete time dynamical system. Its role is to predict the process output in the future (e.g., the next sampling interval) as needed by the controller. Projections are based on the disturbance estimates provided by the estimator and on the control efforts provided by the controller.

Comparator. The comparator provides an incremental measure of the difference between the predicted and actual effects of disturbances on the process output.

Summer. The summer provides a cumulative measure of the errors in predicted effects of the disturbances on the process output. The cumulative effect of model prediction errors E plays the same role in NLIC as the effect of the disturbances on the controlled variable in the linear internal model controller of Figure 2b.

Trajectory Generator. The trajectory generator translates changes in set point and the cumulative projection errors to a desired system response. The output of the trajectory generator becomes a set point, or tracking signal, for the output of the model. In this paper, we consider only linear trajectory generators of the form shown in Figure 1. The trajectory filter is chosen as a lag in exactly the same way that the filter is chosen for linear inferential control systems (Parrish and Brosilow, 1985).

Controller. The controller selects a control effort within the constraints, to make the model output (and so the process output) follow the output specified by the trajectory generator as closely as possible (as measured by some norm).

Let us now verify that the control structure of Figure 1 satisfies the three criteria given at the beginning of this section.

The estimator in Figure 1 uses all available measurements and the process model to provide estimates of the unmeasured disturbances, and extrapolates these estimates into the future. Thus criterion one is satisfied.

Criterion two is satisfied since the control system is tunable via a single, physically meaningful parameter: the trajectory filter time constant. The tuning parameter represents the speed of response of the closed-loop system when the model is perfect and no control effort constraints are encountered. For a value of the tuning parameter approaching zero, the control system provides rapid set-point tracking. For a value of the tuning parameter approaching infinity, a constant controller output is generated and the process response is the open-loop response (Parrish, 1985). The speed of system response may be adjusted to provide stable control in spite of modeling errors.

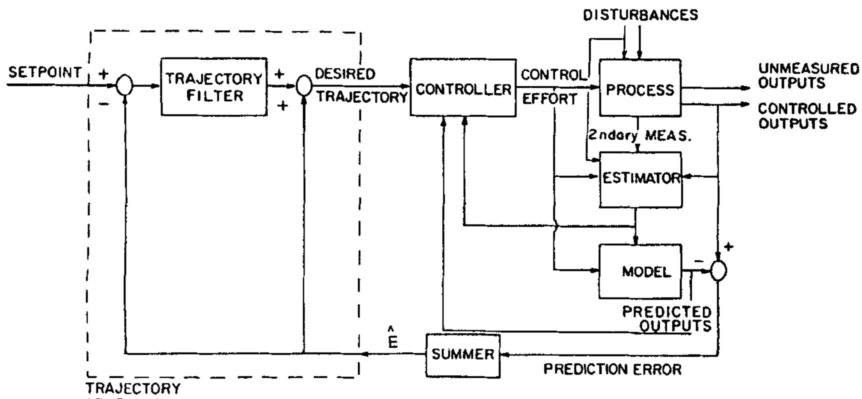

Figure 1. Nonlinear inferential control (NLIC) system.

The control system will provide perfect steady state tracking of the set point provided only that the trajectory generator filter has unity gain. As with the linear internal model controller, the no-offset property is independent of modeling errors.

To achieve the tuning and no-offset properties discussed above, it is necessary that the model be such that there exists a set of inputs (i.e., disturbances) which could have caused the observed outputs. Otherwise, the estimator of Figure 1 will not be realizable. The foregoing, almost self-evident requirement is discussed in some depth in the section titled Process Modeling Requirements.

The NLIC structure shown in Figure 1 reduces to the more familiar control system structure shown in Figure 2a when the process model is linear. Figure 2a is a trajectory tracking version (Popiel et al., 1986) of the linear internal model control (IMC) structure shown in Figure 2b. The trajectory tracking controller structure improves control system performance over linear IMC when control effort saturation is an important constraint. In the trajectory tracking implementation the controller utilizes the model state, and so can compensate for control effort saturation. When there are no constraints on the control effort, the structure of Figure 2a reduces exactly to that of Figure 2b. The role of the controller in the trajectory tracking configuration is to make the model track the desired response as given by the trajectory "filter." In order for the controller to be realizable the trajectory filter must cancel any unrealizable elements in the model inverse. (In practice, Figure 2a is modified by implementing only the lag portion of the trajectory filter and feeding back to the controller only the state of the invertible part of the process model.)

Figure 1 reduces to Figure 2b for linear models in the following sense:

1. The feedback signal E is the same for Figures 1 and 2
2. The responses of the controlled variables are the same

When the process model is perfect, the response of the controlled variable is given by

$$y(s) = F(s)v(s) + [I - F(s)]E \tag{1}$$

where $y(s)$ = controlled process variable
$F(s)$ = trajectory filter (selected to make the controller realizable)
E = accumulated effect of unaccounted disturbance (and modeling errors) on the controlled variable

A proof of the foregoing statement can be found in a paper by Parrish (1985).

There are quite obviously substantial structural differences between Figures 1 and 2a even though they are equivalent for linear process models. These differences arise from:

1. The need to use the disturbance estimates in the nonlinear model to generate the process variables
2. The need to generate the desired output (i.e., controlled variable) of the nonlinear model and not just the desired effect of the control on the output as in the linear case

The process model in NLIC uses the projected value of the disturbances as well as the current control effort to calculate the process variable at the next sampling interval. The control effort is calculated (perhaps iteratively) to make model output at the next sampling interval (or at the preselected time horizon) equal to that specified by the trajectory generator at that future time. The incremental difference between the measured process vari-

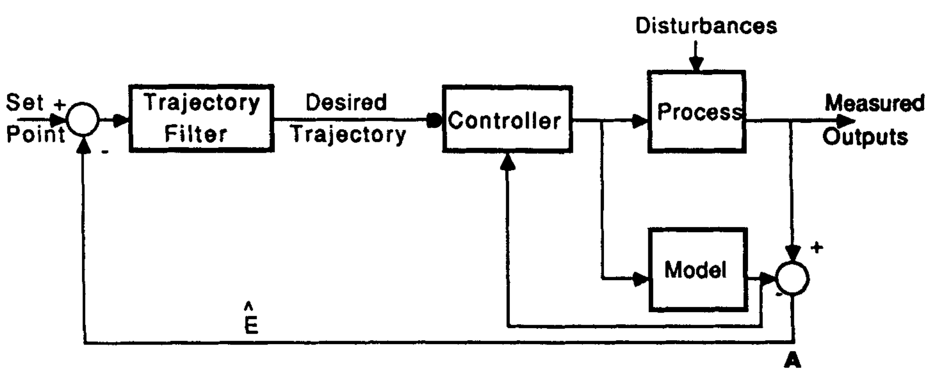

Figure 2a. Trajectory tracking internal model controller.

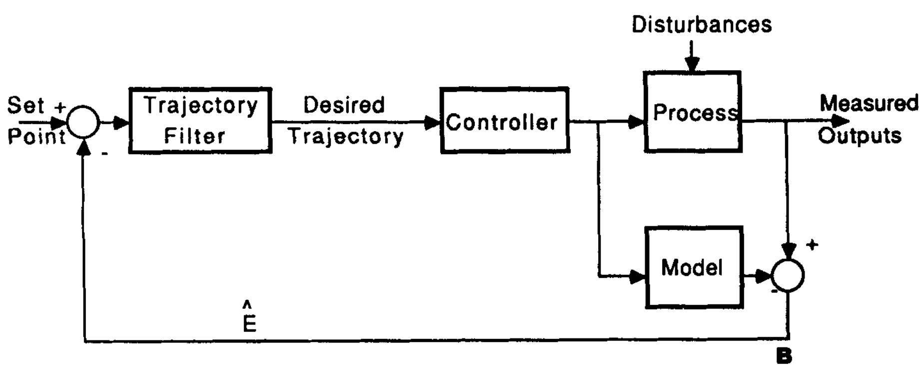

Figure 2b. Linear internal model controller.

able and that projected by the model is due either to errors in projecting the disturbances and/or to modeling errors. Summing the incremental errors gives the accumulated effect of unaccounted disturbances and modeling errors.

The trajectory generator in Figure 1 implements Eq. 1 directly. In trajectory tracking IMC, Figure 2a, the disturbance response part of Eq. 1—i.e., $[I - F(s)]E$—is implemented implicitly by the structure by subtracting the accumulated effect of the disturbances, E, from the set point. For a nonlinear process such an implicit implementation is not valid and is replaced by the implementation in Figure 1. The trajectory filter in Figure 1 is generally a lag cascade with a dead time (which might in general vary with process conditions).

Process Modeling Requirements

The NLIC model relates the control effort and disturbances to the process measurements. The models considered in the current applications of NLIC may be expressed as:

$$X_k = f(X_{k-1}, m_{k-d}, U_k, A) \tag{2}$$

$$y_k = h(X_k) \tag{3}$$

$$\theta_k = \theta(X_k, U_k, m_{k-d}) \tag{4}$$

where X = the state vector of dimension n
y = the model output (a scalar)
θ = process measurements of dimension r
U = input disturbances of dimension p
A = a parameter vector of dimension q
m = control effort (a scalar)
d = control effort time delay ($d > 0$)
k = time index

The following conditions are placed on the process model in order that it be used with the NLIC. These conditions arise so that the NLIC reduces to LIC when a linear model is used, and that the stability of the control system can be assured for large enough values of the filter tuning parameter.

Condition 1. The partial derivatives

$$\left(\frac{\partial f}{\partial X}\right); \left(\frac{\partial f}{\partial m}\right); \left(\frac{\partial f}{\partial U}\right); \left(\frac{\partial f}{\partial A}\right); \left(\frac{\partial h}{\partial X}\right) \tag{5}$$

must all exist and be finite throughout the operating region $\{X, m, U, A\}$.

Condition 1 allows the model (given by Eqs. 2 and 3) to be linearized about all operating points so that at all operating points the model may be written as

$$y(z) = G_1(z) * m(z) + \sum_{j=1}^{p} G_{j+1}(z) * u_j(z) \tag{6}$$

where

$$G_j = \frac{K_j \prod_{i=1}^{n} (a_{ij} z^{-1} + 1)}{\prod_{i=1}^{n} (b_{ij} z^{-1} + 1)}$$

$$j = 1, 2, \ldots, p + 1 \tag{7}$$

where the gains and time constants of Eq. 6 are functions of the stationary point

Condition 2. The process model must be locally asymptotically stable everywhere in the operating range.

Condition 3. The sign of the steady state gain, K, must not change in the operating region. That is $|K| > 0$ where

$$K = \left(\frac{\partial Y}{\partial m}\right) = \left(\frac{\partial h}{\partial X}\right)^T \left(\frac{\partial X}{\partial m}\right) \tag{8}$$

Condition 3 is a carryover from the design of linear controllers for linear processes. It is a necessary condition for the existence of a stable control system with no steady state offset of the controlled variable from its set point (Parrish, 1985).

Condition 4. The model must be capable of exhibiting the measured process response in spite of process modeling errors. This property is referred to as "model consistency." Model consistency is required so that the model response can follow the process. Intuitively, one would expect that if the model were incapable of replicating the process response, then use of the model to predict and specify process performance, as is done in NLIC, would not be successful.

The model given by Eqs. 2 through 4 is said to be consistent if for any sequence of process measurements $\theta_1, \theta_2, \ldots, \theta_k$, controls $m_o, m_1, \ldots, m_{k-d}$, an initial state X_0, and parameters A, there exists an associated sequence of disturbances $U_1, U_2, \ldots$ such that the model outputs $\hat{\theta}_j, j = 1, \ldots, k$ (i.e., the predicted measurements) match the observations $\theta_j, j = 1, \ldots, k$. That is, $\hat{\theta}_j = \theta_j, j = 1, \ldots, k$.

In the above definition, there is no need for the sequence of disturbance vectors $U_j, j = 1, \ldots, k$, initial state X_0, and parameters A to be unique. Indeed, whenever U_k has more elements than θ_k (i.e., dim $U_k >$ dim θ_k) it is likely that multiple values for U_k will yield $\hat{\theta}_k = \theta_k$.

In general, the set of all possible sequences of observations θ_k and controls m_{k-d}, $k = 1, \ldots, d$, is not known *a priori*. However, the approximate boundaries for the set of all possible observations θ_k, disturbances U_k, states X_k, and controls m_k are often available. A sufficient condition for consistency is that the mapping from the set of all feasible disturbances, U, covers the set of all possible observations, θ_k, for any feasible state, X_{k-1}, and controls, m_{k-d}. If a model satisfies the foregoing, then we say it is strongly consistent. Any strongly consistent model is also consistent.

A subset of strongly consistent models comprises those for which the mapping from U_k to θ_k is one to one and onto for any X_{k-1} and m_{k-d}. Such models are said to be uniquely consistent. In the examples that follow, the disturbance descriptions are restricted sufficiently so that the model becomes uniquely consistent. On the other hand, if the original model is not strongly consistent, it can be made so by adding other possible disturbances either from first principles or simply by adding hypothetical disturbances to the measurements. Errors in postulating how disturbances enter the process will exhibit themselves as errors in the predicted process outputs, and these in turn will necessitate a slower response to set point changes and a slower recovery from disturbances.

The linear models used in model predictive-type controllers (Brosilow and Zhao, 1984; Cutler and Ramaker, 1979; Garcia and Morari, 1982; Mehra and Roulani, 1980) are always consistent because the difference between the predicted and observed outputs is interpreted as being due to additive disturbances. Of course, any nonlinear process can also be modeled as having disturbances directly influencing (i.e., adding into) the process outputs, and such a model will be consistent. However, since in nonlinear processes the disturbances can profoundly influence how the controls affect the outputs, it is far better to expend the effort required to model how disturbances actually enter the process than to simply assume that they add to the observations.

Estimator Requirements

The purpose of the estimator module is to provide an on-line estimate of the process state and disturbances at each time step. The estimator provides the estimates based on the control effort and the available process measurements. The disturbance estimates are used by the model module to predict the process output, and by the controller module to choose a control effort that will force the model (and therefore the process) to move along the desired trajectory in spite of disturbances.

There are as yet no general algorithms for the estimation of unmeasured disturbances, process state, and uncertain parameters for processes modeled as in Eqs. 2–4. Nonetheless, for specific examples, such as the neutralization and heat exchanger processes described later, it is often easy enough to construct the desired estimators. Since there are frequently more disturbances than measurements, the estimator design problem is often one of constructing a model of the disturbances, which produces a uniquely consistent model. The Appendix provides a simple technique for obtaining uniquely consistent models for a class of single-input/single-output processes, with the process output as the measured variable and where the disturbances enter either linearly or bilinearly.

Systems Lacking Primary Measurements

Nonlinear inferential control is applicable to systems that have secondary measurements but lack output measurement. The estimator uses the available measurements to estimate the state and disturbances as previously described. Given a state estimate $\tilde{X}$, an output estimate $\tilde{y}$ may be calculated. The estimated output may then be used in the control scheme as the "controlled" output shown in Figure 1. Note that if there are no process measurements, then the estimated output $\tilde{y}$ is always equal to the model output. In this case, E is always zero and the NLIC structure of Figure 1 reduces to a nonlinear open-loop control system.

Trajectory Generator and Controller Requirements

The designs of the trajectory generator (TG) and the controller modules are interrelated. The TG provides the desired trajectory to be tracked by the process and model, while the controller enforces the tracking. The two modules are first considered separately, and then their interrelations are examined.

Trajectory generator

The purpose of the trajectory generator is to provide a reference signal y^d, which is the desired response of the process. As stated earlier, this research deals only with linear trajectory generators, as given by Eqs. 1 and 9 below. For the class of systems considered in this work, the desired response of the system was previously given as

$$y^d = Fv + (1 - F)E \qquad (1)$$

where $F = (f)z^{-D}$
D = the number of sample periods of measurement or control effort time delay
f = the response "filter," which is of the form:

$$f = (\Delta t)^n/(-\epsilon z^{-1} + \epsilon + \Delta t)^n \qquad (9)$$

where ϵ = the TG time constant
n = the order of the filter
Δt = the sample period

The response filter represents the desired rate at which the accumulated effects of unaccounted-for disturbances or modeling errors E are to be eliminated from the output. In general, the actual system response will differ from the ideal (TG) response due to modeling errors and control effort constraints.

The order of the filter is selected to make the continuous time version of the TG-controller pair realizable. The filter has a steady state gain of unity, so that the system will have no steady state error (i.e., $y^d = v$ at steady state).

The filter is tunable, by means of the time constant ϵ, to stabilize the system against modeling errors. System stability is guaranteed for ϵ approaching infinity, regardless of modeling errors, as long as the model, estimator, and TG are consistent (Parrish, 1985). In addition, the local stability of the system is guaranteed for ϵ large enough because NLIC reduces to LIC for operation about a local stationary point.

Because the NLIC structure is specified so that the system reduces to a linear inferential control system when applied to a linear model, it is necessary that the filter-controller pair be realizable for a continuous time system. This constraint is illustrated for the linear system below:

Given: $\hat{G}(s) = N(s)/D(s)$

Then:

the order of the filter, n,

$$\geq \text{order of } D(s) - \text{order of } N(s) \qquad (10)$$

With this requirement, the filter-controller pair

$$F * \hat{G}^{-1} = N_l(s)/D_l(s) \qquad (11)$$

will be realizable, i.e.,

$$\text{order } [D_l(s)] \geq \text{order } [N_l(s)] \qquad (12)$$

For nonlinear processes the method equivalent to the above is to choose the filter order as the relative order of the nonlinear system, as described by Kravaris and Chung (1987). However, if the relative order is difficult to compute, or if the derivative of the output associated with the relative order is only weakly dependent on the control effort, then we recommend choosing a filter order higher than the relative order. The simplest and most conservative rule is to set the filter order equal to the process order. That is,

$$n = \dim [\hat{X}] \qquad (14)$$

where $\hat{X}$ is the state vector of the model as defined in Eq. 2. For the examples described in this work, the two rules provide the same filter order. As with the model and estimator, the TG has a consistency requirement: The desired trajectory must approach the measured process response as ϵ approaches infinity. When the desired trajectory is the measured process response, the control that maintains the desired trajectory is constant. A constant control assures system stability because the process is open-loop stable. In this work, only linear TG's with response filters of the form given by Eq. 9 are used. With an Eq. 9 linear TG, the consistency of the TG is due only to the manner in which the accumulated projection error E is calculated. The desired trajectory will be the open-loop trajectory for ϵ approaching infinity when the accumulated projection error is calculated by Eq. 15 (Parrish, 1985).

$$E_k = E_{k-1} + y_k - \hat{y}_k \tag{15}$$

For a first-order process, the desired trajectory may be directly generated by Eq. 15 and the trajectory generator. When the system is second order or higher, it is generally necessary to project the cumulative differences between the model and the process outputs E into the future beyond t_k. This is done as follows:

$$\tilde{E}_{k+i} = \tilde{E}_{k+i-1} + \tilde{y}_{k+i} - \hat{y}_{k+i} \tag{16}$$

where $\tilde{E}_{k+i}$ = projected cumulative effect of disturbances on the process output (with $\tilde{E}_k = E_k$)

$\hat{y}_{k+i}$ = projected model output (projected from the initial state $\hat{X}_k$ through the model algorithm with some assumption about how $\tilde{u}_k$ will progress into the future)

$\tilde{y}_{k+i}$ = projected process output (projected from the initial state $\tilde{X}_k$ through the model in the same manner as $\hat{y}_{k+i}$)

Equation 15 is equivalent to Eq. 16 if the error is projected only one step into the future. Note that the projection calculation of the process and model are the same, but the initial states may be different. The initial states are in fact different because disturbances affect the process before affecting the model. Specifically, disturbances affect the model only after being estimated.

Controller

The controller module calculates the controls that will force the model output to track the desired trajectory (supplied by the trajectory generator) at selected points in the future. To do this, the controller module uses the future values of the disturbances as projected by the estimator.

The values at the future times (i.e., time horizon) at which the model output is to match the desired trajectory is a tuning parameter that is selected to give a stable control. Ideally, the time horizon should be chosen to be as short as possible so as to minimize the errors introduced by projecting the disturbances into the future. If the model has an effective dead time that cannot be factored out, as is done for the linear controller, then the time horizon must exceed the effective dead time. Similarly, if the model has an inverse reponse, limited computational experience (Bridle, 1985) indicates that the time horizon must exceed the duration of the inverse response. Finally, a rule of thumb that has worked well on nth-order lumped-parameter models without inverse response is to choose a time horizon of n sample periods (Parrish, 1985).

The TG-controller module calculations of the NLIC are potentially the greatest computational load in the control scheme. The relatively large amount of computation is due to two factors: the projection of the system response and control effort saturation. The computation requirements of the controller due to these factors are considered below.

In order to calculate the control effort required for model tracking of the desired response, several of the control system variables must be projected into the future (assuming that the system is higher than first order and/or has delays). The variables requiring projection are:

a. The estimated (or measured) process output y

b. The model output $\hat{y}$

c. The cumulative differences in process and model response E

d. The desired trajectory y^d

In order to project the process and model outputs, a prediction of the control effort is needed. The controller algorithm may then require the simultaneous solution of the equations representing the process, model, and control effort projections. Generally, the solution of such a set of nonlinear equations is difficult. The computational load may be reduced considerably by making assumptions about the projections so as to decouple the set of equations. One such method is to assume that the control effort is constant into the future for the purpose of projecting the process and model outputs. The control effort is then calculated based on the desired trajectory produced by the projections. Other methods for solving the controller equations which minimize computation are given by Parrish (1985).

NLIC Applied to a Neutralization System

The following example illustrates the application of NLIC to a simulated pH process. A conventional (PI) controller response is included for comparison.

The pH process considered in this example is illustrated in Figure 3. The process consists of a continuously stirred tank (CST) into which a feed steam of unknown composition is flowing. The solution in the CST is to be maintained at a desired pH by addition of a basic solution. The feed to the CST is assumed to be a monoprotic acid (HA) and its soluble salts (A−). The concentrations of the acid and the salts in the CST are unknown. The neutralization is carried out using a solution of a strong base (such as NaOH, KOH, etc.) of concentration Cr at a flow rate of m (the manipulated variable). A model of the system is derived based on the following assumptions:

1. The CST is perfectly mixed
2. The acid-base equilibrium is fast
3. The equilibrium of the mixture of acids and salts is effectively represented by a single expression

The third assumption is based on the fact that mixtures of acids and bases often produce titration curves similar to those produced by a single acid and its salts (Gusstafson, 1982; Gusstafson and Waller, 1983, 1984).

The process has been modeled by the following equations:

$$(H+) + (N+) = (A-) + (OH-) \text{ charge balance} \tag{17}$$

$$(HA) + (A-) = U \text{ (an unknown)} \tag{18}$$

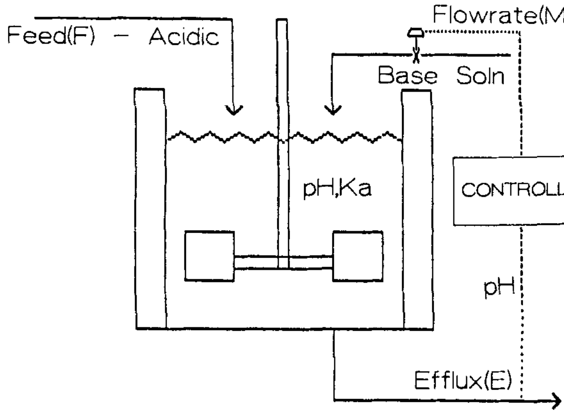

Figure 3. Neutralization process.

$$(H+)(OH-) = Kw = 10^{-14} \tag{19}$$

$$(H+)(A-)/(HA) = Ka = \text{acid/base equilibrium constant} \tag{20}$$

$$\frac{Vd(N+)}{dt} = m * Cr - (F + m) * (N+) \tag{21}$$

$$pH = -\log(H+) \tag{22}$$

Equations 17 to 20 may be reduced to

$$(H+)^3 + [Ka + (N+)](H+)^2 + [(N+)Ka - Kw - KaU](H+) - KaKw = 0 \tag{23}$$

The parameters of the process are

$(N+)$ = concentration of the cation (of the base) in the CST (which is not measured)
Kw = equilibrium constant for water
Ka = relative strength of the acid mixture in the CST; note that if there is a mixture of acids then Ka may be the "apparent value" of the equilibrium constant
F = feed rate to the tank
m = flow rate of the base (control variable)
Cr = concentration of the base solution
U = unknown concentration of the acid and its salts
V = volume of the tank (assumed to be constant)

All parameters are constrained to be positive except the flow of the base, m, which may be zero.

The model is highly nonlinear, with gain variation of two or more orders of magnitude over the operating region. The gain variation is due to several factors, among them changes in operating point pH, buffering level U, and acid strength Ka.

An NLIC system directly utilizes the process model described above. The controller modules are specified by algebraic manipulation of the discrete time equivalent of the model given by Eqs. 21–23. The design follows.

The first step in the design is to check the model characteristics in order to determine if a simple controller is suitable. The model has the following characteristics:

1. The model gain is always positive. The gain corresponds to the slope of a titration curve that is always positive.
2. The model exhibits no inverse response, so a simple control effort projection is suitable.
3. Any model output can be achieved by proper selection of the disturbance U; that is, the model is strongly consistent.
4. The model and process are self-regulating (open-loop stable).

Given that the model is suitable, one can proceed with design of the individual modules. The estimator design utilizes no measurement filtering (cases 1 and 2); the measurements are applied directly in the control algorithms. The estimator algorithm consists of merely solving Eq. 24 for $\hat{U}_k$ where

$$(H+)_k^3 + [Ka + (N+)_k](H+)^2 + [(N+)_k Ka - Kw - KaU_k](H+) - KaKw = 0 \tag{24}$$

where $(N+)_k = (\hat{N}+)_k$ (as given by the model)
$(H+)_k = 10^{-(pH_k)}$
pH_k = output measurement at time k
$\hat{U}_k$ = solution of Eq. 24

A one-step-ahead algorithm is used in the controller. The algorithm forces the model pH to follow the desired pH, $(pH)_{k+1}^{D}$, at the next sampling interval, as follows: From the desired pH, (which is provided by the trajectory generator) a desired value of the cation concentration $(N^+)_{k+1}^d$ is obtained by solving Eq. 24 with $\hat{U}_{k+1} = \hat{U}_k$. Then, the current control effort, m_k, is obtained by solving a discrete form of Eq. 21 with the desired cation concentration replacing the actual cation concentration. The discrete form of Eq. 21 is

$$\hat{V}[(N^+)_{k+1} - (N^+)_k] = \Delta t * m_k * C_r - \Delta t * (\hat{F}_k + m_k) * (N^+)_{k+1} \tag{25}$$

Setting $(N^+)_{k+1} = (N^+)_{k+1}^d$ and solving for m_k gives:

$$m_k = \frac{\hat{F}_k * (N^+)_k + \hat{V} * [(N^+)_{k+1}^d - (N^+)_k]/\Delta t}{[Cr - (N^+)_{k+1}^d]} \tag{26}$$

The trajectory generator filter is a first-order lag like that given by Eq. 9 because the model is only a first-order system.

The NLIC pH controller was applied to a variety of simulated test cases with varying degrees of modeling error, with and without noise corruption of the measurements. The cases considered here are:

1. Perfect modeling
2. Feedforward vs. nonfeedforward control using the feed flow measurement
3. Imperfect modeling

The pH system has a nominal operating point as given in Table 1. During the simulation tests, two disturbances affect the process. The first disturbance is a feed rate change from 10 to 12 L/min, which occurs at 20 min. A second disturbance is a change in the acid/buffering concentration that begins at 35 min and rises as a ramp for 10 min. The acid/buffer concentration becomes constant at 45 min. All of the tests described below and the results shown in Figures 4 to 9 are subject to these disturbance inputs.

Test case 1

The process/model parameters and input changes are given in Table 1. The model is a perfect representation of the process

Table 1. pH Process/Model Parameters and Inputs

Parameter	Value and/or Remark
Ka	Nominal value of 0.001 except where otherwise indicated in test cases
Cr	Nominal value of 1.0 equiv/L
V	Nominal value of 90 L
D	Nominal value of 4 (sample periods) except where otherwise indicated in test cases
Δt	Sampling period, 0.10 min
m	Upper limit of 3.0 L/min Lower limit of 0.0 L/min
F	10 L/min for $0 < t < 20$ min 12 L/min for $20 < t < 70$ min When the flow rate is measured, the model feed rate is equal to the measured (process) flow rate. When the flow rate is not measured, the model rate is constant (10 L/min) throughout the simulation.
U	A measure of the degree of buffering in the system. *U* is assumed to be a primary disturbance which influences the system. It varies as follows: 0.10 mol/L for $0 < t < 35$ min ramping to 0.13 mol/L over $35 < t < 45$ 0.13 mol/L for $45 < t < 70$ min
Set point	4 units for $0 < t < 5$ min 7 units for $5 < t < 70$ min

(i.e., the model and process are represented by identical equations). For this case, the inlet flow rate is measured, and the trajectory filter time constant ϵ is chosen as 1 min. The system response using NLIC and PI controller for this application is given in Figure 4. The PI controller was tuned to provide a stable response for the conditions of the test. The tuning parameters are:

$$\text{Reset time} = 3.0 \text{ minutes}$$

$$\text{Controller gain} = 0.003 \text{ L/min} \cdot \text{pH}$$

The control response is sluggish because the PI controller must be tuned to maintain stability in spite of the severe nonlinearities.

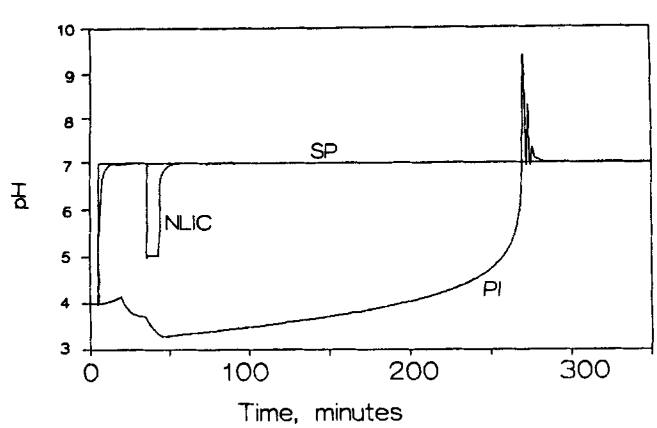

Figure 4. Response of neutralization process to set point changes and disturbances given in Table 1.

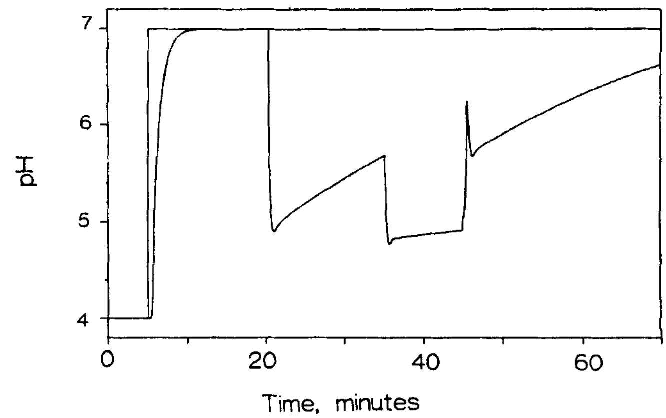

Figure 5. Response of neutralization process to set point changes and disturbances given in Table 1.
Feed flow rate is not measured

The PI control system response could be improved by incorporating some of the model information into the controller. The method normally used is gain scheduling (Shinskey, 1973) that is, modifying the controller gain based on the measured pH. A gain-scheduled controller will allow for compensation of the process gain with pH. However, gain variations are also due to changes in buffering. These gain variations are unaccounted for in the usual gain-scheduling approaches; therefore changes in buffering substantially degrade the gain-scheduled PI controller performance. The NLIC design, however, performs well in spite of buffering changes, because buffering changes are estimated using the process model.

Test case 2

This test is identical to test 1 except that the inlet flow rate is not measured. The model parameter $\hat{F}$ is chosen to be 10 L/min. over the duration of the simulation. Although the model is otherwise identical to the process, the fact that the model inlet flow rate is held constant while the actual flow rate is changing is a source of modeling error. In addition, because $\hat{F}$ is constant, there is no feedforward compensation in the control system when changes in F occur.

The system performance under the conditions indicated in Table 1, and a filtering time constant of 1 min, are illustrated in Figures 5 and 6.

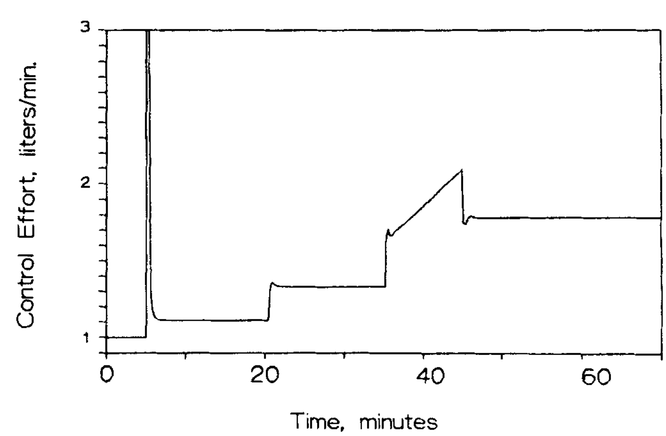

Figure 6. Control effort for responses shown in Figure 5.

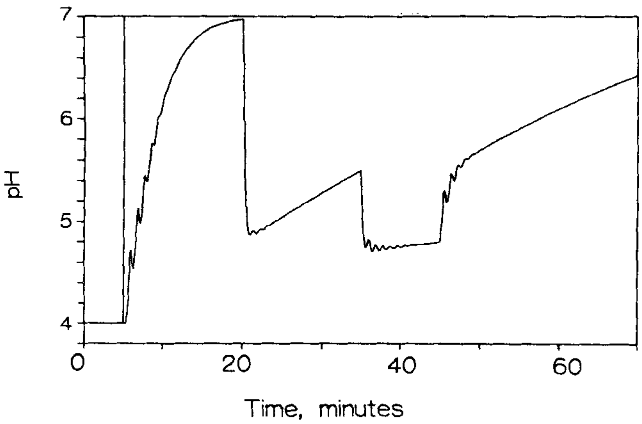

Figure 7. Response of neutralization process using controller having modeling errors.

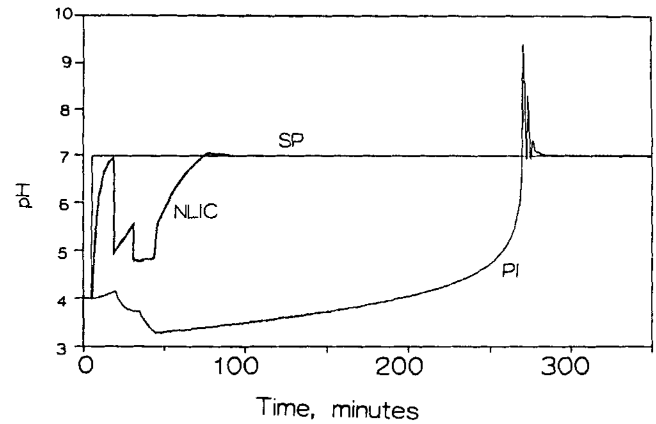

Figure 9. Response of neutralization process.
Comparison under NLIC with modeling errors and a well-tuned PI controller

Test case 3

The performance of the NLIC under conditions of modeling error is examined in this test. The process/model parameters and input changes are as given in Table 1, except:

$$\hat{F} = 10 \text{ L/min (constant)}$$

$$Ka = 0.001, \hat{Ka} = 0.0005$$

$$D = 3 \text{ min}, \hat{D} = 4 \text{ min}$$

An additional dynamic modeling error is due to the control effort passing through an unmodeled first-order lag. The first-order lag has unity gain and a time constant of 0.1 min. The first-order lag could correspond to the response of a cascaded flow control loop on the inlet base flow rate.

Under the modeling error conditions indicated, a filter time constant of 6 min is required to stabilize the system. The NLIC system response is shown in Figures 7 and 8. The response of the NLIC system with modeling errors is again compared to that of a well-tuned PI controller in Figure 9.

Heat Exchange System

The laboratory heat exchange process illustrated in Figure 10 consists of two concentric pipes, with water flowing at a rate m on the inside tube and steam condensing in the annular space. For details of the heat exchanger construction, refer to Shine (1980). The outlet water temperature To is the controlled variable; the water flow rate m is the manipulated variable. The water flow rate is maintained by a PI controller whose set point is to be supplied by the NLIC.

A simple process model may be obtained on the basis of a single heat balance relationship:

$$Cp * Vp * \rho * \frac{dTo}{dt} = \rho * m * Cp * (Ti - To) + U * A * (Ts - To) \quad (27)$$

where

T = water temperature along tube
$Ti = T(0, t)$ = inlet water temp., 11°C
$To = T(L, t)$ = outlet water temp., °C
m = water flow rate, $6.3 \times 10^{-5} < m < 3.2 \times 10^{-4}$ m^3/s
Ts = vapor temperature in annular space; varies from 20 to 100°C
Vp = total volume of pipe = 2.3×10^{-3} m^2
ρ = average density of water and pipe 990 kg/m^3
Cp = average heat capacity of water and pipe = 2,847 J/kg/K

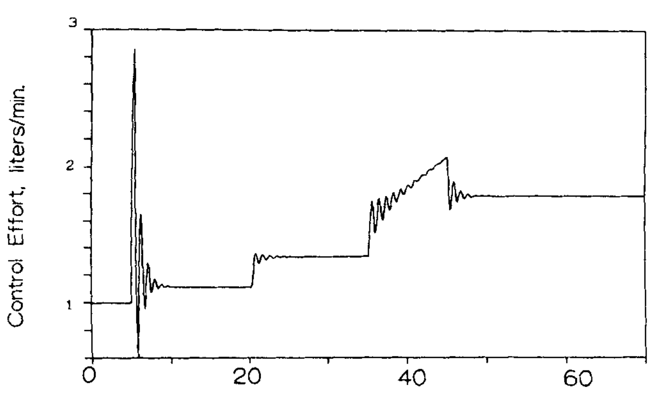

Figure 8. Control effort for response shown in Figure 7.

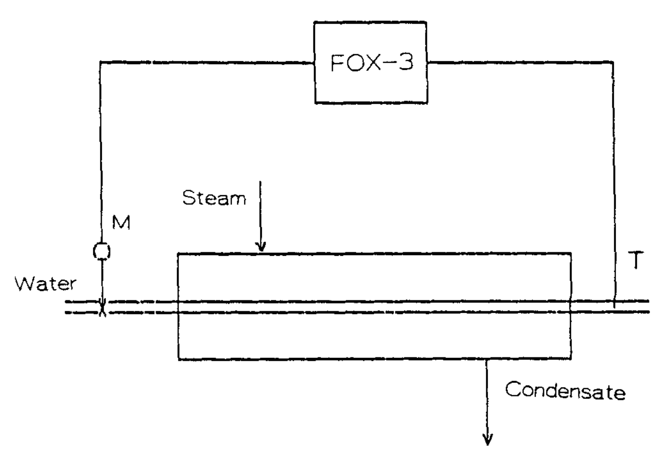

Figure 10. Heat exchanger process.

L = length of condenser = 2.4 m
U_H^* = overall heat transfer coefficient = 1,136 W/m^2/K
A = area for heat transfer = 0.15 m^2

Given the above parameters for the process model, Eq. 27 may be rewritten as

$$dTo/dt = -a' * m * To - b' * To + c' * m + u' \quad (28)$$

where $a' = 1{,}442/\text{m}^3$
$b' = 0.012/\text{s}$
$c' = 15{,}850°\text{C/m}^3$
u' = disturbance = $UA * Ts/Cp * D * \rho > 0$

The simple bilinear model of Eq. 28 will be used for the control. Note that several constraints exist on the process variables which limit the physically realizable region of operation for the model. These are:

$To > Ti$ (water temp. increases in the exchanger) (29)

$u' > 0$ (the disturbance is positive since the steam temperature is positive) (30)

$m \geq 0$ (the water flows in only one direction) (31)

A discrete form of the process model is chosen for implementation of NLIC;

$$(To)_{k+1} = (-a * m_{k+1} - b + 1) * (To)_k + u_{k+1} + c * m_{k+1} \quad (32)$$

where $a = a' * \Delta t$
$b = b' * \Delta t$
$c = c' * \Delta t$
$u = u' * \Delta t$

The only process measurement is the outlet water temperature. This measurement is relatively free of high-frequency noise (as determined by process observation) and so no filtering is needed. The estimator algorithm is chosen as

$$\hat{u}_k = To_k - (-a * m_k - b + 1) * To_{k-1} - c * m_k \quad (33)$$

The controller algorithm projects one step ahead. Setting $To = To^d$ and solving for the control effort m from Eq. 32, the controller algorithm is

$$m_{k+1} = [(To)^d_{k+1} + (b - 1) * (\hat{T}o)_k - \hat{u}_k] / (c - a * (\hat{T}o)) \quad (34)$$

where $(To)^d$ is the desired trajectory (from the trajectory generator).

A first-order trajectory generator is chosen because the model, Eq. 28, is first order. The filter is chosen as

$$TG = \Delta t/(\epsilon * (1 - z^{-1}) + \Delta t) \quad (35)$$

where z is the forward shift operator and ϵ is the filter time constant.

The above algorithms have been implemented on the process control computer and used in control of the heat exchanger process. Figure 11 contrasts the performance of the NLIC (with ϵ = 20 s) with a PI controller (reset time = 20 s, controller gain = −0.067), both of which were tuned to be stable over the entire region of operation ($6.3 \times 10^{-5} < m < 3.2 \times 10^{-4}$ m^3/s; $30 < Ts < 90$°C). Note that both the NLIC and PI controller operate in a cascade mode, that is, they provide the set point to the flow controller.

Conclusions

The results of the pH and heat exchanger control tests show that a substantial improvement in the process responses is obtained with NLIC as compared with PI control, in spite of modeling errors. Thus, we believe that for systems where the

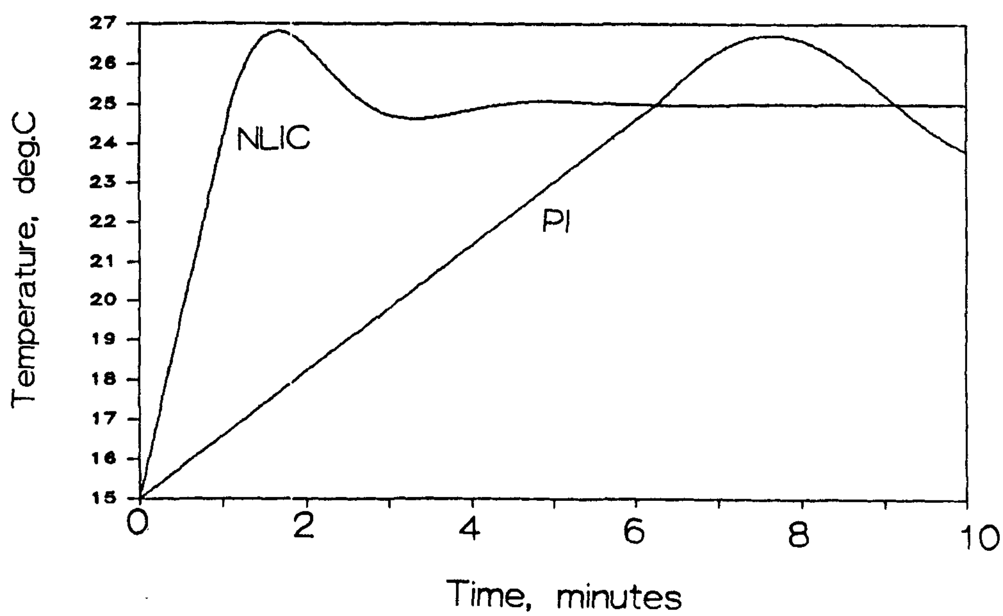

Figure 11. Heat exchanger response.

form of the nonlinearities is captured in the model, NLIC offers superior control. The degree of improvement in control using NLIC in place of linear methods will depend on the types of nonlinearities, the available measurements, and the nature of the modeling errors. Those processes that are poorly controlled using linear methods are good candidates for NLIC.

Notation

A = vector of fixed process parameters
d = control effort delay (no. of sample periods)
$D(s)$ = denominator polynomial of $\hat{G}(s)$
E = accumulated effect of unaccounted for disturbances influencing controlled variable
$f(s)$ = lag portion of trajectory generator filter
$F(s)$ = trajectory generator filter
G = transfer function relating control effort to controlled variable, or disturbance to controlled variable
h = function mapping process state to controlled variable
k = discrete time index
K = process steady state gain
m = control effort
n = order of trajectory generator filter
$N(s)$ = numerator polynomial of $\hat{G}(s)$
s = Laplace domain operator
Δt = sampling time
u = scaler disturbance to process
U = vector of disturbances to process
V = process set point
X = process state variable vector
y = measured controlled variable
y^d = desired system response
z = shift operator
ϵ = time constant of trajectory filter generator
θ = vector of process measurements
$\tilde{}$ = predicted value
$\hat{}$ = model variable, or approximate value

pH system example

Cr = concentration of base solution
D = measurement delay
F = feed rate to tank
Ka = acid/base equilibrium constant
Kw = equilibrium constant for water
m = flow rate of base (control variable)
$N+$ = concentration of cation
pH = pH of solution in tank
U = unknown concentration of acid and its salts (a disturbance)
V = volume of tank

Heat exchanger example

A = area for heat transfer
Cp = average heat capacity of water and pipe
L = length of condenser
m = water flow rate
P = average density of water and pipe
T = water temperature along tube
Ti = inlet water temperature
To = outlet water flow
Ts = vapor temperature in annular space
UH = overall heat transfer coefficient
Vp = volume of pipe

Appendix: Disturbance Estimation for Linear and Bilinear Processes

Consider the process described by

$$Y_k = q(Y_{k-1}, M_{k-1}) + \sum_i a_i u_k^i \qquad (A1)$$

where u_k^i = value of the ith disturbance at time t_k

q = a continuous function of Y_{k-1}, M_{k-1}

The model given by Eq. 1 is linear in the disturbances and is easily solved for the sum of the disturbances from the measured output at t_k and t_{k-1} and the previous control, M_{k-1}, as

$$\sum_i (a_i u_k^i) = U_k = Y_k - g(Y_{k-1}, M_{k-1}) \qquad (A2)$$

If we can obtain a disturbance model that predicts each of the future disturbance u_{k+1}^i as a function of the past disturbances, then we can also obtain a formula that predicts future values of the combined disturbances U_{k+1} from past estimates of the combined disturbances. For example, let future values of the disturbances, u_k^i, be predicted from past observations as

$$\hat{u}_{k+1}^i = \left(\sum_j d_j^i u_{k-j}^i\right) = L_i u_k^i \qquad j = 1, \ldots, n \qquad (A3)$$

where

$$L_i = \sum_j d_j^i E^{-j} \qquad j = 1, \ldots, n$$

and

$Eu_k^i = \hat{u}_{k+1}^i$ = value of u^i at $k+1$

n = number of past observations used to predict u^i at $k+1$

d_j^i = parameters associated with the disturbance model

Equation A3 can be used to obtain a prediction of the combined disturbance by operating on Eq. A2 with the product $(E - L_1)(E - L_2)(E - L_3)$. This gives

$$(E - L_1)(E - L_2)(E - L_3)U_k = 0 \qquad (A4)$$

since from Eq. A3

$$(E - L_i)u_k^i = 0, i = 1, 2, 3 \qquad (A5)$$

Expanding Eq. 4 gives

$$\hat{U}_{k+1} = \left[\sum_i L_i + E^{-1}(L_1L_2 + L_1L_3 + L_2L_3) + E^{-2}L_1L_2L_3\right] U_k \qquad (A6)$$

Equation A6 gives the rule for projecting U into the future from past observations of U. For the example above, the number of past observations required is $3n + 2$. If n is more than 2, it is likely that any practical scheme would have to approximate Eq. 6 with a simpler expression—possibly by neglecting the product terms.

In the above example, individual measurements of u^i can be used for feedforward control, which reduces the number of past measurements needed to estimate future values of the unmeasured disturbances. The latter is important because modeling errors propagate more strongly from more ancient data.

Consider now a model where the disturbances enter bilinearly as, for example,

$$y_k = -y_{k-1}m_k + u_k^1 m_k + u_k^2 \tag{A7}$$

where u^1 and u^2 are independent disturbances.

Even though the control m_j is known for $j = 1, \ldots, k$, it is not clear how to lump the disturbances. However, without combining disturbances it is still possible to obtain u^1 and u^2 from a single measurement with the aid of Eq. A5. Solving Eq. A7 for u^1 and u^2 and applying Eq. A5 gives

$$(E - L_1)[(y_k - u_k^2)/(m_k + y_{k-1})] = 0 \tag{A8a}$$

$$(E - L_2)(y_k + y_{k-1}m_k + u_k^1 m_k) = 0 \tag{A8b}$$

The solution of Eq. A8a together with Eq. A8b, if it exists and is unique, yields the desired estimate. Establishing the existence, uniqueness, and efficient methods of solution for problems like those given by Eq. A8 is an area of future research.

The above approach places heavy emphasis on the disturbance model given by Eq. A3 or, equivalently, Eq. A4. When an accurate disturbance model is not available (which will often be the case) then one should search for an additional measurement to supplement the primary measurement. The additional measurement should of course depend on either or both u^1 and u^2.

Literature Cited

Bridle, S., "Inferential Control of Processes with Riglet Half-Plane Zeroes," Control of Industrial Systems Res. Rept., (Nov. 21, 1985).

Brosilow, C. B., "The Structure and Design of Smith Predictors from the Viewpoint of Inferential Control," paper presented at JACC, Denver (1979).

Brosilow, C. B., and G. Q. Zhao, "Constrained Multivariable Control," *Proc. Am. Control Conf.*, San Diego (1984; also, *Advances in Control and Dynamic Systems*, **27**, to appear).

Cutler, C. R., and B. C. Ramaker, "Dynamic Matrix Control," Paper No. 516, 86th Nat AIChE Meet. (1979).

Garcia, C. E., and M. Morari, "Internal Model Controls: A Unifying Review and Some New Results," *Ind. Eng. Chem. Process Des. Dev.*, **21**, 308 (1982).

Gustafsson, T. K., "Calculation of the pH Value of a Mixture of Solutions—An Illustration of the Use of Reaction Invariants," *Chem. Eng. Sci.*, **39** (9), (1982).

Gustafsson, T. K., and K. V. Waller, "Fundamental Properties of Continuous pH Control," *ISA Trans.*, **22** (1) (1983).

———, "Oscillations in Feedback Systems for pH Control," Rep. No. 84-4, Dept. Chem. Eng., Abo Adademi, Finland (1984).

Hunt, L., R. Su, and G. Meyer, "Global Transformations of Nonlinear Systems," *IEEE Trans. Auto. Control*, **AC-28** (1), 24 (1983).

Joseph, B., and C. B. Brosilow, "Inferential Control of Processes," *AIChE J.*, **24** (3), 485 (May, 1978).

Kravaris, C., and C. Chung, "Nonlinear State Feedback Synthesis by Global Input/Output Linearization," *AIChE J.*, **33** (4), 592 (Apr., 1987).

Mehra, R. K., and R. Roulani, "Model Algorithmic Control Nonminimum Phase Systems," *Proc. JACC*, San Francisco (1980).

Meyer, G., and L. Cicolani, "Application of Nonlinear System Inverses to Automatic Flight Control Design—System Concepts and Flight Evaluations," *Theory and Applications of Optimal Control in Aerospace Systems*, P. Kant, ed., (1980).

Parrish, J. R., "Nonlinear Inferential Control," Ph.D. Diss., Case Western Reserve Univ. (1985).

Parrish, J. R., and C. B. Brosilow, "Inferential Control Applications," *Automatica* (Sept., 1985).

———, "Nonlinear Inferential Control of Reactor Concentration from Temperature and Flow Measurements," ACC, Seattle (1986).

Popiel, L., C. B. Brosilow, and T. Matsko, "Coordinated Control," *Chemical Process Control—CPC III*, Elsevier (1986).

Shine, S., "Development of Unit Operation Experiments: Heat Exchange and Gas Stripping," M.S. Thesis, Case Western Reserve Univ. (1980).

Shinskey, F. G., "Controls for Nonlinear Processes," *Chem. Eng.* (Mar. 19, 1962).

———, pH and pION Control in Process and Waste Streams, Wiley, New York (1973).

Manuscript received Dec. 27, 1985, and revision received Nov. 30, 1987.